Entity Linking to Wikipedia

Grounding entity mentions in natural language text using thematic context distance and collective search

Dissertation
zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

Anja Pilz
aus
Chemnitz

Bonn, 2015

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter:	Prof. Dr. Stefan Wrobel
2. Gutachter:	Prof. Dr. Kristian Kersting
Tag der Promotion:	21.09.2015
Erscheinungsjahr:	2016

Anja Pilz

Fraunhofer Institut für Intelligente Analyse und Informationssysteme IAIS

und

Rheinische Friedrich-Wilhelms-Universität Bonn,
Institut für Informatik III

Acknowledgements

First and foremost, I would like to thank Prof. Dr. Stefan Wrobel and Prof. Dr. Kristian Kersting for giving me the opportunity to work on my thesis in cooperation with the Computer Science Department at the University of Bonn and the Knowledge Discovery Department at Fraunhofer IAIS.

I would like to express my sincere gratitude to all people who have helped, inspired and accompanied me during my PhD studies. This thesis would not have been possible without the support from many other people and each of them contributed directly or indirectly to this thesis.

I am very grateful for the chance to pursuit my PhD studies in the Text Mining Group at Fraunhofer IAIS and have benefited greatly from the presence of all the supportive and involved people who made this an excellent working environment. These include all the former and current members of the Text Mining, STREAM and CAML groups, especially Melanie Knapp, Gerhard Paaß, Hannes Korte, Siehyun Strobel, Andre Bergholz, Florian Schulz, Kristian Kersting and Thomas Gärtner who all offered ideas, advise and support in many different ways. I am especially grateful to my co-author and mentor Gerhard Paaß whose support, encouragement but also criticism finally led to this thesis.

I also want to thank all the fellow PhD students at Fraunhofer IAIS for many fruitful discussions, the exchange of ideas and providing glimpses into other interesting machine learning fields: Fabian Hadji, Babak Ahmadi, Marion Neumann, Mirwaes Wahabzada, Katrin Ullrich, Daniel Paurat, Michael Kamp, Thomas Liebig, Olana Missura, Pascal Welke, Mario Boley, and Ahmed Jawad. Even though most of our group is now scattered across the world, I sincerely hope our paths will cross many times in the near and long future.

Most importantly I would like to thank Fabian Hadji, Babak Ahmadi and Marion Neumann, not only for critical and helpful discussions but also for their friendly support, backing and being honest friends. Especially Fabian's invaluable help and encouragement were indispensable for finishing this thesis.

Last but not least, I want to thank my family and friends, probably the hardest critics of all, for ceaseless support and asking the seemingly easy questions that turned out to be the most difficult and important to answer.

Abstract

This thesis proposes new methods for entity linking in natural language text that assigns entity mentions in unstructured natural language text to the semi-structured encyclopedia Wikipedia. Doing so, entity linking grounds a mention to an encyclopedic entry in Wikipedia and embeds it into this Linked-Open-Data hub. This enables a higher level view on single documents, provides hints for further reading and may be used to add details from other sources. Furthermore, enriching text documents with such links simultaneously resolves the ambiguity of entity names. This ambiguity is an unsolved challenge for many text mining applications: one entity may be designated by a multitude of names and every mention may denote a multitude of entities. Resolving the ambiguity of entity names is thus a crucial step for entity based retrieval, an open problem for most information retrieval and extraction tasks. For instance, search engines relying on heuristic string matches often retrieve irrelevant results as they can not satisfyingly resolve ambiguity.

Moreover, there is a huge number of entity mentions that can not be linked to Wikipedia since albeit of its size, Wikipedia has a restricted coverage. Earlier and current work often ignored this and consequently all mentions of uncovered entities. Other approaches handle only entity mentions of specific types or are focussed on English as target language. Apart from such restrictions, no method achieves perfect linking performance.

These are the tasks approached in this thesis. We introduce new methods for candidate entity retrieval and candidate entity consolidation, the key components to recall and precision, exploiting both the vast amount of structured and unstructured information stored in Wikipedia.

First, we propose a new contextual similarity measure based on latent topic distributions inferred from unstructured natural language text. We show that this thematic distance between mention and candidate entity contexts yields a lower linking error rate than purely word based distances. Being language independent, this method enables high performance entity linking in previously neglected languages such as German and French. This approach is especially suitable, albeit not restricted to link person names, the class of mentions with highest ambiguity.

We next propose a new candidate retrieval method to enable successful entity linking also for other entities that are not referenced canonically or exhibit the thematic coherence of persons. We introduce collective search that uses the structured information encoded in Wikipedia's hyperlink graph to arrive at sets of strongly related candidate entities. This enables us to better handle synonymy, one of the hardest problems in entity linking and not thoroughly treated in previous work. We emphasize on general applicability and evaluate this method on a broad collection of benchmark corpora both in a supervised as well as in an unsupervised setting. We show that candidate enhancement through collective search increases linking performance on nearly all of these corpora and that our method is the most stable compared to other state-of-the-art approaches. Presenting the first unification of diverse performance measures, we also make a step forward to the comparability of entity linking methods.

In conclusion, we provide state-of-the-art entity linking methods for nearly all of the current use cases. When it comes to fine-tuning, we note that entity linking has subjective aspects and adaptions may be necessary depending on the task at hand.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

1.1 Overview

In an information driven society the fast and reliable acquisition of information is of utmost importance. People are at every time of the day searching for information on political developments, job opportunities at newly funded companies, places they want to go, books they want to read, or movies they want to see. At the same time, people also produce a lot of content and contribute to the phenomenon called Web 2.0. They comment articles on news pages, create entries in online encyclopediae, post product reviews in online market places, pose questions or provide problem solutions in online fora, and a multitude of other things. The majority of this content is stored in unstructured natural language text which we first need to analyse to allow the focussed retrieval of information and to enable the extraction of knowledge or facts as a subsequent step. But when writing about a person or a product of interest, people usually do not give full attention to the potential ambiguity of a name, assuming that the interested reader will infer identity through background knowledge or the context expressed in the document. Hence, to retrieve information about specific entities, we first need to identify these entities by assigning their references in a text to a resource providing unique identifiers of these entities.

In this thesis, we link entity mentions against the online encyclopedia Wikipedia. Grounding a textual entity mention to an entry in Wikipedia, we identify the entry in this encyclopedia that corresponds to the underlying entity of a mention. To highlight the difference between mentions and entities, we will use specific fonts for mentions as well as ENTITIES in the remainder of this thesis. Since each article in Wikipedia is uniquely identified through its title, entity linking against Wikipedia enables the distinction among different entities. Based on the unique identifiers predicted through entity linking, entity linking enables entity-based retrieval instead of keyword search, i.e. *things, not strings.* Thus, entity linking allows to aggregate the retrievable information about a specific entity into a more actionable set. We argue that generating a link between a mention and its corresponding entity in Wikipedia determines the identity of the mention's underlying entity and grounds the mention to a unique representative which also resolves potential ambiguity. Doing so, we

Figure 1.1: Entity linking enriches text documents with links to Wikipedia. This is here shown for a mention of Barack Obama in a German online news paper.

assign an unstructured piece of text to a (semi-structured) encyclopaedia entry. As a side effect, Wikipedia provides additional semantic information that can be used to enrich the context of a mention with encyclopaedic knowledge.

Fig. 1.1 illustrates entity linking against Wikipedia. The figure shows a screen shot of a system that automatically extracts mentions of named entities in German newspaper articles and links these mentions to articles in the German Wikipedia. In the figure, mentions of entities are marked by type: persons in blue (e.g. Barack Obama), locations in red (e.g. New York) and organizational entities in green (e.g. SPIEGEL). As depicted for the mention Barack Obama, entity linking aligns a mention with the encyclopedia Wikipedia by assigning it to the Wikipedia article corresponding to the referenced entity, here Barack Obama.

In this example, one might argue that a simple heuristic string matching technique might be sufficient for linkage since the name Barack Obama is unique in the German version of Wikipedia. However, this is not the case for the English Wikipedia. The English version contains another article with a very similar title, namely the article on the father Barack Obama, Sr. Furthermore, the mention SPIEGEL is not only the name of the German news magazine Der Spiegel, but also the German word for mirror. Therefore we need to resolve the synonymy of entity names when linking to Wikipedia.

Again, one might argue that there are many more references of Barack Obama in news papers, web pages and other sources. Then, linking every mention Barack Obama to the most popular candidate in Wikipedia, i.e. Barack Obama instead of Barack Obama, Sr., might result in a linking accuracy of more than 90%. This is a reasonable assumption, since often a name is tied by frequency of reference to a high popularity entity that is far more often referenced than its namesakes. However, such a naive model will fail to correctly link many other mentions. There is a huge number of entity mentions for which either no such high popularity candidate exists or where the seemingly most obvious candidate entity is not the correct one. Especially in local newspapers we may find a mention Helmut Schmidt that does neither refer to the former German chancellor nor to one of the two soccer players of that name covered in Wikipedia but to some entirely different, **uncovered** person.

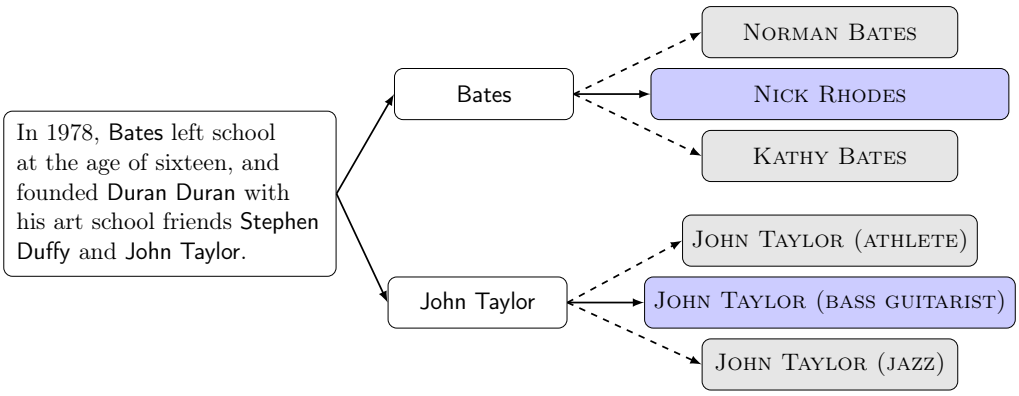


Figure 1.2: Entity Linking needs to handle synonymy and polysemy to detect the true underlying entity of a mention (blue) among potential candidates (grey).

Even though heuristic measures such as string similarity or popularity priors are strong indicators, they are not sufficient to arrive at a linking model with both high precision and recall. Entity linking methods based solely on popularity priors are likely to retrieve all mentions of popular entities but none for other, less well known entities. Consequently, linking mentions to the most popular entity may lead to a low recall for under-represented entities for which in turn new information is most beneficial. On the other hand, it may also lead to the erroneous assignments of mentions to high popularity entities. The **focus on popularity** is one of three common shortcomings of related approaches to entity linking and also leads to the ignorance of mentions of uncovered entities. This thesis tackles both mentions of covered as well as uncovered entities.

Another common shortcoming of related work is the **focus on English** and hence also the **focus on specific corpora in English**. But the two challenges of polysemy and synonymy also apply to many other languages. Polysemy means that one name may denote many different entities. Synonymy means that one entity may be known under different names. This thesis presents the first approach that performed entity linking for German. Additionally we present a linking model for French and also tackle English as the most prominent language in entity linking to allow a better comparability with related work. The next example therefore uses the English Wikipedia to further illustrate the problems of synonymy and polysemy.

Fig. 1.2 shows a text snippet taken from Wikipedia and both the true underlying entities as well as potential candidates for a selection of mentions. The mention Bates in this context refers to the singer Nick Rhodes whose birth name is Nicholas James Bates. Again, note that a naive matching based solely on character overlap between mention and entity name in Wikipedia would not link Bates to Nick Rhodes but to one of the more obvious candidates such as the fictional character Norman Bates or the actress Kathy Bates.

Table 1.1: Most frequent person names in Wikipedia with number of articles and the number of search results for each name in WhitePages and Google. WhitePages lists distinct persons that live in the United States and are listed in public sources. Google hits are not grouped by underlying entity and thus much more diverse[1].

person name	Wikipedia	WhitePages	Google
John Smith	52	34968	3.6 Million
John Campbell	51	8242	2 Million
John Williams	50	25657	3.7 Million
John Taylor	50	13383	3.9 Million
John Anderson	47	14716	2.9 Million

There are many more examples for the usage of synonyms, among those nick names for cities (Big Apple – New York, Charm City – Baltimore) or soccer teams (Equipe tricolore – French national soccer team) but also entities that underwent a name change (Burma – Myanmar, Datsun – Nissan). Furthermore, acronyms are commonly used in many texts. For instance, the acronym NBA may stand for National Bar Association, National Boxing Association and National Basketball Association. The usage of acronyms significantly increases the number of synonyms for an entity and simultaneously the polysemy among entity names.

Synonymy is non-trivial to resolve but requires sophisticated candidate retrieval techniques and carefully designed alias dictionaries. Without them, we would not be able to handle synonymy and fail to retrieve the true underlying entity in substantially many cases. The literature has proposed a number of possible alias resources, the most prominent is Wikipedia itself. Providing the assets to create comprehensive alias dictionaries in its hyperlink and redirect structures, Wikipedia is superior to a simple entity catalogue and also other encyclopedias.

Fig. 1.2 also illustrates the polysemy of names as challenge in entity linking. The mention John Taylor refers to John Taylor (bass guitarist), one of 52 articles in the English version of Wikipedia[2] describing a person called John Taylor (see also Tab. 1.1). Even though there are more than 52 candidate articles in Wikipedia, this obviously covers only a fraction of the actual number of persons called that name. Note that the polysemy of names is not resolved by common search engines: matching a query term against the textual content of web sites will usually return all pages containing the term, without distinction among the underlying entities. This is illustrated in Tab. 1.1 through a snapshot of the five most frequent person

[1]Figures retrieved in July 2014.
[2]Retrieved from the Wikipedia version of September 1st, 2011.

names in Wikipedia. The figures retrieved from WhitePages[1] show that there are 34968 distinct persons named John Smith who live in the United States and are recorded in public sources. Obviously, this is merely a lower bound on the worldwide population. The high polysemy of names is also reflected in the number of Google search results: a search for John Smith returns about 3.6 million results. These results are not grouped by underlying entities and first need to be analysed to retrieve sources for one specific individual. At this point, it is noteworthy that Google's disambiguation module distinguishes only among popular entities such as George W. Bush and George H. W. Bush. It does not directly provide the means to distinguish less popular entities such as the journalist Michael Jackson (writer) from the famous singer Michael Jackson.

Thus, entity linking is also entity disambiguation as it resolves the potential ambiguity in entity names. Resolving synonymy means that we retrieve all relevant candidates, resolving polysemy means that we assign a mention to at most one entity in Wikipedia. Using machine learning methods such as classification and ranking, we predict links using contextual and relational attributes. Each predicted link then either grounds a mention to an article in Wikipedia or states that this mention is not covered, i.e. not linkable. The importance of the latter is emphasized by the gap between the number of John Smith's listed in Wikipedia and the number listed in public sources and thus potentially mentioned in any piece of text to be analysed.

The last two examples showed entity linking for named entities, a specific class of mentions. Most notably, named entities such as persons (Nick Rhodes, John Taylor) are unique individuals. Many approaches such as Cucerzan [2007], Hoffart et al. [2011b], Mendes et al. [2011] or Ploch [2011] treat only mentions of such specific type or are even more focussed by linking only person names (Bunescu and Pasca [2006]). In these approaches, a mention referring to the mirror instead of the news paper, would not be linked.

The **focus on entity type** is the third common shortcoming of related approaches. It leads to a more restricted set of predicted links that neglects many other entities, while at the same time also heavily depending on the quality of preceding natural language processing models. In this thesis, we start with the linking of person names, the mention class with highest ambiguity, and then move on to more general, possibly abstract entities or concepts. We assume that an entity mention to be linked may refer to any existing being, e.g. a person or a location, but also an abstract concept such as a thing or an object. This can be more difficult compared to person name linking, where underlying entities are unique and have other characteristic properties. For instance, person name mentions often exhibit a strong thematic coherence and furthermore, at least in editorial texts, these names are often canonical. This need not be the case for other entities. Many entities may be mentioned in a text without evident relation to the thematic content, e.g.

[1] `http://names.whitepages.com`

locations are often mentioned as a geographical anchor at the beginning of news articles. Some entity mentions may also require relational clues such as the co-occurrence with other entities where the influential entity or factor first needs to be detected.

Thus, we also approach the more general task of word sense disambiguation where a mention may refer to a conceptual or abstract entity such as BASS that subsumes all the individuals belonging to this species of fish. Here, mentions referring to abstract concepts subsuming different individuals that are not distinguishable by a rigid designator, are encompassed by the best fitting concept they belong to. While not explicitly excluding adjectives and verbs, we focus on entities or concepts usually denoted by nouns or noun phrases. This is more general than named entity disambiguation since we aim at linking mentions independent of their type. Also, research in word sense disambiguation does usually not handle named entities or proper nouns. Further, it generally assumes a complete sense inventory containing all possible senses of a word. This assumption does not hold for our approach since, albeit of its size, Wikipedia has restricted coverage. Note that there exists also no inventory covering all persons in the world.

Consequently, we need to handle mentions denoting entities that are not covered in Wikipedia. This means also means that we need to account for entity mentions that may have a candidate in Wikipedia but do indeed refer to somebody or something not represented by an article in Wikipedia. If entity linking can not retrieve a corresponding entry in Wikipedia, which should only be the case if Wikipedia does not cover it, entity linking should state that a specific entity mention does not relate to any of the known ones but refers to an unknown or uncovered entity that may require further investigation. This important aspect is not approached thoroughly in the related work. For example, Hoffart et al. [2011b] explicitly ignore entities that can not be linked to the knowledge base YAGO, a derivative of Wikipedia (Suchanek et al. [2008]).

In contrast, this thesis does not assume completeness or ignores mentions of uncovered entities but aims at distinguishing between linkable (covered) and not-linkable (uncovered) entities. This thesis shows methods that solve the problem of linking mentions in unstructured natural language text to entities in Wikipedia and provides state-of-the-art methods, ranging from person name disambiguation to general entity linking treating various kinds of entities. Even though there are other potential resources for entity linking, we chose Wikipedia. We aim at linking both named entities and conceptual or abstract entities from a broad range of topics. Here, Wikipedia is the first choice due to its coverage, its availability in many languages, which allows us to formulate linking for other languages and other reason that will be detailed in the subsequent chapters of this thesis. Wikipedia is the most widely used resource for entity linking. As a result of its prominence, the terms entity linking and *Wikification* are also used interchangeably in the literature. Apart from the massive benefits provided by Wikipedia that facilitate entity linking, using

Wikipedia in this thesis also allows for a better comparability with related work. Among those benefits are the textual descriptions of entities that are comparable to mention contexts when using appropriate measures and the hyperlink graph that allows the extraction of extensive alias dictionaries, the computation of semantic relatedness on an entity level and, perhaps most importantly, the extraction of disambiguated example collections necessary to construct supervised linking models for several languages. For more specific tasks, there exist also other databases such as DBLP[1] that can be used for author disambiguation, the gene database Entrez Gene (Maglott et al. [2011]) for biological contexts, and many more. However, other resources are very specific and techniques usually do not generalize.

Having described entity linking to Wikipedia, its necessity and the challenges in approaching it, we will conclude this introduction with an outline of this thesis and detail the contributions made by it.

1.2 Outline and Contributions

This thesis presents solutions to the following open challenges in entity linking:

Polysemy We propose models that choose the true underlying entity when multiple candidates are given for an ambiguous mention. By weighting contextual and relational evidence against popularity priors, we avoid erroneous linking of mentions to the most popular candidate and thus increase linking precision.

Synonymy We propose a model that retrieves comprehensive sets of relevant candidates which remarkably increases linking recall and thus enables a wide range of potential applications. This is shown empirically on a representative collection of benchmark corpora from varying sources, topics and linking tasks.

Uncovered entities We propose models that learn whether a mention refers to an uncovered entity without the need for human interference such as manual threshold adaptions.

While thoroughly taking into account mentions of uncovered entities and tackling them through an abstracted concept in the proposed linking models, we focus on real-world entities covered in Wikipedia. By grounding mentions to Wikipedia, we align textual name appearances with unique entity definitions of real-world entities provided in Wikipedia. This is an attractive solution as we do not only do a step forward to resolve polysemy and synonymy but also provide the means to retrieve additional information from the semi-structured encyclopedia Wikipedia.

We are not the first to observe the manifold benefits of using Wikipedia as dedicated knowledge base and entity linking to Wikipedia has received much scientific

[1] `http://www.dblp.org/db/`

attention in recent years. We describe most of these benefits and the structures they arise from in Chapter 2. Simultaneously, we show how the resources provided by Wikipedia can be used to build supervised entity linking models and, based upon this, formulate entity linking as a supervised classification or ranking task. This is compared to alternative approaches in a comprehensive overview of the research and state-of-the-art in entity linking and related tasks, ranging from person name disambiguation and word sense disambiguation to general entity linking. We focus on approaches using Wikipedia since these are most relevant to this thesis and point out how they compare to the methods proposed in this thesis.

We explicitly avoid building models for specific entities. Instead of creating one profile for every entity covered in Wikipedia, and consequently creating one model per name or mention, we aim for methods that can link mentions to entities previously unknown to the model. This is realized through models based on similarity measures defined independent of specific entities and thoroughly described in the first part of this thesis, Chapter 3.

This part is concerned with the disambiguation of person names and focused on resolving polysemy based on contextual similarity measures to resolve the remarkably high ambiguity of person names. Opposed to record linkage in structured data, entity linking in text needs to interpret unstructured input data. Exploiting the simultaneous presence of unstructured text in Wikipedia articles we formulate similarity functions based on contextual or *thematic* similarity. We propose different formulations to measure the contextual similarity between mentions and entities and compare them to related approaches using the results published in Pilz et al. [2009], Pilz and Paaß [2009], Pilz [2010] and Pilz and Paaß [2011].

In Pilz and Paaß [2011] we approach entity linking using thematic information derived from Latent Dirichlet Allocation. We create topic models over the unstructured natural language content of Wikipedia articles and use topic probability distributions derived from this model to compare the context of a mention with the contexts of candidate entities in Wikipedia. We evaluate various distances over topic distributions in a supervised classification setting to find the most suitable candidate entity, which is either covered in Wikipedia or unknown. This chapter covers the following contributions:

- Both ambiguous as well as unambiguous person names can very reliably be linked to their true entity using thematic distances derived from topic models. We compare this method to the most relevant, categorization-based, approach of Bunescu and Pasca [2006] and show that our method achieves significantly better results in predictive performance, regarding both entities covered in Wikipedia as well as uncovered entities. Thematic context distances are more general than purely word based context distances and especially well suited for linking against the biographical person entries in Wikipedia.

- Using unsupervised topic models, we propose the first method for person name

disambiguation that is applicable in more than one language with one and the same methodology. We exploit the availability of Wikipedia in multiple language versions to design the first language independent entity linking models. We empirically show that we obtain equally good results for person name disambiguation using the English, the German and the French Wikipedia and conclude that this design prevents our method from being restricted to texts in specific languages.

In the second part of this thesis, in Chapter 4, we generalize to entities of arbitrary type and other abstract concepts. We propose a new retrieval engine that allows the linkage and disambiguation of nearly all kinds of terms. In contrast to approaches restricted to handling only named entities of specific types, the proposed method has a higher coverage, since ambiguous terms such as `tree` are not treated by most named entity linking systems, as they are not recognized as named entities. This method is potentially more stable since errors made by named entity recognition models are less harmful. We may use the type information but do not exclusively rely on it, which renders our method also applicable for languages where the development of named entity recognition models is more difficult compared to English.

In Pilz and Paaß [2012] we propose a collective linking method that exploits both contextual as well as relational evidence encoded in inverted indices. We combine efficient search methods over these indices for candidate retrieval with a supervised ranking method to automatically fine-tune retrieved results and detect mentions of uncovered entities. This chapter covers the following contributions:

- We propose a powerful high-recall candidate retrieval engine based on inverted indices. We create a term-based index over Wikipedia article texts to retrieve candidates from local, contextual clues and combine these with global, relational information derived from the link structure of Wikipedia. The latter is encoded in an auxiliary index which allows us to exploit relational information expressed implicitly through the co-occurrence of entities.

- We treat all mentions in a document simultaneously in a collective search query over the indexed link graph to arrive at coherent candidate sets. We propose new coherence measures that can efficiently be computed using these indices and embed each mention into a more globalized view in the hyperlink graph. We show that exploiting the co-occurrence of entities in this graph is a highly reliable method to link mentions appearing in contexts where thematic clues might not be available. This completely unsupervised method is already able to correctly link most mentions to their true underlying entity while at the same time scalable and memory efficient.

- We combine unsupervised candidate retrieval and supervised ranking methods to validate the retrieved candidates. This further increases linking precision

and enables us to reliably detect mentions of uncovered entities without the need to manually set boundaries on similarity thresholds. For the ranking method, we also use additional information such as entity popularity or thematic attributes that were not available in the unsupervised search step.

- We show that the proposed method has a high general applicability and validate this claim in a thorough empirical evaluation over most relevant entity linking benchmark corpora. While most approaches in the literature are evaluated only on specific corpora with distinct goals, treating only named entities or even ignoring the concept of uncovered entities, our entity type independent method achieves superior results to most other approaches.

The final chapter concludes this thesis with a summary, integrates the findings and discusses shortcomings, advantages and potential future directions as well as applications.

Publications

The main contributions of this thesis have been published by the author in the following publications.

Conference Papers:

- Pilz and Paaß [2011]: **Anja Pilz** and Gerhard Paaß. From Names to Entities using Thematic Context Distance. In *Proceedings of 20th ACM Conference on Information and Knowledge Management (CIKM 2011)*, pages 857–866, Glasgow, Scotland, UK, October 2011. ACM.

- Pilz and Paaß [2012]: **Anja Pilz** and Gerhard Paaß. Collective Search for Concept Disambiguation. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 2243–2258, Mumbai, India, December 2012. The COLING 2012 Organizing Committee.

Workshop Papers:

- Pilz [2010]: **Anja Pilz**. Entity Disambiguation using Link based Relations extracted from Wikipedia. In *First Workshop on Automated knowledge base Construction (AKBC 2010)*, Grenoble, France, May 2010.

- Pilz and Paaß [2009]: **Anja Pilz** and Gerhard Paaß. Named Entity Resolution using Automatically Extracted Semantic Information. In *Workshop on Knowledge Discovery, Data Mining, and Machine Learning (KDML 2009)*, pages 84–91, Darmstadt, Germany, September 2009.

- Pilz et al. [2009]: **Anja Pilz**, Lukas Molzberger, and Gerhard Paaß. Entity Resolution by Kernel Methods. In *Proceedings of the SABRE Conference on Text Mining Services (TMS 2009)*, pages 71–80, Leipzig, Germany, March 2009.

Other publications:

- Paaß et al. [2012]: Gerhard Paaß, Andre Bergholz, and **Anja Pilz**. A Knowledge-extraction Approach to Identify and Present Verbatim Quotes in Free Text. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies (i-KNOW 2012)*, pages 31:1–31:4, Graz, Austria, September 2012. ACM.

- Wahabzada et al. [2011]: Mirwaes Wahabzada, Kristian Kersting, **Anja Pilz**, and Christian Bauckhage. More influence means less work: Fast latent Dirichlet allocation by influence scheduling. In *Proceedings of 20th ACM Conference on Information and Knowledge Management (CIKM 2011)*, Glasgow, Scotland, UK, October 2011. ACM. (Poster Paper).

- Paaß et al. [2009]: Gerhard Paaß, **Anja Pilz**, and Jochen Schwenninger. Named Entity Recognition of Spoken Documents Using Subword Units. In *Proceedings of the third IEEE International Conference on Semantic Computing (ICSC 2009)*, pages 529- 534, Berkeley, CA, USA, September 2009. IEEE.

Chapter 2

Entity Linking: Preliminaries

Outline

This chapter provides the preliminaries for the main contributions of this thesis and introduces the basic notation. Handling unstructured natural language text, we first introduce the relevant notions of natural language processing (Section 2.1). We then introduce entity linking to Wikipedia (Section 2.2) and formally define the entity linking task in this context. In Section 2.3, we describe how entities are represented in Wikipedia and depict their interlinkage through Wikipedia's hyperlink graph in Section 2.4. This graph provides the means to extract grounded example data sets and to compute important figures such as semantic relatedness. We conclude this chapter with a general overview of related work and explain the major differences among recent approaches in entity linking to Wikipedia in Section 2.5.

2.1 Notions from Natural Language Processing

In this thesis, we consider as input natural language text containing *entity mentions* that are to be linked to Wikipedia. An entity mention is a *proper name* or *name phrase* appearing in a textual *context*, for instance a news paper article. A mention may consist of a single word or a sequence of words that jointly constitute the surface form of a mention. To link such a mention to an article in Wikipedia, we need to analyse unstructured natural language text which relates entity linking to the general task of natural language processing (NLP). Before we describe Wikipedia and entity linking to Wikipedia, we will therefore first give a very condensed overview of NLP.

NLP has constantly achieved much attention in the scientific community and is the core component for many Text Mining use cases due to the huge amount of information stored in unstructured natural language text. NLP thus covers a broad range of topics such as Part-of-Speech tagging, named entity recognition, relation extraction, or co-reference resolution. Since a thorough overview of NLP is out of the scope of this thesis, we refer the interested reader to Jurafsky and Martin [2009] or Aggarwal and Zhai [2012] and focus here on the **name phrase extraction** and **named entity recognition** tasks. These are most relevant for entity linking since

they can be used to identify potential or interesting entity mentions and thus also dictate the nature of the linking approach.

For entity linking, we are mostly concerned with name phrases that are usually formed of consecutive nouns. Here, the first step is to assign a sequence of part-of-speech tags to a sequence of words. Part-of-speech tags, or short PoS tags, are word classes and encompass nouns, verbs, adjectives and adverbs. They can be assigned by so called PoS taggers, statistical models that categorize words into one of these classes. Noun or name phrases can then be detected using chunking that analyses word sequences and their PoS tags grouping consecutive nouns to noun phrases. Other phrase types are noun phrases that also contain adjectives, e.g. "great food". Such phrases are of particular interest for research in opinion mining (Liu and Zhang [2012]).

While chunking may be used to extract all noun phrases in a given context, named entity recognition (NER) extracts word sequences that are the proper names of *named entities*. A named entity is a concrete being of a specific type where the most common types are *person*, *location* (places or sites) or *organization* (clubs, companies, etc.) (Sang and Meulder [2003], Nadeau and Sekine [2007]).

Technically, the definition of a named entity is a philosophical question. According to Kripke [1980], a named entity is an entity for which one or more proper names (rigid designators) exist. Given its proper name, a named entity is assumed to be unique over all contexts it appears in. This distinguishes a named entity such as Albert Einstein from an abstract entity such as the fish species bass. For instance, without context, it is unclear whether either the fish on Paul's or the fish on Michael's plate is meant while presumably the person Albert Einstein is unique.

Given the ambiguity of entity names, the definition of Kripke [1980] may be considered questionable. However, there is a generally accepted interpretation in most NER tasks and challenges. The proposed NER models are usually sequential statistical models that use PoS tags and contextual clues to assign a sequence of words denoting a name phrase to one of the above mentioned types. PoS tags are important since the proper names of named entities are usually noun phrases and contextual clues from neighbouring activity verbs and conjunctions distinguish locations from persons.

The focus of this thesis is entity linking and, as already stated, we assume mentions to be linked have already been extracted. Instead of investigating new models for phrase or named entity extraction, we focus on the inherent challenges of entity linking: resolving *polysemy* and *synonymy*. Due to the absence of compulsory naming rules and the ambiguity of natural language in general, one mention may denote a multitude of different entities (polysemy) and one and the same entity may be referenced with various mentions of different surface forms (synonymy). Because of synonymy and polysemy, we observe a many-to-many mapping between mentions and entities which we need to resolve in order to achieve high linking precision and recall.

Given a text document with an arbitrary number of mentions, we want to identify each mention's underlying entity in order to render it usable for semantic search or other information retrieval tasks. Semantic retrieval requires not only the detection of entity mentions but also the identification of the unique underlying entity of a mention. Using Wikipedia as collection of target entities is a natural choice since the entities covered in Wikipedia are not only uniquely identified but also mirrored in the Linked Open Data hub DBpedia (Bizer et al. [2009]).

Natural Language Processing using Wikipedia

In the last years, Wikipedia has been widely used for NLP and other related research tasks. Wikipedia is a large corpus of crowd knowledge that can be used as a background corpus or training corpus for diverse tasks. For instance, Wikipedia was used for automatic summarization (Woodsend and Lapata [2011]), text categorization (Gabrilovich and Markovitch [2006]), indexing (Medelyan et al. [2008]), clustering (Banerjee et al. [2007]), searching (Milne et al. [2007]), knowledge modelling (Ponzetto and Strube [2011]) but also NER (Nothman et al. [2009]) and relation extraction (Yan et al. [2009]), for instance learnt from the nearly structured information encoded in its infoboxes (Wu and Weld [2007]).

The Semantic Web aims to annotate natural language text with semantic markup that renders web resources interpretable or at least processable for computers (Shadbolt et al. [2006]). This is the basis for semantic retrieval and obviously entity linking is a subtask of this long term goal. The idea of the Semantic Web spurred the research in entity linking and led to a publication flood in the neighbouring scientific communities of NLP, data mining, knowledge discovery and knowledge management. Nearly all of the research in entity linking relies on the presence of knowledge bases or other resources providing target entities against which mentions are to be linked. This process was triggered by the emergence of Wikipedia that quickly became the most prominent resource for entity linking. In the next sections, we will specify entity linking against Wikipedia, describe the exploitable assets Wikipedia provides and give a general overview of related work.

2.2 Entity Linking to Wikipedia

Wikipedia is an encyclopedia covering a broad range of topics. It includes succinct but comprehensive descriptions of persons, sport events, pieces of art, general concepts from computer science, history and medicine and many more. Strube and Ponzetto [2006] found Wikipedia to have an accuracy and coverage similar to Encyclopedia Britannica[1], an English expert reviewed reference book covering all branches of knowledge. This renders Wikipedia an ideal collection of target entities

[1] `http://www.britannica.com/`

for entity linking. Using Wikipedia as resource, entity linking assigns mentions of entities appearing in text to a uniquely identified article in this encyclopedia.

Notation (Wikipedia as collection of entities)
The collection of articles in Wikipedia is denoted by $\boldsymbol{\mathcal{W}} = \{e_1, \ldots, e_{|\boldsymbol{\mathcal{W}}|}\}$. Each article $e \in \boldsymbol{\mathcal{W}}$ represents and describes one entity e.

Technically, Wikipedia is a collection of interlinked web pages that includes also many meta-pages, disambiguation pages and listings. Such pages are excluded from $\boldsymbol{\mathcal{W}}$ since we do not consider them articles providing textual entity descriptions. Furthermore, albeit of its size[1], Wikipedia covers only a subset of all real-world entities. Consequently, there are many mentions that can not be linked to an entity in Wikipedia. Due to this restricted coverage, entity linking should handle mentions of *uncovered entities* for which no corresponding entity exists in Wikipedia. In this thesis, we thoroughly handle uncovered entities, but since concentrating on linking mentions of entities for which reference information is available, we do not distinguish among these uncovered entities and use the following collective notation.

Notation (Uncovered entities)
We use the place-holder NIL to subsume uncovered entities for which mentions can not be linked to a corresponding entity in $\boldsymbol{\mathcal{W}}$.

Taking into account entities in Wikipedia as well as uncovered entities, we define the entity linking task as follows. Entity linking assigns a textual mention m either to its corresponding entity $e \in \boldsymbol{\mathcal{W}}$ or states that m is not covered in $\boldsymbol{\mathcal{W}}$ by assigning m to NIL. This can be described through a linking model or a linking function $f : m \mapsto \boldsymbol{\mathcal{W}} \cup \{\text{NIL}\}$ with

$$f : m \mapsto \begin{cases} e_i \in \boldsymbol{\mathcal{W}}, & \text{if } e_i \text{ is the corresponding entity of } m \text{ in } \boldsymbol{\mathcal{W}}, \\ \text{NIL}, & \text{if } m \text{ has no corresponding entity in } \boldsymbol{\mathcal{W}}. \end{cases} \tag{2.1}$$

In the following, the context and all other attributes of a mention are implicitly indicated by m. When referring to specific attributes, we will distinguish among them using individual notation. For instance, we use $name(m)$ to denote the surface form of a mention, $text(m)$ to denote its context and $type(m)$ for its named entity type, e.g. $type(m) = person$.

Fig. 2.1 depicts this formulation of entity linking. The dotted shape of the real-world entity set $\boldsymbol{\mathcal{E}}$ indicates that the cardinality of this set is not determinable in practice. Due to the absence of a context independent naming scheme we can not distinguish among all real-world entities and therefore also not determine the

[1]There exist 3 million articles in the English, 1.2 million articles in the German and 1 million articles in the French version as of July, 2013.

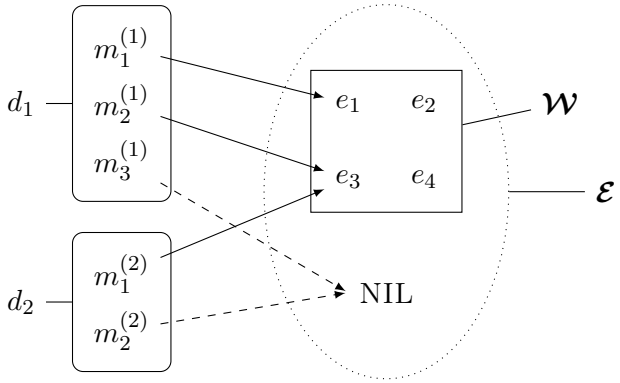


Figure 2.1: Entity linking grounds a mention m, given its context in a document d, either to an entity in the encyclopedia $\mathcal{W}$ that covers a subset of all real-world entities $\mathcal{E}$, or to the representative NIL subsuming all uncovered entities.

cardinality of $\mathcal{E}$. In contrast, the cardinality of $\mathcal{W}$ is given by the number of distinct articles that are uniquely identified through the respective Wikipedia URL.

Now, the output of a linking function f is correct, if the predicted entity corresponds to the underlying ground truth target entity of a mention. We denote the ground truth entity of a mention as follows.

Notation (Ground truth entity)
The true underlying entity referenced by a mention m is denoted by $e^+(m)$.

Following Eq. 2.1, the true underlying entity of a mention is either an entity in Wikipedia or NIL, i.e. $e^+(m) \in \mathcal{W}$ or $e^+(m) = \text{NIL}$. If the ground truth entity corresponds to an entity in Wikipedia, the linking model should predict the title of the corresponding article. To do so, it must distinguish among all entities in Wikipedia that can be a *candidate* for the underlying entity of a mention m.

Notation (Candidate entity)
We denote a potential match for the true underlying entity of a mention m as *candidate entity* $e(m) \in \mathbf{e}(m) = \{e_i(m)\}_{i=1}^{|\mathbf{e}(m)|}$ where $\mathbf{e}(m)$ is the set of all candidates.

To illustrate the notion of candidate entities, the next example shows mentions that each correspond to an entity in Wikipedia but yield several potential candidate entities (see also Fig. 2.2).

> **Example 1**
> For each mention m_i of a person in Fig. 2.2, the figure shows the true entity $e^+(m_i)$ and, if available, two other candidates $e_1(m_i)$ and $e_2(m_i)$. Linking the mentions m_i to entities in Wikipedia should result in the following assignments

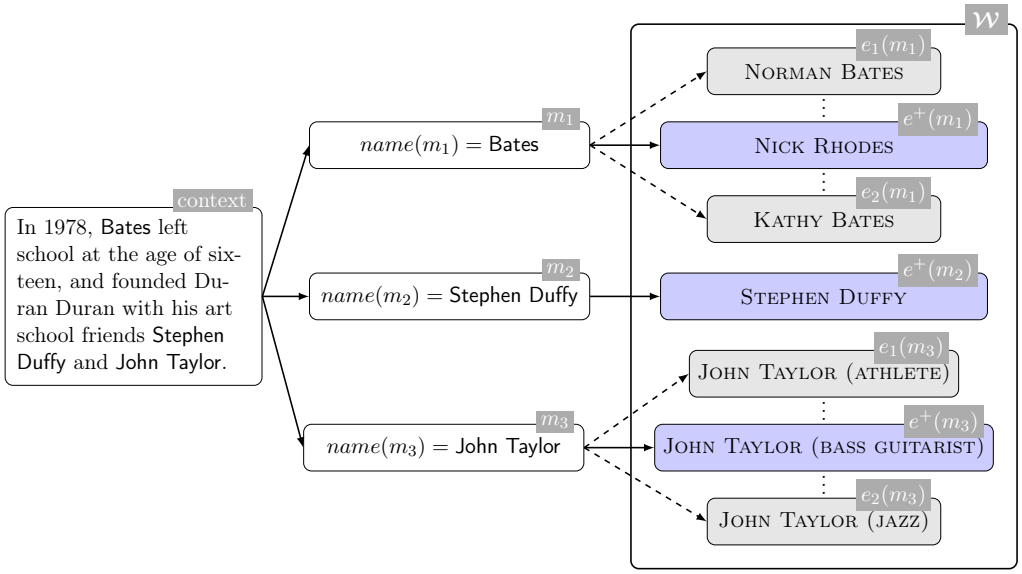


Figure 2.2: Entity Linking grounds textual mentions of entities, here persons (in bold), to entries in Wikipedia ($\boldsymbol{\mathcal{W}}$). To correctly link a mention m_i to its true entity $e^+(m_i)$ (blue), we need to retrieve all relevant candidates $e_i(m_i)$ (grey) and detect the true entity among these candidates. To avoid clutter, the figure shows only a selection of all potential candidates and omits the NIL candidate for each mention.

where $e^+(m)$ is the true underlying entity of m:

$$e^+(\text{Bates}) = \text{Nick Rhodes}$$
$$e^+(\text{Stephen Duffy}) = \text{Stephen Duffy}$$
$$e^+(\text{John Taylor}) = \text{John Taylor (bass guitarist)}$$

First, note that each entity in Wikipedia is uniquely identified through its corresponding article title. This title is used to form the article's URL and thus also provides an interface to Wikipedia's derivatives DBpedia and YAGO. Typically, the title of an article is the most common name of the described entity. Since Wikipedia covers a substantial amount of entities with identical names, *disambiguations* terms are used to prevent name collisions and distinguish among entities with identical names. These terms are often entity specific key phrases that are appended to the name of the entity and may for instance denote a profession (e.g. Jason Taylor (bass guitarist)) or an administrative district (e.g. Berlin, Wisconsin). Thus, entity titles can be considered as rigid designators since names shared by more than one entity in Wikipedia are rendered unique through the concatenation of disambiguation terms. Often, when a prominent entity with an ambiguous name exists, qualifiers are added only to the names of the less well known entities. We use the

following notation to distinguish among an entity's *title*, which is its unique identifier, and its *name* in Wikipedia, which is the title without artificial disambiguation term.

Notation (Title, Name)
For any entity $e \in \mathcal{W}$ we use $title(e)$ to denote its unique title. We use $name(e)$ to denote the name of the entity, which is the title without qualifying term.

Example 1 also illustrates the necessity to resolve synonymy and polysemy: Bates and Nick Rhodes are both synonyms for the artist Nicholas James Bates and many other persons apart from the guitarist are named John Taylor. The mention Bates refers to the surname of NICK RHODES' birth name Nicholas James Bates. We find other candidate entities for this mention in KATHY BATES and NORMAN BATES since the titles of these articles also contain the surface form. In this example, the birth name Nicholas James Bates is a redirect (we will detail redirects in the next section) for the alias NICK RHODES, which was chosen as article title presumably due to its more prominent usage. A naive matching based solely on character overlap between mention and title would not return NICK RHODES as candidate entity for the mention Bates. Instead, it would prefer to return the fictional character NORMAN BATES or the actress KATHY BATES. Note that maximum similarity is achieved returning BATES (AUTOMOBILE). Thus, in this and many other cases, the correct entity can not be retrieved if we consider only candidates whose title matches the surface form of a mention. In many cases, such a string-based similarity approach may fail to retrieve the true underlying entity as a candidate. Furthermore, simple string matching may also result in an incorrect prediction as shown by the following example.

Example 2
The following sentence gives an example of the mention Tom Sharpe that refers to an uncovered entity.

> The Gardner-Webb University will present a unique concert featuring world-famous percussionist Tom Sharpe.

The mention Tom Sharpe in the sentence above does not refer to the writer TOM SHARPE, who is covered in Wikipedia, but instead to a musician, who is not covered in Wikipedia. Thus, the ground truth target for this mention is $e^+(\text{Tom Sharpe}) = \text{NIL}$.

A naive string-matching might link the mention in the example above to the writer. While the writer is a valid candidate in the example above, the mention must not be linked to him but to the representative NIL. There are different avenues

to achieve this. For instance, Bunescu and Pasca [2006] add the NIL entity as a dedicated candidate, e.g.

$$\mathbf{e}(\mathsf{Tom\ Sharpe}) = \{\textsc{Tom Sharpe}\} \cup \{\text{NIL}\}.$$

This allows Bunescu and Pasca [2006] to automatically learn thresholds for NIL predictions based on the weights of indicative features. Ratinov et al. [2011] first learn a model to rank all candidates and then use the predictions of this model in a second model to decide on one specific candidate which may also be NIL. In Pilz and Paaß [2011] we showed that it can also be effective to use the threshold induced by the decision boundary of a binary Support Vector Machine classifier. Generally, entity linking should aim for models that need not be fine-tuned through manual threshold adaption and that automatically choose the best candidate, either by assigning a mention to an entity in Wikipedia, i.e. to $e \in \boldsymbol{\mathcal{W}}$, or assigning it to NIL $\notin \boldsymbol{\mathcal{W}}$ stating that this mention is not covered.

The last two examples and the following discussion show the need for two key components of entity linking: *candidate consolidation* and *candidate retrieval*.

2.2.1 Candidate Consolidation and Candidate Retrieval

The candidate consolidation part of a linking model selects one specific candidate as the target entity. Predicting the correct underlying entity of a mention, based on some linking model, is the key to a high linking precision. This also means that we must not link an uncovered entity mention to an entity in Wikipedia even if a string-match between the surface form of the mention and the name of the entity indicates a perfect match.

We assume in this thesis that a mention must not be linked to more than one entity, i.e. either to exactly one entity $e_i \in \boldsymbol{\mathcal{W}}$, or to the representative of uncovered entities NIL. We learn our linking models from examples of mentions that are grounded to no more than one entity and enforce the decision to link a mention m to at most one unique entity e. A more general definition of entity linking may also allow a result set containing more than one entity. For instance, aiming at aggressive linkage, Kulkarni et al. [2009] also used Wikipedia disambiguation pages as ground truth annotations. While technically the predicted link is then also a unique title in Wikipedia, it is not the description of one entity but indeed a listing of potential candidates. In this thesis, we argue that resolving the ambiguity of a mention means that we ground it to at most one entity which is the unique and unambiguous link target. Our goal is to retrieve only one entity per mention so that no further manual decision needs to be made.

The candidate retrieval part is most influential for high linking recall. This part has the purpose to retrieve all relevant candidate entities among which the consolidation part needs to decide. Wikipedia provides various means to create an elaborate

candidate retrieval model. These will be described next, together with the other main attributes of entities in Wikipedia, i.e. article texts and categories.

2.3 Entities in Wikipedia

2.3.1 Textual Descriptions

In Wikipedia, each entity has a natural language context in its article text. Article texts provide textual descriptions of entities that can be used to assess the contextual similarity of mentions and entities.

Notation (Article text)
For any entity $e \in \boldsymbol{\mathcal{W}}$ we use $text(e)$ to denote the entity's context which is derived from the respective Wikipedia article text.

By Wikipedia standards, an article is supposed to describe its entity in a concise but comprehensive way. In the following we consider the Wikipedia article text, i.e. the plain text without markup, tables, infoboxes or figures, as a natural language text definition of the described entity. Analogously, the document referencing a mention provides the (natural) language context $text(m)$ of a mention m.

Notation (Mention context)
For a mention m we use $text(m)$ to denote its context which is derived from the document in which the mention appears.

In general, we assume a context $text(m)$ to comprise all words surrounding a mention m, meaning either the complete document or a restricted, localized context window, for example five words left and right of the mention. This context is assumed to disambiguate the mention so that its true underlying entity can be inferred. Note that the natural language text in Wikipedia is comparable to the natural language context of a mention, assuming overlap in the underlying vocabularies. This allows us to formulate entity linking based on a similarity function over the two contexts $text(m)$ and $text(e)$. The most prominent contextual similarity measure is cosine similarity that compares the two word-vectors of entity and mention context. We will give further details in Chapter 3 where we also propose new contextual measures for entity linking.

2.3.2 Entity Categorization

To group articles on similar subjects, Wikipedia employs a categorization system. Below the top-level categories distinguishing persons from cultural or economical entities, many other categories exist that further describe the entity depicted in an

Table 2.1: Entities and a selection of their assigned categories from the English Wikipedia (distinct categories are separated by a semicolon).

title(e)	categories $\boldsymbol{c}(e)$
John Taylor (bass guitarist)	Living people; English rock bass guitarists; Power Station (band) members; Duran Duran members; English Roman Catholics; Ivor Novello Award winners; ...
John Taylor (jazz)	Living people; Post-bop pianists; ECM artists; Musicians from Manchester; British jazz pianists; ...
John Taylor (athlete)	American sprinters; Athletes (track and field) at the 1908 summer Olympics; Olympic medallists in athletics (track and field); Olympic track and field athletes of the united states

article. Categories may be thematically related to the article content but also state the gender of a person or the founding year of an organization.

By Wikipedia standards, every article is required to have at least one category that is manually assigned by a contributor using Wikipedia markup language. We use the following notation to refer to the categories of an entity.

Notation (Categories)
We denote the collection of all Wikipedia categories by $\mathbf{C}_{\boldsymbol{\mathcal{W}}} = \{c_1, \ldots, c_{|\mathbf{C}_{\boldsymbol{\mathcal{W}}}|}\}$. The subset of categories applying to a specific entity $e \in \boldsymbol{\mathcal{W}}$ is denoted by $\boldsymbol{c}(e) = \{c_1(e), \ldots, c_{|\boldsymbol{c}(e)|}(e)\} \subset \mathbf{C}_{\boldsymbol{\mathcal{W}}}$.

Tab. 2.1 lists some exemplary categories from the English Wikipedia. For example, the categories assigned to John Taylor(bass guitarist) depict his profession as musician and the genre of music, i.e. rock, he is involved with. While grouping this entity with the musician John Taylor (jazz) on a higher level, more specific categories distinguish the rock guitarist from the jazz pianist (e.g. *English rock bass guitarists* and *British jazz pianists*).

Originally a tree, the Wikipedia category system has evolved to graph with many interconnections and loops. Due to these loops and also other inconsistencies, we found the analysis of Wikipedia's category system non-trivial in preliminary studies. Moreover, even though there exist guidelines on categorization[1], Wikipedia categories can be very general but also overly specific. Rather general categories such as *Living People* apply to very many entities, overly specific categories such as *Fictional elephants* apply to only very few entities.

As categories group entities by subject, they can be used to measure semantic relatedness among entities and also to extend contextual information. The semantic relatedness expressed by categories (Strube and Ponzetto [2006]) has also been

[1] `http://en.wikipedia.org/wiki/Wikipedia:Categorization`

Table 2.2: Examples of Wikipedia titles and associated redirects (distinct redirects are separated by a semicolon).

$title(e)$	$\boldsymbol{r}(e)$
Nick Rhodes	Nicholas James Bates
Stephen Duffy	Steven Tin Tin Duffy; Stephen TinTin Duffy; Stephen 'Tin Tin' Duffy; Stephen Tin Tin Duffy; Duffy (group)
John Taylor (bass guitarist)	John Taylor (Duran Duran); Nigel John Taylor

exploited in entity linking approaches. These approaches usually do not consider all available categories but use, often manually, selected subsets, either to avoid noise or to emphasize semantic relatedness in specific subgroups. For example, Cucerzan [2007] used filtered subsets for a named entity disambiguation model, Bunescu and Pasca [2006] used the specific branch *People by Occupation* for their person name disambiguation model.

2.3.3 Alias Names for Entities

Wikipedia provides several means to collect alias names for its entities by which the synonymy and polysemy of entity names can be resolved. The first important means are *redirects* that can be used to collect alternative names for an entity and thus account for the synonymy of entity names. A redirect is a meta-page that contains only a forwarding link to an actual entity. The title of a redirect page is considered as an alias of the target entity the redirect page points to. An entity in Wikipedia may have several redirects, one for every alternative name that a Wikipedia contributor used to refer to it.

Notation (Redirects)
For any entity $e \in \boldsymbol{\mathcal{W}}$, we use $\boldsymbol{r}(e) = \{r_1(e), \ldots, r_r(e)\}$ to denote the collection of titles that redirect to e.

Tab. 2.2 lists examples of entity titles and their associated redirects. For instance, redirects may hold the full name of a person (e.g. Nicholas James Bates for Nick Rhodes), cover nickname variants (e.g.Stephen 'Tin Tin' Duffy for Stephen Duffy) or provide more name variants.

Redirects provide a large resource of synonyms and have been exploited extensively in the literature. However, seldom considered is the fact that redirects can also be misleading since they do not necessarily compose equivalence relations. For instance, the German chancellor Angela Merkel has a redirect Ulrich Merkel. This is not an identity relation as Ulrich Merkel is a different person, namely the

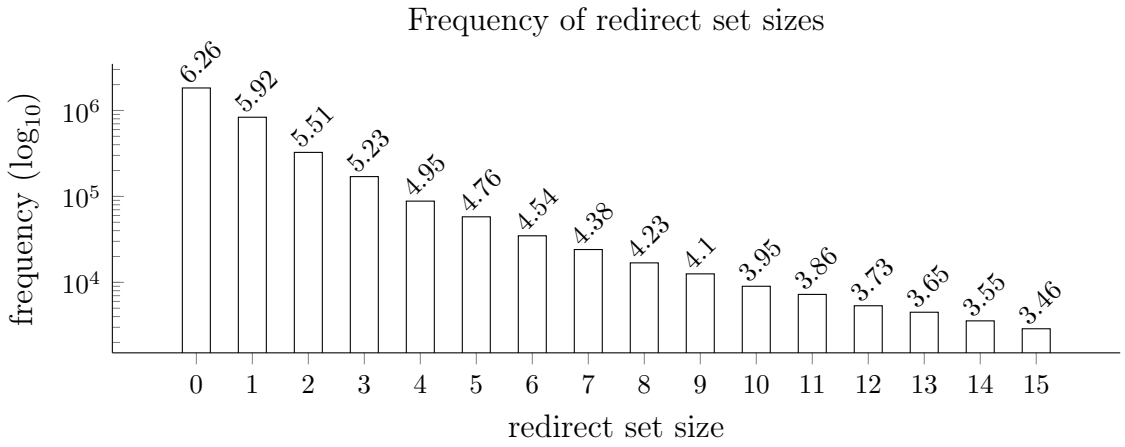


Figure 2.3: To measure synonymy in terms of redirects, the figure shows the cardinality of redirect set sizes by frequency. This is clamped to set sizes between 0 and 15 due to the long tail of entities with up to 900 distinct redirects. The number of distinct redirects of a Wikipedia entity may be used as an approximation on the number of its synonyms.

first husband of Angela Merkel. Thus, the usage of redirects may introduce errors in the disambiguation process. In the worst case scenario, we would link an uncovered mention to an entity in Wikipedia because of an erroneous redirect mapping in this alias dictionary. Alternatively, we may erroneously link a mention to a merely related entity, as would be the case for Ulrich Merkel. But then, assuming that the creation of such a redirect was somehow intentional, we argue that the predicted entity may at least provide useful information both for the reader of the linked context or other link consuming systems.

In general, we assume such errors to be rare and furthermore, a better defined redirect scheme would already require a disambiguation step. Therefore we consider all redirects *as is* without pre-processing and create alias dictionaries that map all redirects of an entity to its unique title (and vice versa).

Notably, using redirects, we can also estimate the number of synonyms for an entity. As they provide name alternatives, redirects are comparable to synonyms and the number of redirects of an entity may serve as an approximation of the true number of potential synonyms of this entity. To illustrate synonymy in terms of redirect numbers, we counted the number of redirects for all articles in Wikipedia and depict in Fig. 2.3 how often a specific number appears. For visualization purposes, this is restricted to redirect set sizes between zero and 15 as there is a long tail of entities with more than 100 and up to 900 distinct redirects. From the figure we see that about 23% of the 3.4 million entities in the English Wikipedia have at least two redirects. This finding has two implications. First, a rather large number of entities can be assumed to have several synonyms and thus a high variation in

their references needs to be resolved through entity linking. Second, the absence of redirects may also imply that about 77% of the articles are more or less consistently referenced by their most common name, at least in Wikipedia.

Additionally to redirects, Wikipedia provides *disambiguation pages* as a means to handle polysemy. Disambiguation pages are manually created lists of entities that may be referenced by the same name where the name is indicated by the title of the disambiguation page. As an example, the English disambiguation page for the surname Kohl lists eight distinct entities. The German version lists already more than 50 persons plus 13 additional references to persons that are as of yet uncovered.[1]

Disambiguation pages are often used to enrich alias dictionaries (Bunescu and Pasca [2006], Mihalcea and Csomai [2007], Cucerzan [2007], Hoffart et al. [2011b], Varma et al. [2009] and others). In this thesis, we do not use them as we found these listings difficult to parse and also often inconsistent or incomplete.

To summarize, comprehensive alias dictionaries are crucial for entity linking as they heavily affect candidate retrieval which again influences the recall of the linking model. The number of retrievable candidates is an upper bound on the linking performance and thus most approaches use carefully designed candidate retrieval methods. Even though it can be sufficient that the title of an entity matches the mention name on a character level, in many cases, however, more elaborate candidate selection methods are required. These should take into account synonyms, abbreviations and other name variations such as spelling mistakes. Thus, redirects are a valuable resource since they provide additional aliases that need not be derivable from the article of an entity. Aliases can also be used as baselines in context free entity linking. Such baselines "predict" entities by comparing the surface form of a mention against their titles or redirects and can be particularly effective for unambiguous mentions (Hachey et al. [2013], Ratinov et al. [2011]). Alias baselines can also be useful to measure the influence of more elaborate methods using contextual or relational information but also to assess the average ambiguity of the mentions in a given corpus.

The literature in entity linking exploits Wikipedia aliases to different extents. Most approaches use titles, names, redirects and disambiguation page titles. The main difference lays in the incorporation of link anchor texts into the alias dictionary. This was first proposed by Cucerzan [2007] and subsequently used in many other approaches. Link anchor texts are derived from Wikipedia's hyperlink graph which will be described in next.

2.4 Interlinkage of Entities in Wikipedia

Links in Wikipedia are supposed to provide further details related to the subject described in an article. When mentioning an entity e' with an existing article page

[1] Figures retrieved from the respective language versions of Wikipedia in November, 2013.

in the article text of another entity e, contributing authors are expected to link at least the first mention to the corresponding article of e'.

More specifically, a link l in Wikipedia is a triple of *link source*, *link anchor text* and *link target*.

Notation (Links in Wikipedia)
Let $\mathbf{L}$ denote the collection of all links in $\boldsymbol{\mathcal{W}}$. A link $l \in \mathbf{L}$ is a triple

$$l = (l_s, l_t, l_a),$$

where l_s denotes the link source, l_t the link target and l_a the link anchor text.

Link sources and link targets are entities in Wikipedia and thus we have $l_s \in \boldsymbol{\mathcal{W}}$ and $l_t \in \boldsymbol{\mathcal{W}}$. Link anchor texts are part of an entity's article text and formed of one or more strings from the Wikipedia vocabulary $V_{\boldsymbol{\mathcal{W}}}$, i.e $l_a \in V_{\boldsymbol{\mathcal{W}}} \times \ldots \times V_{\boldsymbol{\mathcal{W}}}$. A link l is placed in the article text of its link source l_s using Wikipedia markup notation. This notation couples link anchor text and link target, i.e. $[[l_t \mid l_a]]$. Alternatively, it may merely hold the link target, i.e. $[[l_t]]$, if link anchor text and link target do not differ, i.e. $l_a = title(e)$.

The collection of all links $\mathbf{L}$ in Wikipedia encodes a directed hyperlink graph, where nodes are entities and edges the links between them. Using the following notation, we distinguish among *inlinks* and *outlinks*.

Notation (Inlinks and Outlinks)
The *inlinks* $\mathbf{L}_{in}(e)$ of an entity e are the links where e is the link target l_t:

$$\mathbf{L}_{in}(e) = \{l \in \mathbf{L} \mid (l_s = \cdot, l_t = e, l_a = \cdot)\} \subset \mathbf{L}.$$

The *outlinks* $\mathbf{L}_{out}(e)$ of an entity e are the links where e is the source l_s:

$$\mathbf{L}_{out}(e) = \{l \in \mathbf{L} \mid (l_s = e, l_t = \cdot, l_a = \cdot)\} \subset \mathbf{L}.$$

For illustration, we give the following example using the links depicted in Fig. 2.4.

Example 3
As Fig. 2.4 shows, the article text of Duran Duran contains the link l^1 as outlink, i.e. $l^1 \in \mathbf{L}_{out}$(Duran Duran). This link is described by

$$\begin{aligned} \text{link source: } l_s &= \text{Duran Duran} \\ \text{link anchor text: } l_a &= \text{John Taylor} \\ \text{link target: } l_t &= \text{John Taylor (bass guitarist)}. \end{aligned}$$

At the same time, the link l^1 is an inlink of the link target and we have $l_1 \in \mathbf{L}_{in}$(John Taylor (bass guitarist)). Here, the link target is obfuscated by Wikipedia's markup notation, i.e.

$$[[\text{John Taylor (bass guitarist)} \mid \text{John Taylor}]].$$

text(DURAN DURAN)

John Taylor and Nick Rhodes formed Duran Duran in Birmingham, England in 1978, where they became the resident band at the city's Rum Runner nightclub.

$l^1, \ldots, l^4 \in \mathbf{L}_{out}$(DURAN DURAN)

$l^1 = (l_s = \text{DURAN DURAN}, l_t = \text{JOHN TAYLOR (BASS GUITARIST)}, l_a = \text{John Taylor})$
$l^2 = (l_s = \text{DURAN DURAN}, l_t = \text{NICK RHODES}, l_a = \text{Nick Rhodes})$
$l^3 = (l_s = \text{DURAN DURAN}, l_t = \text{BIRMINGHAM}, l_a = \text{Birmingham})$
$l^4 = (l_s = \text{DURAN DURAN}, l_t = \text{RUM RUNNER (NIGHTCLUB)}, l_a = \text{Rum Runner})$

Figure 2.4: Excerpt from the article text of the entity DURAN DURAN (top) together with the contained links (bottom). Each link $l^1, \ldots, l^4$ is contained in the outlink set $\mathbf{L}_{out}$(DURAN DURAN) and we have $l_s =$ DURAN DURAN for each $l^1, \ldots, l^4$.

The example above shows two properties of links. First, the link anchor text l_a can be considered as an *alias* for the target entity that is referenced through the link target l_t. Second, a link constitutes a disambiguated mention m. The link l^1 in this example provides a grounded mention of the entity JOHN TAYLOR (BASS GUITARIST) as the underlying entity for the ambiguous mention John Taylor is annotated in the link target.

This property renders Wikipedia a source of disambiguated data. Each inlink $l \in \mathbf{L}_{in}(e)$ constitutes a textual mention of the entity e in the article text of the link source e'. Through the link target l_t, the link anchor text l_a is annotated with the ground truth entity, i.e. $e^+(l_a) = l_t = e$. For each Wikipedia entity e with at least one inlink, we can extract the textual contents of all referencing articles in $\mathbf{L}_{in}(e)$. The derived dataset can then be used both for the training as well as the evaluation of a disambiguation model, as done first by Bunescu and Pasca [2006] and subsequently in many other approaches (Pilz et al. [2009], Pilz and Paaß [2009], Pilz [2010], Ratinov et al. [2011]). This procedure allows the learning of entity linking models in a supervised setting not only for English but all other language versions of Wikipedia. For instance in this thesis, we learn linking models for German and French, two languages previously neglected in the entity linking literature (Chapter 3).

Furthermore, we may use the same data base to learn a linking model that accounts for uncovered entities. One possible avenue for this is using the links that point to a target that does not yet exists in Wikipedia[1]. However, such links are rare and usually appear in listings with only few natural language context such as disambiguation pages. They may also result from a faulty annotation, for instance when an author does not realize that the relevant article already exists and erroneously chooses a different link target.

[1]The Wikipedia software colours such links in red.

A presumably better alternative was proposed in Bunescu and Pasca [2006]. The authors simulate uncovered mentions by removing the article for a fixed fraction of entities. All mentions previously linked to these entities are then re-assigned to NIL. We follow this strategy when learning models with Wikipedia data and will detail the extraction of such datasets in Chapter 3.

2.4.1 Link-Based Relatedness of Entities

A link in Wikipedia implies a semantic relation between the two Wikipedia entities it connects. Therefore, Wikipedia's hyperlink graph allows the derivation of relatedness measures and is thus a key component in most research on entity linking. The most commonly used measure was introduced by Milne and Witten [2008a], who presented a Wikipedia adaption of the normalized Google distance (Cilibrasi and Vitanyi [2007]). Replacing Google search results with Wikipedia links, Milne and Witten [2008a] defined the *semantic relatedness* (SRL) of two entities e and e' over their inlink sets $\mathbf{L}_{in}(e)$ and $\mathbf{L}_{in}(e')$:

$$\mathrm{SRL}(e, e') = \frac{\log\left(\max(|\mathbf{L}_{in}(e)|, |\mathbf{L}_{in}(e')|)\right) - \log\left(|\mathbf{L}_{in}(e) \cap \mathbf{L}_{in}(e')|\right)}{\log(|\mathbf{L}|) - \log\left(\min(|\mathbf{L}_{in}(e)|, |\mathbf{L}_{in}(e')|)\right)} \in [0, 1] \quad (2.2)$$

Note that the range of [0,1] given above only holds if we to take into account edge cases that are not explicitly covered in Milne and Witten [2008a]. These arise when an entity has no inlinks or the shared set of inlinks is empty. Thus, we define

$$\mathbf{L}_{in}(e) = \emptyset \vee \mathbf{L}_{in}(e') = \emptyset \vee \mathbf{L}_{in}(e) \cap \mathbf{L}_{in}(e') = \emptyset \rightarrow \mathrm{SRL}(e, e') := 1. \quad (2.3)$$

With the above definition, SRL may take values in the interval $[0, 1]$. Low values of SRL are realized for similar sets of inlinks and high values for dissimilar inlink sets. The lower bound is obtained only if the sets $\mathbf{L}_{in}(e)$ and $\mathbf{L}_{in}(e')$ are identical. The upper bound could only be obtained if all links in Wikipedia would be targeted towards one entity, an extremely unlikely edge case. Thus, while Milne and Witten [2008a] somehow counter-intuitively termed SRL a semantic relatedness measure, we argue that this behaviour is that of a dissimilarity measure. Since in this thesis we will use SRL as a similarity measure, we define for all implementations of SRL

$$\mathrm{SRL}^* := 1 - \mathrm{SRL} \quad (2.4)$$

where SRL is based on the definition in Eq. 2.2 and the adaption in Eq. 2.3. Then, $\mathrm{SRL}^*(e_i, e_j) = 0$ implies unrelated entities while $\mathrm{SRL}^*(e_i, e_j) = 1$ states that two entities have identical inlink targets. Still, the upper bound is not likely to be obtained due to the magnitude of $\mathbf{L}$ in Eq. 2.2. Ratinov et al. [2011] also evaluated semantic relatedness over outlinks sets $\mathbf{L}_{out}(e)$ and $\mathbf{L}_{out}(e')$. Analogously to SRL,

the operating figure is given by:

$$\mathrm{SRL}_{\mathrm{out}}(e, e') = \frac{\log\left(\max(|\mathbf{L}_{out}(e)|, |\mathbf{L}_{out}(e')|)\right) - \log\left(|\mathbf{L}_{out}(e) \cap \mathbf{L}_{out}(e')|\right)}{\log(|\mathbf{L}|) - \log\left(\min(|\mathbf{L}_{out}(e)|, |\mathbf{L}_{out}(e')|)\right)} \in [0, 1]. \tag{2.5}$$

Again, to ensure a range of [0,1] we need to account for edge cases and analogously to Eq. 2.3 define

$$\mathbf{L}_{out}(e) = \emptyset \vee \mathbf{L}_{out}(e') = \emptyset \vee \mathbf{L}_{out}(e) \cap \mathbf{L}_{out}(e') = \emptyset \rightarrow \mathrm{SRL}_{\mathrm{out}}(e, e') := 1 \tag{2.6}$$

to arrive at $\mathrm{SRL}_{\mathrm{out}} \in [0, 1]$. If not otherwise stated, we use SRL^* to refer to the measure computed over inlinks. The semantic relatedness measure over outlinks is denoted with $\mathrm{SRL}_{\mathrm{out}}$.

Semantic relatedness computed over shared inlinks is basically a measure for the co-occurrence of Wikipedia entities. From this co-occurrence we may derive coherence among entities and for instance conclude that a document jointly mentioning Michael Jordan and NBA is more likely to refer to the basketball player and the basketball association instead of the machine learning professor and the boxing association.

2.4.2 Priors derived from Links

Wikipedia's hyperlink graph also allows the derivation of priors. Ratinov et al. [2011] formulate *entity-mention probability* (EMP) as the prior probability that a link anchor text m refers to an entity e by analysing all pairs of link target and link anchor text in Wikipedia. Then, entity-mention probability is the ratio of times an entity e is the target l_t for a link anchor text $l_a = m$ to the overall number of targets referenced by m:

$$\begin{aligned} p(e|m) &= p(l_t = e|l_a = m) \\ &\approx \frac{|\{l \in \mathbf{L}|l = (l_s = \cdot, l_t = e, l_a = m)\}|}{\sum_{e' \in \boldsymbol{\mathcal{W}}} |\{l \in \mathbf{L}|l = (l_s = \cdot, l_t = e', l_a = m)\}|}. \end{aligned} \tag{2.7}$$

The numerator in Eq. 2.7 is the absolute frequency of e being the target $l_t = e$ of l with anchor text $l_a = m$ and the denominator is the sum over all possible entities $e' \in \boldsymbol{\mathcal{W}}$ that have been referenced by m through a link anchor text l_a. While the above formulation results theoretically in a true probability with range $[0, 1]$, we point out that in practice we may observe that $\sum_m p(e|m) \neq 1$. Due to parsing errors, erroneous links or other pitfalls, we need to interpret EMP as an approximation.

The following example illustrates EMP with values computed using the link index proposed in Pilz and Paaß [2012] (this link index will be thoroughly in Section 4.3.2).

Example 4 (Entity-Mention Probability (EMP))
Based upon Eq. 2.7, we obtain entity-mention probabilities such as:

$$p(e = \text{WASHINGTON, D.C.} \mid m = \mathsf{Washington}) \approx 0.9$$
$$p(e = \text{WASHINGTON, D.C.} \mid m = \mathsf{D.C.}) \approx 0.2$$
$$p(e = \text{WASHINGTON (STATE)} \mid m = \mathsf{Washington}) \approx 0.1$$

Note that EMP is a value that is not stored in the article text itself and can only be extracted from a knowledge base similar to Wikipedia that provides this information through its internal link structure. It was found to be a proven feature for disambiguation in many approaches (e.g. Milne and Witten [2008b], Fader et al. [2009], Ratinov et al. [2011], Hoffart et al. [2011b], Pilz and Paaß [2012]). However, note that this formulation of EMP puts a bias towards Wikipedia entries: any mention of an uncovered entity that has a surface form matching a prominent Wikipedia entity is likely to be assigned to this entity when no additional information is used.

Another figure derivable from the link graph is the *popularity prior* of an entity in Wikipedia. The popularity prior of an entity e is the ratio of articles linking to e and the total number of links in Wikipedia:

$$p(e) \approx \frac{|\mathbf{L}_{in}(e)|}{|\mathbf{L}|}. \tag{2.8}$$

This measure stands in analogy to the in-degree of a node in a graph but is normalized through the number of all links in Wikipedia. While defined over Wikipedia links, it may also serve as a prior for the popularity of an entity in other contexts assuming that entities often interlinked in Wikipedia are also frequently mentioned for instance in news articles.

The popularity prior has been successfully used as a baseline attribute for entity linking (Ratinov et al. [2011]). However, especially in the English version of Wikipedia, the overall number of links is with 54 million very large[1]. Therefore the popularity prior for most entities is very small or close to zero and only a handful of entities have priors greater than a few per mill. For instance, the highest popularity prior we observed in the context of this thesis was 0.006 for the entity UNITED STATES. In Pilz and Paaß [2012] we therefore proposed to use the more effective absolute value of $|\mathbf{L}_{in}(e)|$ without normalization factor. In Chapter 4, we will give more details and show how we use this prior for an adaptive filtering of mention candidates.

[1]Version from September 1st, 2011.

2.5 An Overview of Related Work

To give a general overview, this section summarizes the major differences among recent approaches in entity linking. We will give more details on the most important related methods in the subsequent chapters. Since entity linking has attracted a lot of attention in recent years, there has also been a major amount of publications. To keep focus in comparison with related work, we concentrate on approaches that link entity mentions in natural language text to Wikipedia or one of its derivatives. The major differences of recent approaches to entity linking can be characterized by the nature of the handled entities and the individual or joint linkage of mentions.

2.5.1 Focus on Mention and Entity Types

Research in **named entity linking** is focussed on linking mentions that denote named entities, i.e. persons, locations or organizations. These are the most common named entity types that can be assigned to a mention by standard NER models. The most prominent works in named entity linking are those of Cucerzan [2007], Dredze et al. [2010], Hoffart et al. [2011b] and Shen et al. [2012]. Other approaches focus on a specific named entity type such as persons (Bunescu and Pasca [2006], Han and Zhao [2009]), locations (Pouliquen et al. [2006], Volz et al. [2007]) or cultural entities (Gruhl et al. [2009]).

NER models may have limited predictive performance. For instance, a mention like Alice Springs is challenging as it may denote some person or the Australian town. Since Alice is a common female surname, it is likely that a NER model erroneously predicts the type person even if the mention indeed refers to the town. Creating linking models that strongly depend on these predictions, perhaps even in hard coded decision rules, can thus be harmful for the linking performance. But while erroneous type assignments can be handled by a good linking model, mentions that are not detected by a NER model will not be handled by the linking model. Thus, it maybe even more important that NER with sufficient performance is available only for a minority of languages, most prominently English. This can limit the applicability of name entity linking to certain languages. The major focus of Chapter 3 is in linking person names, especially ambiguous ones. However, the method generalizes to some extent also to other types of entities and yields state-of-the-art results in less well studied languages such as German and French.

More general **entity linking** approaches aim at linking all kinds of terms or phrases without type restrictions. The mentions to be linked can be extracted using any kind of phrase extraction or text segmentation method. General entity linking overlaps with word sense disambiguation methods, but does usually not handle adjectives or verbs. For entity linking, the most prominent methods were presented in Mihalcea and Csomai [2007], Milne and Witten [2008b], Kulkarni et al. [2009], Mendes et al. [2011], Han et al. [2011]. They aim at linking all interesting mentions

such as key terms appearing in a document (Mihalcea and Csomai [2007], Milne and Witten [2008b]) or all phrases or terms for which a link can be generated (Kulkarni et al. [2009], Han et al. [2011]).

The above distinction concerns the entities in Wikipedia that are potential candidates for a mention. Another distinction of current work in entity linking lies in the handling of uncovered entities. While some other approaches ignore such entities (for instance Hoffart et al. [2011b]), we include them both in model design and evaluation. Apart from the handling of uncovered entities, the type focus of a method does usually not effect model design. However, the design depends on whether an approach is local or global. This distinction will be depicted next.

2.5.2 Local and Global Methods

Entity linking methods can be characterised as *local* or *global* methods or combinations of both. Local methods link each mention individually and use local, mention specific attributes. They often measure the compatibility of mention and entity through the textual similarity between mention context and entity description (Bunescu and Pasca [2006], Mihalcea and Csomai [2007], Dredze et al. [2010]). This was also the avenue we pursued in Pilz and Paaß [2011]. The approach of Bagga and Baldwin [1998] was also local as they considered only one ambiguous name mention in their documents. Starting with simple bag-of-word descriptions more advanced features were developed to characterize the sense of context words and measure the contextual similarity between mention context and entity definition. Among those are the correlation of context features with Wikipedia categories (Bunescu and Pasca [2006]) and the generalization to thematic context similarity using topic models (Pilz and Paaß [2011]). We will detail the contributions from Pilz and Paaß [2011] in Chapter 3 where we compare the proposed methods to the seminal work of Bunescu and Pasca [2006].

One drawback of local methods can be that they do not explicitly make use of higher level semantic relations expressed only implicitly in documents and therefore may miss important relational information. *Global* or *document level* approaches try to fill this gap. Either relying on the assumption that there is more than one mention in the document to be linked, or generating additional ones (Ratinov et al. [2011]), these approaches aim at the simultaneous disambiguation of all mentions in a document. Local attributes such as contextual similarity are then combined with a document level coherence computed among all candidate entities for all mentions in a document. The most prominent method uses Wikipedia's link graph to estimate pairwise semantic relatedness over shared in- or outlinks (cf. Section 2.4.1) among candidates (Milne and Witten [2008b], Kulkarni et al. [2009], Ratinov et al. [2011], Han and Zhao [2009]). Other approaches exploit the semantic relatedness in categories (Cucerzan [2007]).

However, estimating pairwise relatedness among candidates is computationally

expensive as all candidate entities need to be related and finding the optimal set is NP-Hard (Ratinov et al. [2011], Kulkarni et al. [2009]). One line of research is therefore to approximate the optimal set by assuming that the collection of ground truth entities forms a *coherent* set which is supposed to eliminate irrelevant candidate entities (Cucerzan [2007], Milne and Witten [2008b], Ratinov et al. [2011], Kulkarni et al. [2009], Han and Zhao [2009]). However, this does not indicate a truly collective or joint optimization. Global approaches are computationally more expensive and most useful when the number of mentions in an input document exceeds a certain level. Therefore, the combination of local and global methods is currently the most promising line of research. This is the avenue we pursued in Pilz and Paaß [2012]. While still linking each mention individually, we use a global search over all links in Wikipedia to arrive at the most coherent candidate set. We will detail the contributions from Pilz and Paaß [2012] in Chapter 4.

2.5.3 Types of Linking Models

The literature has proposed sundry structures for entity linking models. Some linking models explicitly incorporate collective disambiguation into their structure to arrive at an approximate joint linkage, for instance in Markov Logic Networks (Fahrni and Strube [2012]) and many other generative approaches over graphical models (Kulkarni et al. [2009], Han et al. [2011], Han and Sun [2012]). Others use discriminative classifiers such as decision trees (Milne and Witten [2008b]), Naive Bayes (Mihalcea and Csomai [2007]), Support Vector Machines (Pilz and Paaß [2011]), Ranking Support Vector Machines (Bunescu and Pasca [2006], Pilz and Paaß [2012]) or combinations of both (Ratinov et al. [2011]). In discriminative methods, relational attributes can be used as distinct features that are computed individually per mention but taking into account document-level information (Ratinov et al. [2011], Pilz and Paaß [2012]) or directly on the document-level to maximize the collective agreement among candidates (Cucerzan [2007]).

To summarize, most approaches rely on supervised machine learning models. Wikipedia provides not only a collection of ground truth target entities for linking models but simultaneously also disambiguated example references for these entities which facilitates the training and evaluation of supervised models without additional annotation costs. Therefore, the majority of recent research investigates supervised methods whereas earlier name discrimination methods explored unsupervised clustering methods. In the main contributions of this thesis, we follow this approach. In Pilz and Paaß [2011] we describe a classification scenario and use a Support Vector Machine to classify a candidate entity as correct or incorrect. In Pilz and Paaß [2012] we create a ranking scenario and use a Ranking Support Vector Machine to detect the best matching candidate relative to its rank towards other candidates.

2.5.4 Alternative Resources for Entity Linking

In this thesis, we make use of the fact that Wikipedia provides information about well and less well known entities in a multitude of languages. In the preceding sections we have shown that Wikipedia provides semi structured information for each of these entities, rendering it superior to a simple entity catalogue with mere listings of entity names. Therefore, the only true alternative resources are Wikipedia's derivatives YAGO (Suchanek et al. [2008]) and DBpedia (Bizer et al. [2009]). Another potential resource for entity linking is Freebase[1], a structured knowledge base build collaboratively by contributors from various sources. Freebase covers notably more topics than Wikipedia (approximately 47 million topics and 2.7 billion facts according to the website[2]) but does not provide context for most of these entities. It lacks entity descriptions and also the very important hyperlinks that Wikipedia provides (Zheng et al. [2012]). Therefore, Freebase is here not considered a true alternative but mentioned for the sake of completeness.

Also, we decided against YAGO and DBPedia for the following reasons. YAGO is an ontology that joins Wikipedia and WordNet (Miller [1995]) by endowing Wikipedia articles with WordNet attributes. However, at the time of publication, YAGO covered only the English Wikipedia, a multilingual version has been announced to appear in 2015. The Linked Open Data hub DBpedia is a structured database created over Wikipedia. DBpedia has only recently been adapted for German, but also for the English version it does not contain the full article texts but only (extended) abstracts. Using these two resources would thus more or less have limited this thesis to the investigation of linking models in English. To overcome this, we directly rely on the original sources extracted from Wikipedia dumps. This necessitates on one hand from the lack of disambiguated corpora for the training and evaluation of entity linking models in languages other than English and, on the other hand, from the fact that the link structure along with the link anchor texts and their contexts can, at least currently, only be extracted from Wikipedia itself.

Summary

In this chapter we have introduced the basic concepts and the notation used in this thesis. We have defined the task of linking entity mentions in natural language text to unique entities in the encyclopedia Wikipedia, motivated the usage of Wikipedia and described the main components that are used for the entity linking models proposed in this thesis. We have given a summarized overview of related work that we will further detail in the subsequent chapters of this thesis where we compare our proposed linking models to the most relevant state-of-the-art methods. Chapter 3

[1] `www.freebase.com`

[2] Figures retrieved from the web site in January, 2015

presents a new model for contextual similarity which is especially suitable for person name disambiguation, the type of entity with presumably highest degree of name ambiguity. In Chapter 4 we generalize to entity linking without type focus and, combining local and global approaches, propose a new method for joint candidate retrieval and weighted semantic relatedness.

Chapter 3

Topic Models for Person Linking

Outline

In this chapter, we propose a contextual approach for entity linking that extends standard word-based approaches with latent topics. The major focus of this chapter is person name linking and we start with a condensed overview on person name discrimination (Section 3.1). This related line of research inspired many later approaches by proposing cosine similarity, a very prominent and effective contextual baseline for entity linking (Section 3.2). More recent work extends contextual similarity with semantic similarity that exploits the relational similarity among entities, for instance based on shared Wikipedia categories. The most related methods will be described in Section 3.3. We will then introduce topic modelling by Latent Dirichlet Allocation that we use to arrive at a more general representation of contexts (Section 3.4). We propose two novel models based on topics. The first interprets topics as semantic labels and is used for German person name disambiguation (Section 3.5). The second formulates thematic distance over mention and entity context (Section 3.6). Evaluating different distances, we show that the proposed kernel method based on symmetric Kullback-Leibler distance yields superior results for person name linking in English, German and French Wikipedia datasets. This method is the first approach that treats person name disambiguation in multiple languages without model reformulation and, albeit being especially suitable for person names, it is also shown to be applicable for general entity linking.

This chapter covers the ideas and findings published in Pilz and Paaß [2009, 2011] and provides additional experimental evaluation to demonstrate the performance of the proposed method.

3.1 Person Name Discrimination

Using Wikipedia as a reference resource distinguishes entity linking from **name discrimination**. Given a set of mentions in a set of documents, name discriminations decides which mentions refer to the same entity but assumes no background information or reference data for these entities. Consequently, most name discrimination

approaches use unsupervised clustering techniques to create batches of documents according to the entity they refer (Mann and Yarowsky [2003], Bekkerman and McCallum [2005], Pedersen et al. [2005], Pedersen and Kulkarni [2008]). Thus, name discrimination is also termed cross-document co-reference resolution. While less closely related to this thesis, the contributions in this field provided important inspiration for later research in entity linking. Bagga and Baldwin [1998] presented one of the first studies in name discrimination. The authors aimed at discriminating mentions of different persons called John Smith in an English news corpus. To arrive at batches of documents referring to one specific entity, the authors clustered documents via the cosine similarity of their word-vectors. Word vectors map words from a dictionary to their counts in a context, here word vectors were formed of all words contained in a context window around the entity mention. Later approaches used different contextual representations and replaced words with semantic units. For instance, Chen and Martin [2007] created context vectors using noun phrases co-occurring with the mention on the sentence level and other named entities appearing in the document. This more selective approach was found to be superior to the all-word context windows proposed by Bagga and Baldwin [1998].

Contextual similarity, and especially cosine similarity, is used in most approaches to entity linking. In this chapter, we will describe context representation via latent topics. Before introducing the theory behind latent topics, we will first detail cosine similarity, the most prominently used contextual similarity measure in the literature.

3.2 Contextual Similarity

Contextual information is one of the most important aspects for entity linking in natural language text. Especially for natural language text, the usage of contextual similarity is motivated by the insight pronounced in Miller and Charles [1991]: the meaning of a word is strongly dependent on the context it appears in and words with similar meanings often appear in similar contexts. This assumption can be generalized to proper names, especially persons, since particular entities will likely be mentioned in certain contexts. For example, we can assume that the basketball player Michael J. Jordan will be mentioned more often with National Basketball Association than the machine learning professor Michael I. Jordan.

Name discrimination approaches may rely merely on the contexts of different mentions. With Wikipedia as a reference, we have both the context of a mention as well as the context of a candidate entity, the latter in form of article texts. This allows us to formulate a similarity function over the two contexts $text(m)$ and $text(e)$. In the literature, the most frequently used function is cosine similarity, which, for a mention context $text(m)$ and an entity context $text(e)$ is given by the

scalar product

$$\cos(text(m), text(e)) = \frac{V(text(m))}{||V(text(m))||} \frac{V(text(e))}{||V(text(e))||} \in [0, 1]. \tag{3.1}$$

The contexts $text(m)$ and $text(e)$ are represented in the standard vector space model over a vocabulary V, i.e. $V(text(m))$ resp. $V(text(e))$. Each vector index corresponds to a term in the underlying vocabulary V and the associated value for example to the term's frequency in the respective context. For our purpose, a comprehensive vocabulary $V_{\mathcal{W}}$ can be created from the collection of all Wikipedia article texts. Then, $V_{\mathcal{W}}(text(e))$ holds terms from $V_{\mathcal{W}}$ that appear in the article text of e. In practice, such a dictionary is not always necessary since cosine similarity can be implemented by counting the number of common words, i.e. words that appear in both contexts, and summing up the total number of words in each context. However, the creation of a vocabulary V is necessary when term weighting schemes such as TF-IDF (Baeza-Yates and Ribeiro-Neto [1999]) are used to replace absolute term frequencies. TF-IDF (short for term frequency-inverse document frequency) reflects how important a word w is for a document d, given a background corpus D. It is computed from the product of the word's frequency in the document, i.e. the number of times $\text{tf}_w(d)$ the word w occurs in d, and the inverse ratio of documents in D containing w:

$$\text{tf-idf}_w = \text{tf}_w(d) \cdot \log \frac{|D|}{|\{d \in D | w \in d\}|} \tag{3.2}$$

Both in the weighted form as well as in the variant using absolute frequencies, the range of cosine similarity is [0,1]. A similarity value at the upper bound of 1 indicates that the two contexts are identical and a similarity value at the lower bound of 0 means that they share no common word. This contextual similarity is an important indicator and word based similarity assessment was proven to be effective for name discrimination and entity linking in the literature (e.g. Bagga and Baldwin [1998], Varma et al. [2009], Mendes et al. [2011], Ratinov et al. [2011]).

In the early study published in Pilz et al. [2009], we evaluated a purely context based linking model over German Wikipedia references of persons[1]. We found that TF-IDF weighted contextual information was sufficient to learn the distinction between covered and uncovered entities but that the discriminative power was not sufficient on highly ambiguous datasets. On a dataset with 10 candidates per mention we obtained an F-measure of only 74.8%, compared to an remarkably higher value of 83.2% for a less ambiguous dataset with only two candidates per mention.

In some cases, contextual cosine similarity can be sufficient for successful linking. Then, in a straightforward approach, we would assign a mention to the candidate

[1]We used the German Wikipedia version from August, 2008.

with highest contextual overlap, i.e.

$$\hat{e}(m) = \underset{e_i(m) \in \mathbf{e}(m)}{\arg\max} \cos(text(m), text(e_i)). \tag{3.3}$$

However, purely contextual approaches based on word comparison have certain drawbacks that may hamper performance. For instance, cosine similarity can not grasp the fine, underlying semantics of words and does not reflect synonymy and polysemy. We will show in this chapter that a generalized representation through *latent topics* is a superior alternative. In short, topics are latent variables in a text and provide important information that may be expressed only implicitly in a document and often does not emerge from a word vector representation. Furthermore, topic models inherently possess the ability to resolve the polysemy and synonymy of words. We will give more details in Chapter 3, where we also introduce a new *thematic* distance that computes similarity on topic level instead of word level. Before we provide details from the topic modelling theory, we first describe other approaches that use semantic information derived from Wikipedia.

3.3 Semantic Similarity

While contextual similarity is defined on a syntactical level, i.e. over the string similarity of contexts, semantic similarity is defined over the likeness of concepts or entities. As there exists no formal, cross-domain definition of semantic similarity, the concrete measure of likeness is often chosen differently by individual approaches to entity linking. The most frequently used measures are here the aforementioned SRL (Milne and Witten [2008a], Eq. 2.2) that measures semantic similarity over shared links, and Wikipedia category based measures that for instance evaluate the overlap of categories assigned to entities (Bunescu and Pasca [2006], Cucerzan [2007]).

Through its hyperlink graph and category system, Wikipedia provides potent resources to extend contextual similarity with semantic similarity. Bunescu and Pasca [2006] were among the first to recognize this and, in the first approach to person name disambiguation towards Wikipedia, demonstrated the usefulness of Wikipedia categories for entity linking. Their approach can be considered one of the most influential in linking to Wikipedia and also provided many inspirations for this thesis.

More specifically, the model proposed by Bunescu and Pasca [2006] extends the word vector approach with a more semantic view through the correlation of context words with Wikipedia categories, in particular the categories assigned to candidate entities. Additionally to features derived from word-category correlation, the authors use the cosine similarity between mention context and candidate context to train and evaluate a supervised Ranking Support Vector Machine (SVM) (Joachims [2002]) on entity references derived from Wikipedia. This model learns the magnitude of semantic correlations between words and categories in Wikipedia and assigns

a mention to the candidate entity whose category set has a higher correlation with the words in the mention context. The ability to predict NIL entities is enabled through a dedicated feature that is active only for NIL candidates. This allows feature based threshold learning which is more elegant than manual threshold tuning. On a dataset of 38726 mentions for ambiguous person names the authors report an accuracy of 84.8% where 10% of the underlying persons were simulated as uncovered NIL entities. Since we compare directly to this method, we will further detail it in Section 3.5.1.

Treating only entities that apply to the *People by occupation* category, Bunescu and Pasca's model is focused on persons. This was extended by Cucerzan [2007], who used categories in the first collective approach for named entity linking to Wikipedia. Cucerzan assumes that related candidates share categories, e.g. Space Shuttle Columbia and Space Shuttle Discovery share the category *Space Shuttle orbiters*, whereas Columbia Pictures shares no categories with these two entities. Based on this, Cucerzan maximizes both the contextual as well as the categorical agreement among all candidate entities for the mentions in a document. This collective agreement is achieved through the usage of so called document vectors. A document vector contains the union of all context terms and the categories from all candidate entities for all mentions in the input document, whereas a candidate entity vector contains all of its entity's context terms and categories. By maximizing the non-normalized scalar products of candidate entity vectors and document vector, those targets are predicted that have the highest contextual overlap and categorical relatedness. Here, context terms are extracted both from a candidate's article text, as well as from selected inlink references of this candidate. The relevant categories are derived from a filtered set of Wikipedia categories and categorical list pages such as List of Television Series.

Recently, Hachey et al. [2013] reported comparable results for named entity linking achieved with own implementations of the systems proposed by Bunescu and Pasca [2006] and Cucerzan [2007]. However, to render Bunescu and Pasca's approach more suitable for named entity linking, the authors adapted the category set used for the implementation of Bunescu and Pasca, as originally this set was chosen to be most useful for person name linking. We will further elaborate the findings of Hachey et al. [2013] in Section 3.7 where we discuss named entity linking in more detail.

Both of the described approaches showed satisfactory performance using semantic information from categories. However, categorization is expensive and requires humans to provide expressive categories and assign them to Wikipedia articles. Furthermore, both the extraction as well as the selection of a useful category set is non trivial, especially when aiming for more than one language version of Wikipedia. In a first study, we therefore experimentally evaluated an approach that replaces the semantic information from categories with the semantic information from topic models (Pilz and Paaß [2009]) and found results comparable to Bunescu and Pasca. We will detail our findings in Section 3.5 and show that automatically generated

semantic information can replace manually assigned categories.

Another Wikipedia resource of semantic relatedness are links. Clues from connectivity or shared links, as first observed in Bekkerman and McCallum [2005], are now most prominent in the semantic relatedness measure introduced by Milne and Witten [2008b]. This powerful coherence measure for entities is used in most of the recent entity linking methods, for instance as the weight for edges among entity nodes in graphical models (e.g. Han et al. [2011], Hoffart et al. [2011b]). In Pilz [2010] we explicitly investigated the semantic but also discriminative information conveyed by links. Building upon the word-category correlation proposed in Bunescu and Pasca [2006], we replaced categories with outlink targets assuming that the semantics of outlinks provide similar discriminative information. We compared this method to Bunescu and Pasca's approach on person references from the German Wikipedia. However, even though using additional weights derived from SRL gave better results than a binary variant, the outlink-word correlation yielded comparable but inferior results to the word-category correlation approach. Wikipedia categories provide a very distinctive feature set with inherent semantics. Links, in contrast, form a more diverse and also noisier feature set. For instance, we observed an average of 20 outlinks per entity, while the average for categories was 6 per entity. The weighted variant being significantly superior to the binary variant, we assume that a more sophisticated selection of relevant outlink targets might have been more useful. For example, Cucerzan [2007] used only outlinks appearing in the first paragraph or pointing back to the entity of interest.

To summarize, semantic information derived from links or categories is a good extension for purely contextual information. But retrieving and selecting relevant and useful sets of either links or categories is a difficult task that needs further investigation. As an alternative, we propose automatically generated semantic information derived from topic models. We will describe one instance of topic models, namely Latent Dirichlet Allocation, in the next section and empirically demonstrate its strength in the remainder of this chapter, either as a means to derive semantic labels or as a means of context representation.

3.4 Latent Dirichlet Allocation

Since its introduction by Blei et al. [2003], topic modelling by Latent Dirichlet Allocation (LDA) achieved very much attention in a growing number of research fields, ranging from text analysis to computer vision. In the context of natural language processing, topics generated by LDA are clusters of words that often co-occur. These topics are used for example to find related documents, to summarize documents or create the input for other text categorization tasks (Blei et al. [2003], Griffiths and Steyvers [2004], Rubin et al. [2012] among many others).

LDA has a its core a Bayesian probabilistic model that describes document corpora

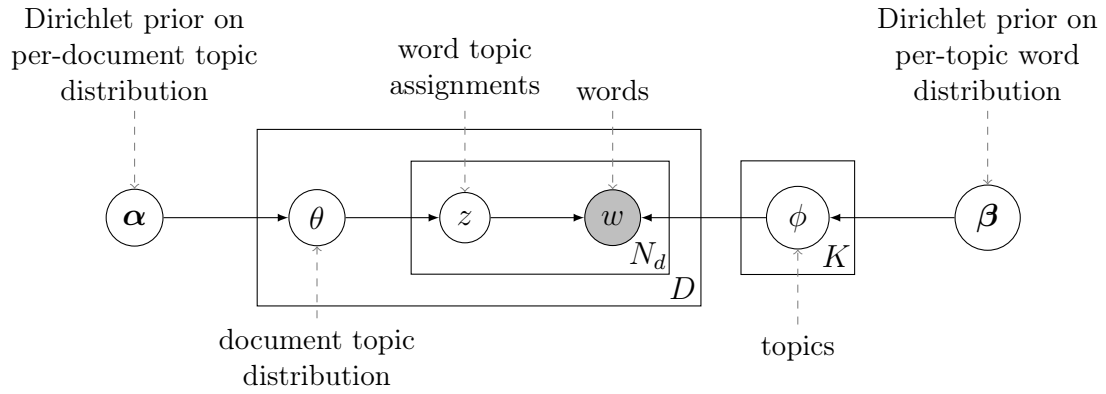


Figure 3.1: Plate notation of smoothed LDA after Blei et al. [2003]. The plates (rectangles) represent repetitions of variables (circles) in the graphical model. The outer plate represents a collection of D documents, the inner plate represents the repeated choice of topics and words within a document. The observable variables, i.e. words, are shaded in grey.

in a fully generative way. LDA assumes a fixed number K of underlying topics in a document collection where each document is a mixture of topics and generated by picking a distribution over the latent topics. Given this mixture, the topic of each word is chosen and, given their topics, the observable variables, i.e. the words, are generated. This process is depicted in Fig. 3.1 and formally described as follows.

Assume we have a vocabulary V of $|V|$ words and want to generate D documents of sizes $N_1, \ldots, N_D$:

1. Randomly draw the overall topic distribution $\phi_k \sim \text{Dir}(\boldsymbol{\beta})$, $\forall\ k = 1, \ldots, K$ with $\phi_k \in \mathbb{R}^{|V|}$, $\phi_{k,i} \geq 0$ and $\sum_{i=1}^{|V|} \phi_{k,i} = 1$. K is a fixed number used to assess the number of latent topics in the corpus. The parameter $\boldsymbol{\beta} \in (0, \infty)^{|V|}$ is the prior vector on the per-topic word distribution.

2. Randomly draw document-specific topic proportions $\theta_d \sim \text{Dir}(\boldsymbol{\alpha})$, $\forall\ d = 1, \ldots, D$ with $\theta_d \in \mathbb{R}^K$, $\theta_{d,k} \geq 0$ and $\sum_{k=1}^{K} \theta_{d,k} = 1$. The probability vector θ_d describes the distribution of topics in document d. The parameter $\boldsymbol{\alpha}$, also a positive vector of dimension K, is the concentration parameter of the Dirichlet prior on the per-document topic distributions.

3. For each of the words $w_{d,n}$, $\forall\ d = 1, \ldots, D$, $\forall\ n = 1, \ldots, N_d$
   a) Randomly draw a topic $z_{d,n} \sim \text{Multinomial}\,(\theta_d)$, $z_{d,n} \in \{1, \ldots, K\}$.
   b) Finally, the observed word $w_{d,n} \in V$ is randomly drawn from the distribution of the selected topic: $w_{d,n} \sim \text{Multinomial}\left(\phi_{z_{d,n}}\right)$.

In Fig. 3.1, the repeated draws of random variables and observable variables (circles) are depicted through the plates (rectangles).

The fundamental idea in probabilistic topic models is that the words of a document d are generated according to a mixture of topic distributions, where the mixture depends on the document-specific mixture weights θ_d. LDA introduces a Dirichlet prior on θ and in this way extends Probabilistic Latent Semantic Indexing (Hofmann [1999]), which makes no assumption on the prior distributions. The Dirichlet distribution, a conjugate prior for the Multinomial distribution, is a convenient choice as prior, simplifying the problem of statistical inference. Using a Dirichlet prior for the topic distribution θ results in a smoothed topic distribution, with the amount of smoothing determined by the parameter $\boldsymbol{\alpha} = \alpha_1, \ldots, \alpha_K$. Each parameter α_k can be interpreted as a prior observation count for the number of times topic k is sampled in a document, before having observed any actual words from that document (Gelman et al. [2013]).

The nature and influence of the priors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ was studied in Wallach et al. [2009]. The authors empirically found that an asymmetric prior $\boldsymbol{\alpha}$ over document-topic distributions and a symmetric prior $\boldsymbol{\beta}$ over topic-word distributions gives best results. In the symmetric prior, all $\beta_1, \ldots, \beta_{|V|}$ have the same value, in the asymmetric prior, all $\alpha_1, \ldots, \alpha_K$ have different values. The findings are implemented in the toolkit Mallet (McCallum [2002]), which optimizes the prior $\boldsymbol{\alpha}$ according to the underlying collection in a Markov Chain Monte Carlo method. Mallet uses an implementation of Gibbs sampling, i.e. SparseLDA (Yao et al. [2009]), a statistical technique meant to quickly construct a sample distribution. It repeatedly samples a topic for each word in each document using the distributions defined by the model. After some time this distribution converges to a stationary state where the topic probability distribution of each word in a document remains constant. All topic models used in this thesis are generated using this software[1].

To infer the topic distribution for a new document, the topic distribution is sampled in the same way as for training. Given the set of observed words, LDA estimates which topic configuration is most likely to have generated the data by sampling a distribution based on the word counts. The average probability of topic ϕ_k for a document is then the average of the probabilities of topic ϕ_k for each word w in this document.

The properties described above allow topic models based on LDA to alleviate synonymy and polysemy. Synonyms such as *car* and *automobile* have the same meaning and will usually co-occur in similar contexts and hence usually belong to the same topic. On the other hand, as LDA is a generative model, there is no notion of mutual exclusivity. Words may belong to more than one topic. This allows LDA to capture polysemy: depending on the context at hand, different topics will be assigned to a word like *plant*. If a document is mostly about industry LDA will assign a topic that hints at the industrial plant. If the document is mostly about biology, LDA will assign a topic that hints at the biological plant.

[1]The newest version of this software is available at `http://mallet.cs.umass.edu`

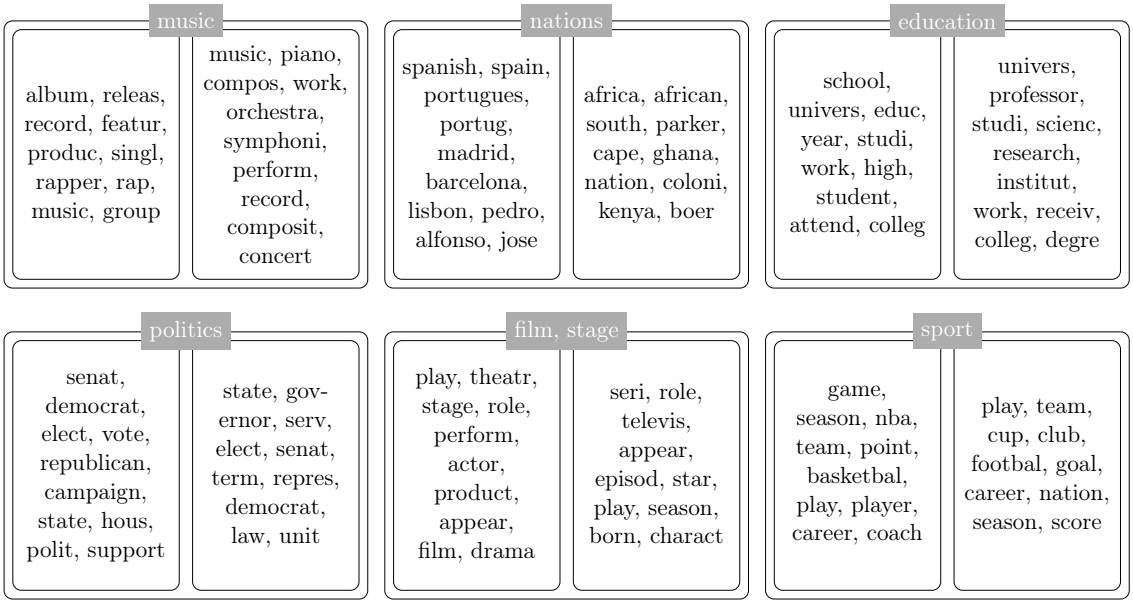


Figure 3.2: Topics from a topic model ($K = 200$) trained with 100k Wikipedia articles. Each topic is depicted by its associated words and a manually assigned label (grey box). The order of appearance implies the importance of a word for a topic.

Another example is the music topic in Fig. 3.2. As the figure shows, the word *music* appears in two music related topics, where one is more about classical music and the other more about modern music. Similarly, the word *season* appears both in the basketball and the soccer topic in Fig. 3.2.

3.4.1 Topic Distributions as Context Descriptions

Vanilla LDA is based on the bag-of-words assumption, as the only relevant information is the number of times words are generated in a document. However, it allows for a more general document representation. LDA effectively generates low-dimensional representations from sparse high-dimensional data and is a means to substitute high-dimensional bag-of-words vectors with low-dimensional topic mixture vectors. Accordingly, we may represent a document as a mixture of K topics that summarizes the main content of the document, relative to the topic model. At the same time, the representation via topic clusters provides a generalization to a wider context as topic clusters potentially grasp more information than the implicitly expressed, and thus latent information carried by the observed words in a text document.

To illustrate how topics describe entities, Fig. 3.3 summarizes the topic distributions for the contexts of three entities called John Taylor from the English Wikipedia.

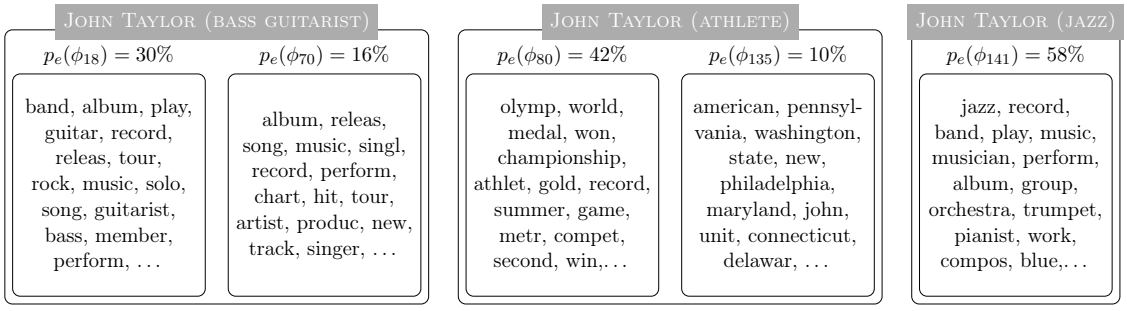


Figure 3.3: Topics for concrete entities in the English Wikipedia. The figure shows the most important words of the topics ϕ_k with highest probability $p_e(\phi_k)$ to have generated the article texts $text(e)$ of the respective entities.

The depicted topics are generated from a topic model with $K = 200$ that was trained on a random selection of 100k Wikipedia articles describing persons and used to infer the topic probability distributions on the article texts of the entities John Taylor (athlete), John Taylor (bass guitarist) and John Taylor (jazz). For each of these entities, Fig. 3.3 shows the two topics ϕ_k that have the highest probability $p_e(\phi_k)$ to have generated the article texts $text(e)$ of the respective entities. Each topic ϕ_k is depicted by a selection of words associated with it. The association of words and topics is based on the probability that a topic has generated a word. Here, a selection of high probability words is shown. These words can be interpreted as important for a topic and also be understood as a summary of an entity's article text. This example also illustrates the dimensionality reduction provided by LDA: the three entities here are well described by only one or two topic clusters that also enable a distinction among these entities on the first glance.

For example, the most prominent topic ϕ_{80} derived for John Taylor (athlete) describes his sportive success in the Olympic Games. The topic ϕ_{135} with lower probability can be interpreted as an indicator for his nationality. This entity is described by a rather short article text and therefore less informative topics such as the nationality topic may get higher weight. Note that we should generally consider document length as this length influences the total word count. Hence it also influences the inferred topic distribution of the document. In very short documents we often find one very prominent topic, longer documents usually have a higher variety but nevertheless a large number of topics with a near-zero probability.

To distinguish among topic distributions for entity and mention context, we use the following notation.

Notation (Topic distribution over mention and entity context)
We denote the probability distribution of K topics in the mention context $text(m)$ with

$$\mathcal{T}_m = \mathcal{T}(text(m)) = (p_m(\phi_1), \ldots, p_m(\phi_K)) \,,$$

where $p_m(\phi_k)$ denotes the probability of topic ϕ_k in the context of mention m. Analogously, we denote the probability distribution of K topics in the entity context $text(e)$ with

$$\mathcal{T}_e = \mathcal{T}(text(e)) = (p_e(\phi_1), \ldots, p_e(\phi_K)),$$

where $p_e(\phi_k)$ denotes the probability of topic ϕ_k in the context of entity e.

3.5 Semantic Labelling of Entities

Since topic probability distributions can be interpreted as semantic labels, LDA is also applicable for (soft) multi-label document classification or categorization tasks (Rubin et al. [2012]). From a given set of labels (or topics), the most relevant labels are assigned to a document based on the textual evidence in the context. In extension to standard multi-label classification, topic labels also have an associated probability value that provides the relevance of each label.

In this section, we propose the usage of semantic labels from topics for person name disambiguation. We compare to the method proposed in Bunescu and Pasca [2006] who employed Wikipedia categories as indicators of semantic relatedness for the disambiguation of person names. To describe both methods in detail, we start with the approach of Bunescu and Pasca.

3.5.1 Semantic Labels from Categories

The model proposed in Bunescu and Pasca [2006] uses categories of Wikipedia articles to learn the magnitude of semantic correlations between words contained in the Wikipedia dictionary $V_{\mathcal{W}}$ and Wikipedia categories $\mathbf{C}_{\mathcal{W}}$. Being focused on the disambiguation of person names, the authors did not use all Wikipedia categories $\mathbf{C}_{\mathcal{W}}$ but a specialized subset. This subset was formed of 540 child-categories of the category *People by occupation* and each category in this subset was required to relate to at least 200 articles. For simplicity and to describe Bunescu and Pasca's model in general, we will here use the notation $\mathbf{C}_{\mathcal{W}}$ also to refer to this category set. Important differences among category sets will be emphasized appropriately when necessary.

The proposed word-category-correlation (WCC) is realized in a word-category dictionary that pairs each word $w_i \in V_{\mathcal{W}}$ with all categories $c_j \in \mathbf{C}_{\mathcal{W}}$, i.e.

$$V_{\mathcal{W}} \times \mathbf{C}_{\mathcal{W}} = \{(w_i, c_j)\},\ w_i \in V_{\mathcal{W}}, c_j \in \mathbf{C}_{\mathcal{W}}. \tag{3.4}$$

For a given mention m, Bunescu and Pasca then use the words from the mention context $text(m)$ that are contained in $V_{\mathcal{W}}$, i.e. $V_{\mathcal{W}}(w_i \in text(m))$, and the categories $\boldsymbol{c}(e)$ applying to a candidate entity e, to create binary feature vectors for all

candidates $e(m) \in \mathbf{e}(m)$:

$$\text{for } w_i \in text(m), c_j \in \boldsymbol{c}(e), e(m) \in \mathbf{e}(m): \quad x_{\text{WCC}}(m,e) = \begin{cases} 1, & \forall\,(w_i, c_j) \in \{V_{\boldsymbol{\mathcal{W}}}(w_i \in text(m)) \times \boldsymbol{c}(e)\} \\ 0, & \text{else.} \end{cases} \tag{3.5}$$

According to Eq. 3.5, a feature vector $x_{\text{WCC}}(m,e)$ contains a binary feature for every possible pair (w_i, c_j) of context words $w_i \in text(m)$ contained in the vocabulary $V_{\boldsymbol{\mathcal{W}}}$ and entity categories $c_j \in \boldsymbol{c}(e) \subset \mathbf{C}_{\boldsymbol{\mathcal{W}}}$. For illustration, we give the following example.

Example 5 (Word-Category-Correlation (WCC))
Assume two candidate entities e_1 and e_2 with categories $\boldsymbol{c}(e_1) = \{c_1, c_2\}$ and $\boldsymbol{c}(e_2) = \{c_3, c_4\}$, i.e. we have $\mathbf{C}_{\boldsymbol{\mathcal{W}}} = \{c_1, c_2, c_3, c_4\}$. Assume further a mention context $text(m) = \{w_1, w_2\}$ and a Wikipedia vocabulary $V_{\boldsymbol{\mathcal{W}}} = \{w_1, w_2, w_3\}$. As by Eq. 3.4, the word-category dictionary here consists of

$$V_{\boldsymbol{\mathcal{W}}} \times \mathbf{C}_{\boldsymbol{\mathcal{W}}} = \{(w_1, c_1), (w_1, c_2), (w_1, c_3), (w_1, c_4), \ldots, (w_3, c_1), (w_3, c_2), (w_3, c_3), (w_3, c_4)\}.$$

According to Eq. 3.5, the feature vector $x_{\text{WCC}}(m, e_1)$ relating candidate e_1 to the mention m is composed of:

$$x_{\text{WCC}}(m,e_1) = \begin{cases} 1, & \forall\,(w_i, c_j) \in \{(w_1, c_1), (w_1, c_2), (w_2, c_1), (w_2, c_2)\} \\ 0, & \forall\,(w_i, c_j) \in \{(w_1, c_3), (w_1, c_4), (w_2, c_3), (w_2, c_4)\}. \end{cases}$$

The vector representing the pair (m, e_2) is build analogously:

$$x_{\text{WCC}}(m,e_2) = \begin{cases} 0, & \forall\,(w_i, c_j) \in \{(w_1, c_1), (w_1, c_2), (w_2, c_1), (w_2, c_2)\} \\ 1, & \forall\,(w_i, c_j) \in \{(w_1, c_3), (w_1, c_4), (w_2, c_3), (w_2, c_4)\}. \end{cases}$$

Bunescu and Pasca choose for the words constituting the context $text(m)$ a context window of width 25 around the mention m. Technically, a vector $x_{\text{WCC}}(m,e)$ can be empty when no word from the context of a mention m is contained in the dictionary $V_{\boldsymbol{\mathcal{W}}}$. Bunescu and Pasca therefore use the cosine similarity as in Eq. 3.1 as baseline attribute. In that case, the representing vector would have a zero as sole entry.

Alternatively to using the full vocabulary $V_{\boldsymbol{\mathcal{W}}}$, we may also treat only common words $w_i \in text(m) \cap text(e)$, i.e. words that appear simultaneously in the mention and the candidate entity context. This can be viewed as a candidate specific feature

selection where only those words are assumed to be influential that are used in both contexts. This results in a slightly different feature vector representation:

$$\text{for } w_i \in text(m) \cap text(e) \in V_{\boldsymbol{\mathcal{W}}}, c_j \in \boldsymbol{c}(e), e(m) \in \mathbf{e}(m) :$$
$$x_{\text{cWCC}}(m,e) = \begin{cases} 1, & \forall\,(w_i, c_j) \in \{(w_i \in text(e \cap text(e))) \times \boldsymbol{c}(e)\} \\ 0, & \text{else.} \end{cases} \tag{3.6}$$

The formulation above was used for the results reported Pilz and Paaß [2009] and Pilz and Paaß [2011]. We will present the obtained results in Section 3.5.4. For a better comparability of the method proposed in Pilz and Paaß [2011], we will additionally evaluate against WCC with a full vocabulary in Section 3.6.4. There, we will also show that the candidate specific feature selection using only common words can increase linking performance.

We will now describe the approach proposed in Pilz and Paaß [2009] by motivating the usage of topics as semantic labels for person name disambiguation.

3.5.2 Semantic Labels from Topics

Since they group articles by subject, Wikipedia categories can also be interpreted as document labels. However, they do not express relevance or any other weighting scheme comparable to a topic distribution. Moreover, categories are manually assigned by contributors and hence subject to the individual taste of an author. Even though there exist clear guidelines on the assignment of existing and the creation of new Wikipedia categories, these are not necessarily strictly followed. Analysing Wikipedia categories, we observed that categories can be very general but also overly specific. One example from the German Wikipedia is the category *Träger des Bundesverdienstkreuzes (Großkreuz in besonderer Ausführung)* that applies only to the two entities Konrad Adenauer and Helmut Kohl. Notably, the latter entity even has its own Wikipedia category. On the other hand, categorization may also be incomplete and not fully descriptive for an entity.

Assuming that topic distributions provide a more expressive summary of the article text, we propose a model that replaces Wikipedia categories with semantic labels derived from topic probability distributions, i.e. topic indices. This method was published in Pilz and Paaß [2009], the first kernel based entity linking approach using the German Wikipedia.

For further motivation, we give an example that compares the information content of topic distributions and Wikipedia categorization for two entities from the German Wikipedia: a politician Willi Weyer (Politiker)[1] and a soccer player Willi Weyer (Fussballspieler). For this, we use a topic model with $K = 200$ topics

[1]While the original title does not contain a disambiguation term, we use one here for better distinction.

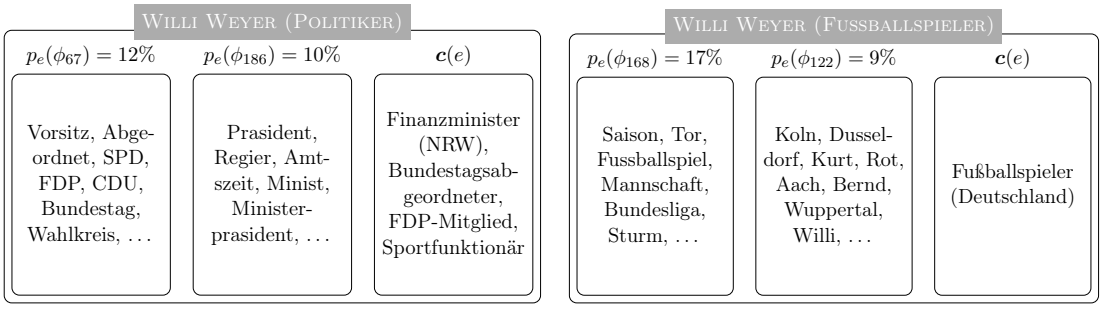


Figure 3.4: The most important words of topics ϕ_k for the entities Willi Weyer (Politiker) and Willi Weyer (Fussballspieler) along with each entity's categories $\boldsymbol{c}(e)$. Topics are automatically inferred from LDA, categories are manually assigned by Wikipedia contributors.

that was trained on 100K randomly selected articles from the German Wikipedia that describe persons. From these articles we extracted the full text, removed markup language and used words in stemmed form with stems obtained from the German version of the Snowball algorithm (Porter [2001]). Using this model, we inferred the topic distributions summarized in Fig. 3.4 for our example entities. As depicted in the figure, we observe two topics for the politician that represent his occupation. For the soccer player, we also find one dominant topic describing his occupation. We also find a second topic showing the names of cities that represent the football teams he was engaged with, e.g. Cologne.

Now, the information covered by the respective Wikipedia categories differs notably. The politician is assigned several categories related to his affiliation, for instance his appointment as finance minister (*Finanzminister (NRW)*[1]) or his political party (*FDP-Mitglied*). Comparing manually assigned categories and automatically generated LDA topics we here find a strong semantic overlap. However, the soccer player Willi Weyer (Fussballspieler) is assigned only one category, i.e. *Fußballspieler (Deutschland)*, that expresses his profession and nationality. The inferred topic distribution does also express his profession, but furthermore relates him to the city of Cologne and thus also hints at the soccer club he was engaged with. While not being expressed explicitly but latently, the inferred topic distribution seems to carry much more information than the assigned category.

Note that the indices of the topics in this example, i.e. ϕ_{67}, ϕ_{186}, ϕ_{168}, and ϕ_{122}, may serve as abstract labels for their specific distribution over words. Even though these semantic labels have limited interpretability for a human, at least without the knowledge of the associated words, we argue that they can be used as a replacement for the manually assigned Wikipedia categories. We also assume that since topic models rely on the article text and not on the contributor's intuition they potentially

[1]NRW is the acronym for the German federal state Nordrhein-Westfalen.

yield a more representative assignment of labels.

Having motivated that topic distributions can be interpreted as semantic multi-label assignments, we will now describe the entity linking based upon this. This model is inspired by Bunescu and Pasca's word-category correlation method but replaces Wikipedia categories with topic assignments and is therefore independent of error-prone and costly manual document categorization.

We evaluate word-topic correlation (WTC) by correlating each common word $w_i \in text(m) \cap text(e)$ with the topic distribution $\mathcal{T}_e$ of the candidate context. This is analogous to the formulation of cWCC (Eq. 3.6), where we assumed that words shared by the two contexts are more influential and that entity specific feature selection is also useful. So here we have a word-topic-correlation dictionary that pairs each common word w_i from the mention context with the probability $p_e(\phi_k)$ for a topic in the candidate context, i.e

$$V_{\mathcal{W}} \times \mathcal{T}_e = \{(w_i, p_e(\phi_k))\},\ w_i \in text(m) \cap text(e), \phi_k, k = 1, \ldots, K. \tag{3.7}$$

Building upon Eq. 3.6, we substitute categories with topics and binary values with document topic probability values $p_e(\phi_k), k = 1, \ldots, K$, i.e. the probability of each specific topic in the context of candidate entity e. More specifically, for each candidate $e(m) \in \mathbf{e}(m)$ we create feature vectors according to

$$\text{for } w_i \in text(m) \cap text(e) \in V_{\mathcal{W}}, \phi_k, k = 1, \ldots, K, e(m) \in \mathbf{e}(m):$$
$$x_{\text{WTC}}(m, e) = \begin{cases} p_e(\phi_k), & \forall\, (w_i, \phi_k) \in \{(w_i \in text(m) \cap text(e)) \times \mathcal{T}_e\} \\ 0, & \text{else.} \end{cases} \tag{3.8}$$

With the formulation above, the vector $x_{\text{WTC}}(m, e)$ representing the word-topic-correlation for a mention-entity pair (m, e) contains K probability values for every common word w_i that appears both in the mention's as well as in the candidate's context. The maximum dimension of such a vector is then $K \cdot |V_{\mathcal{W}}|$ where at most $K \cdot |text(m) \cap text(e)|$ entries have non-zero values.

Analogously to the feature vector representations of WCC (Eq. 3.5) and cWCC (Eq. 3.6), the vector $x_{\text{WTC}}(m, e)$ would be empty if the mention context and the candidate context share no common word. Therefore we extend this feature vector with the cosine similarity (Eq. 3.1) as baseline feature that in such cases evaluates to $\cos(text(m), text(e)) = 0$. As for WCC, we give also here a small example for better illustration.

Example 6 (Word-Topic-Correlation (WTC))
Assume a topic model with $K = 2$, build over the contexts of two entities e_1 and e_2, with $text(e_1) = \{w_1, w_2\}$ and $text(e_2) = \{w_3, w_4\}$. This results in the word-topic dictionary

$$V_{\mathcal{W}} \times \mathcal{T}_e = \{(w_1, \phi_1), (w_1, \phi_2), (w_2, \phi_1), (w_2, \phi_2), (w_3, \phi_1), \ldots, (w_4, \phi_2)\}.$$

Further, assume the topic distribution $\mathcal{T}_{e_1} = \{0.3, 0.7\}$ and a mention context $text(m) = \{w_1, w_2\}$. According to Eq. 3.8, the vector $x_{\text{WTC}}(m, e_1)$ representing the pair of candidate e_1 and mention m is composed of:

$$x_{\text{WTC}}(m, e_1) = \begin{cases} p_{e_1}(\phi_1) = 0.3, & \forall\, (w, \phi_k) \in \{(w_1, \phi_1), (w_2, \phi_1)\} \\ p_{e_1}(\phi_2) = 0.7, & \forall\, (w, \phi_k) \in \{(w_1, \phi_2), (w_2, \phi_2)\} \\ 0, & \text{else.} \end{cases}$$

The full instantiation of this vector is given by

$$x_{\text{WTC}}(m, e_1) = \underset{\substack{\downarrow \\ (w_1,\phi_1)}}{[0.3,} \quad \underset{\substack{\uparrow \\ (w_2,\phi_1)}}{0.3,} \quad \underset{\substack{\downarrow \\ (w_3,\phi_1)}}{0,} \quad \underset{\substack{\uparrow \\ (w_4,\phi_1)}}{0,} \quad \underset{\substack{\downarrow \\ (w_1,\phi_2)}}{0.7,} \quad \underset{\substack{\uparrow \\ (w_2,\phi_2)}}{0.7,} \quad \underset{\substack{\downarrow \\ (w_3,\phi_2)}}{0,} \quad \underset{\substack{\uparrow \\ (w_4,\phi_2)}}{0]} \in [0,1]^{K \cdot |V_{\mathcal{W}}|}.$$

Since $text(m) \cap text(e_2) = \emptyset$, the vector $x_{\text{WTC}}(m, e_2)$ representing the pair (m, e_2) has no word-topic correlation features and contains only a zero representing the cosine similarity of the contexts.

Having detailed the feature design of our method WTC and that of its inspiration WCC proposed in Bunescu and Pasca [2006], we will now come to the machine learning method exploiting these designs in order to learn a model for entity linking. This model is based on ranking candidate entities with respect to a given mention and its context and defines a feature based threshold learning for the detection of uncovered entities. We will then use this method in our experiments to compare WTC and WCC for person name disambiguation in German.

3.5.3 Linking as a Ranking Problem

Bunescu and Pasca [2006] learn their model based on a ranking algorithm, more specifically a Ranking SVM. This algorithm was first introduced in Joachims [2002] in the context of search engine analysis and was also used in later linking approaches, for instance in Dredze et al. [2010], Zheng et al. [2010] and Pilz and Paaß [2012]. Since providing full details on SVMs and Ranking SVMs is out of the scope of this thesis, we here provide the basic ideas and briefly point out how Ranking SVMs differ from standard SVMs. We assume background knowledge on SVMs and hint the kind reader at Cortes and Vapnik [1995] or Vapnik [2000] for further details. We will again refer to Ranking SVMs in Chapter 4, where we use this algorithm for general entity linking. For all ranking and classification models trained and

evaluated in this thesis, we use the SVM$^{\text{Light}}$ implementation by Thorsten Joachims[1] that provides both standard classification as well as an adaption for ranking.

Now, a ranking approach for entity linking can be summarized as follows. For a mention m and a set of n candidates $\mathbf{e}(m) = \{e_1(m), \ldots, e_n(m)\}$, the optimal result of a ranking algorithm is a ranking $r^* = \{r_1, \ldots, r_n\} \in \mathbb{R}^n$ that orders the n candidate entities $\mathbf{e}(m)$ according to their fitness to the mention (or the mention context). In our case, a ranking can be considered correct if the correct underlying entity $e^+(m)$ is ranked at the top position. To describe the underlying technique, we use the description as in Pilz and Paaß [2009] that closely follows that in Joachims [2002] but adapt notation.

As in Joachims [2002] we start with a collection of entities $\mathbf{e} = \{e_1, \ldots, e_{|\boldsymbol{\mathcal{W}}|}\}$. For a mention m we want to determine a list of relevant entities in $\mathbf{e}$, where the most relevant entities appear first. This corresponds to a ranking relation $r^*(m) \subseteq \mathbf{e} \times \mathbf{e}$ that fulfills the properties of a weak ordering, i.e. asymmetric and transitive. If an entity e_i is ranked higher than e_j for an ordering r, i.e. $e_i <_r e_j$, then $(e_i, e_j) \in r$, otherwise $(e_i, e_j) \notin r$.

We have to measure the similarity of a proposed ranking $r(m)$ and the target ranking $r^*(m)$. Such a measure is Kendall's τ (Kendall [1955]) which is a function of the number n_e of concordant pairs in relation to all pairs. A pair $e_i \neq e_j$ is *concordant* if either $(e_i, e_j) \in r_a \wedge (e_i, e_j) \in r_b$ or $(e_j, e_i) \in r_a \wedge (e_j, e_i) \in r_b$.

Now assume we have a training set $\mathcal{D}$ containing n different i.i.d. mentions m_i with target rankings

$$\mathcal{D} = (m_1, r_1^*), (m_2, r_2^*), \ldots, (m_n, r_n^*), \tag{3.9}$$

where $r_i^* \in \mathbf{e} \times \mathbf{e}$ is a ranking on the entities at hand. To achieve a ranking close to the ground truth r^*, a learner will select a ranking function $f(m)$ based on the training instance $\mathcal{D}$ that maximizes the empirical $\tau_{\mathcal{D}}$ (Kendall [1955]), which measures the similarity of two rankings on the training sample, i.e.

$$\tau_{\mathcal{D}}(f) = \frac{1}{n} \sum_{k=1}^{n} \tau(r_{f(x(m,e_k(m)))}, r_k^*), \tag{3.10}$$

where $r_{f(x(m,e_k(m)))}$ is the ranking induced by the ranking function f and r_k^* the target ranking.

Maximizing Eq. 3.10 is analogous to classification by minimizing training error, with the difference that the target is not a class label, but a binary ordering relation. Thus, whereas in standard SVMs constraints are formulated over the offset from a separating hyperplane, Ranking SVMs impose different constraints, since additionally the relative ordering of the examples has to be modelled. Consider the class of linear ranking functions

$$(e_i, e_j) \in f_w(m) \iff w \cdot x(m, e_i) > w \cdot x(m, e_j) \tag{3.11}$$

[1]The software is available at `http://svmlight.joachims.org`

where $x(m, e_i) \in \mathbb{R}^d$ is a vector of d real-valued features that for instance describe the fitness between candidate and mention and $w \in \mathbb{R}^d$ is a weight vector of matching dimension. For the class of linear ranking functions in Eq. 3.11, maximizing the number of concordant pairs, i.e. maximizing Eq. 3.10, is equivalent to finding the weight vector w so that the maximum number of the following inequalities hold:

$$\begin{aligned} \forall(e_i, e_j) \in r_1^* &: w \cdot x(m_1, e_i) > w \cdot x(m_1, e_j) \\ &\vdots \\ \forall(e_i, e_j) \in r_n^* &: w \cdot x(m_n, e_i) > w \cdot x(m_n, e_j) \end{aligned} \tag{3.12}$$

The exact solution of this problem is NP-hard. As proposed in Joachims [2002], and just like in classification SVMs, the solution is approximated by introducing non-negative slack variables $\xi_{i,j,k}$ and minimizing the upper bound, i.e. the sum of slack variables $\sum \xi_{i,j,k}$. Regularizing the length of w to maximize margins leads to the following optimization problem:

$$\text{minimize}: \; V(w, \xi) = \frac{1}{2} w \cdot w + C \sum_{i=1}^{|\mathbf{e}|} \sum_{j=1}^{|\mathbf{e}|} \sum_{k=1}^{n} \xi_{i,j,k} \tag{3.13}$$

$$\begin{aligned} &\text{subject to}: \\ \forall(e_i, e_j) \in r_1^* &: w \cdot x(m_1, e_i) \geq w \cdot x(m_1, e_j) + 1 - \xi_{i,j,1} \\ &\vdots \\ \forall(e_i, e_j) \in r_k^* &: w \cdot x(m_k, e_i) \geq w \cdot x(m_k, e_j) + 1 - \xi_{i,j,k} \\ \forall i \; \forall j \; \forall k \; &: \; \xi_{i,j,k} \geq 0 \end{aligned} \tag{3.14}$$

The parameter C is the usual parameter capturing the trade-off between margin size and training error in terms of n_e. As noted in Joachims [2002], this optimization problem is comparable to the ordinal regression approach in Herbrich et al. [2000]. Further, it is convex and has no local optima. By rearranging the constraints in Eq. 3.14 as

$$w \cdot (x(m_k, e_i) - x(m_k, e_j)) \geq 1 - \xi_{i,j,k} \tag{3.15}$$

it becomes apparent that the optimization problem is equivalent to that of a classification SVM on pairwise difference vectors $x(m_k, e_i) - x(m_k, e_j)$. Due to this similarity, it can be solved using decomposition algorithms similar to those used for SVM classification.

To formulate inference using such a ranking function, we first note that it can be shown that a learned ranking function $f_{w^*}(m)$ can always be represented as a linear combination of the feature vectors:

$$\begin{aligned} (e_i, e_j) \in f_{w^*}(m) &\Leftrightarrow w^* \cdot x(m, e_i) > w^* \cdot x(m, e_j) \\ &\Leftrightarrow \sum a^*_{k,l} x(m_k, e_l) \cdot x(m, e_i) > \sum a^*_{k,l} x(m_k, e_l) \cdot x(m, e_j), \end{aligned} \tag{3.16}$$

where w^* is the learned weight vector and $a^*_{k,l}$ are derived from the values of the Lagrangian dual variables at the solution. Further, we note that the learned ranking

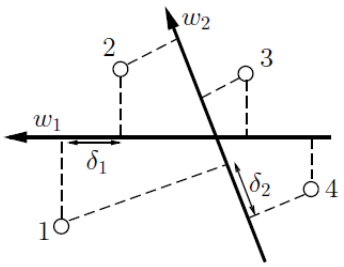


Figure 3.5: Example of two weight vectors w_1 and w_2 ranking four points (after Joachims [2002]). The margin δ is the distance between the closest two projections within all target rankings. For w_1 and δ_1, these are the points 1 and 2, for w_2 and δ_2 the points 1 and 4.

function $f_{w^*}(m)$ is here used to rank a set of candidates according to a mention m. Aiming at the candidate with highest rank, it is then sufficient to sort these candidates by their value of

$$rank(x(m, e_i)) = w^* \cdot x(m, e_i) = \sum a^*_{k,l} x(m_k, e_l) \cdot x(m, e_j). \tag{3.17}$$

The final prediction $\hat{e}$ is then given by

$$\hat{e} = \underset{e_i \in \mathbf{e}(m)}{\arg\max}\, rank(x(m, e_i)) = \underset{e_i \in \mathbf{e}(m)}{\arg\max}\, w^* \cdot x(m, e_i). \tag{3.18}$$

An exemplary ordering implied by a weight vector w is illustrated in Fig. 3.5 (adapted from Joachims [2002]). The figure illustrates how a weight vector w determines the ordering of four points in a two-dimensional example. For any weight vector w, the points are ordered by their projection onto w, which is equivalent to an ordering by the signed distance to a hyperplane with normal vector w. In the example in Fig. 3.5, this means that for w_1 the points are ordered (1,2,3,4), while w_2 implies the ordering (2,3,1,4) (Joachims [2002]).

While Ranking SVMs may just as standard SVMs be used with all kinds of kernels, a linear kernel has the advantage that weights of features can be directly extracted without computational effort. Bunescu and Pasca make use of this to automatically learn the threshold for a decision on NIL candidates. They have demonstrated that, using a linear kernel in the Ranking SVM, this threshold can be learned automatically from the weight of an indicative feature:

$$x_{nil}(m, e) = \mathbb{1}(e, \text{NIL}). \tag{3.19}$$

This binary feature is active only for a NIL candidate that needs to be provided for each mention in order to learn the threshold from the available features. We may therefore create candidate sets $\mathbf{e}(m) = \{e_i(m)\} \subset \boldsymbol{\mathcal{W}} \cup \{\text{NIL}\}$ that cover

all candidates in Wikipedia $\{e_i(m)\} \subset \boldsymbol{\mathcal{W}}$ and add for each mention an artificial candidate NIL.

To create training instances, we need to assign each training instance representing a mention-candidate pair a ranking. For our implementation of Bunescu and Pasca's method, we unsuccessfully tried to communicate with the authors on how these target rankings are created for the training data. Since the paper does not indicate otherwise, we assume that the ranking used in Bunescu and Pasca [2006] is a weak ordering where the correct candidate is assigned the top position and all other candidates that do not represent the ground truth entity share a place in the ordering. In practice, this ordering is realised through real-valued scalars $y \in \mathbb{R}$. These are assigned to each vector $x(m, e_i)$ and a high value of y indicates a leading position in the ranking, a low value of y indicates a late position in the ranking. In our case, i.e. the case of a weak ordering, it suffices to chose a value $y \in \{-1, +1\}$. Then, for instance in the case of three candidates e_1, e_2 and e_3 for a mention m, we have

$$\begin{aligned} 1e_1 = e^+(m) : y(x(m, e_1)) &= +1 \\ e_2 \neq e^+(m) : y(x(m, e_2)) &= -1 \\ e_3 \neq e^+(m) : y(x(m, e_3)) &= -1 \end{aligned}$$

which puts $x(m, e_1)$ at the leading position and lets $x(m, e_2)$ and $x(m, e_3)$ share the same but lower position.

Having described the model designs of WTC and WCC and the learner used by Pilz and Paaß [2009] as well as by Bunescu and Pasca, we will now experimentally compare these approaches for person name disambiguation in German.

3.5.4 Evaluation

Before we start with the experimental evaluation, we first detail the employed performance measures and evaluation data. The literature has proposed different measures that we will also discuss later in this thesis. In this chapter, we will evaluate the proposed approach using micro and macro performance. These measures and the differences between them will be described next.

Micro and Macro Performance

Micro and macro performance are inspired by the performance evaluation in multi-class text classification (Yang [1999]) where classes are often not distributed equally in a dataset of interest. To account for this, micro and macro performance average performance either on the instance level or on the class level. Since in our scenario, classes correspond to the given set of ground truth entities, using these measures allows us to judge performance both for prominent as well as less prominent entities.

More specifically, micro performance, or per-mention performance, gives equal weight to each mention and averages performance over all mention-entity pairs, i.e. on the instance level. Macro performance averages performance on the class level (here per-entity level) and thus gives equal weight to each ground truth entity, regardless of its frequency. Using micro and macro performance for evaluation, we aim to avoid misinterpretation of results when some dominant entities are always predicted correctly and entities with few examples are not.

Both micro and macro performance use the performance indicators *Precision*, *Recall* and *F-Measure*. These indicators are computed over *true positive*, *false negative* and *false positive* assignments. For the application of these measures in entity linking and the computation of the necessary quantities, we replace classes with ground truth targets and define micro and macro performance as follows.

Let $e^+(m)$ be the ground truth target for a mention m and $\hat{e}(m)$ be the predicted target. If the prediction of a model is correct, i.e. $\hat{e}(m) = e^+(m)$, we have a *true positive* (*tp*):

$$tp(e^+(m)) = \begin{cases} 1, \text{if } e^+(m) = \hat{e}(m) \\ 0, \text{else.} \end{cases} \tag{3.20}$$

If the prediction is not correct, i.e. $\hat{e}(m) \neq e^+(m)$, we have a *false negative* (*fn*) for $e^+(m)$

$$fn(e^+(m)) = \begin{cases} 1, \text{if } e^+(m) \neq \hat{e}(m) \\ 0, \text{else.} \end{cases} \tag{3.21}$$

Analogously, we have a *false positive* (*fp*) for $\hat{e}(m)$

$$fp(\hat{e}(m)) = \begin{cases} 1, \text{if } e^+(m) \neq \hat{e}(m) \\ 0, \text{else.} \end{cases} \tag{3.22}$$

Now, assume a collection of mentions $\mathbf{M} = \{m_i\}_{i=1}^{|\mathbf{M}|}$ with associated ground truth entities $\boldsymbol{\mathcal{E}}_{\mathbf{M}} = \{e^+(m_i)\}_{i=1}^{|\mathbf{M}|}$. Micro performance first computes the total number of true positives (TP), false positives (FP) and false negatives (FN) over all mention instances:

$$\text{TP} = \sum_{m_i \in \mathbf{M}} tp(e^+(m_i)),\ \text{FN} = \sum_{m_i \in \mathbf{M}} fn(e^+(m_i)),\ \text{FP} = \sum_{m_i \in \mathbf{M}} fp(e^+(m_i)). \tag{3.23}$$

Then, Precision (P_{micro}), Recall (R_{micro}) and F-measure (F_{micro}) are computed independently of the underlying entity:

$$P_{micro} = \frac{\text{TP}}{\text{FP} + \text{TP}}, \quad R_{micro} = \frac{\text{TP}}{\text{FN} + \text{TP}}, \quad F_{micro} = \frac{2 \cdot P_{micro} \cdot R_{micro}}{P_{micro} + R_{micro}}. \tag{3.24}$$

In contrast to micro performance, macro performance first computes Precision (P_{macro}), Recall (R_{macro}) and F-measure (F_{macro}) separately for each entity e in the

ground truth set $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$

$$P_{macro}(e) = \sum_{\substack{m_i \in \mathbf{M} \\ e^+(m_i)=e}} \frac{tp(e^+(m_i))}{fp(e^+(m_i)) + tp(e^+(m_i))}, \tag{3.25}$$

$$R_{macro}(e) = \sum_{\substack{m_i \in \mathbf{M} \\ e^+(m_i)=e}} \frac{tp(e^+(m_i))}{tp(e^+(m_i)) + fn(e^+(m_i))}, \tag{3.26}$$

$$F_{macro}(e) = \sum_{\substack{m_i \in \mathbf{M} \\ e^+(m_i)=e}} \frac{2 \cdot P_{macro}(e) \cdot R_{macro}(e)}{P_{macro}(e) + R_{macro}(e)}. \tag{3.27}$$

Then, these values are averaged over all ground truth entities $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$

$$P_{macro} = \frac{1}{|\boldsymbol{\mathcal{E}}_{\mathbf{M}}|} \sum_{e \in \boldsymbol{\mathcal{E}}_{\mathbf{M}}} P_{macro}(e), \tag{3.28}$$

$$R_{macro} = \frac{1}{|\boldsymbol{\mathcal{E}}_{\mathbf{M}}|} \sum_{e \in \boldsymbol{\mathcal{E}}_{\mathbf{M}}} R_{macro}(e), \tag{3.29}$$

$$F_{macro} = \frac{1}{|\boldsymbol{\mathcal{E}}_{\mathbf{M}}|} \sum_{e \in \boldsymbol{\mathcal{E}}_{\mathbf{M}}} F_{macro}(e) \tag{3.30}$$

where $|\boldsymbol{\mathcal{E}}_{\mathbf{M}}|$ is the number of distinct entities in $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$. Note that due to the averaging over ground truth targets $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$, F_{macro} is here not to be interpreted as the harmonic mean of Precision and Recall as is the case for F_{micro}.

Both measures are computed over the ground truth entities $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$ and not over all possible predictions, which are technically all entities in Wikipedia. The recall for entities not contained in the ground truth collection $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$ can not be computed in a meaningful way as there are no positive examples for them. Consequently, if a model prediction $\hat{e}(m)$ is not contained in the ground truth $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$, this is counted as a false negative for the ground truth target $e^+(m)$ but not as a false positive for the predicted entity $\hat{e}(m)$. Computing micro precision with respect to all possible targets would be analogous to standard accuracy, since then the number of false positives is equal to the number of false negatives.

We give the following example to illustrate the computation and the aforementioned differences between micro and macro entity performance.

Example 7 (Micro and Macro Performance)
Consider a collection of mentions $\mathbf{M} = \{m_1, m_2, m_3, m_4\}$ with associated ground truth entities $\boldsymbol{\mathcal{E}}_{\mathbf{M}} = \{e^+(m_1) = e_1, e^+(m_2) = e_1, e^+(m_e) = e_1, e^+(m_4) = e_2\}$. If all mentions are linked correctly to their underlying entities, we have

$$tp(e_1) = 3, fp(e_1) = fn(e_1) = 0 \text{ and } tp(e_2) = 1, fp(e_2) = 0, fn(e_2) = 1.$$

This results in micro and macro performance values of

$$P_{micro} = R_{micro} = F_{micro} = 1 \text{ and } P_{macro} = R_{macro} = F_{macro} = 1.$$

If now all mentions were linked to e_1, which means an erroneous link of m_4 to e_1, we would have

$$tp(e_1) = 3, fp(e_1) = 1, fn(e_1) = 0 \text{ and } tp(e_2) = fp(e_2) = 0, fn(e_2) = 1.$$

According to Eq. 3.23 and Eq. 3.24, this results in micro performance measures of

$$P_{micro} = \tfrac{3}{4}, R_{micro} = \tfrac{3}{4}, F_{micro} = \tfrac{3}{4},$$

whereas following Eq. 3.25 to Eq. 3.30 the according macro performance values are notably lower:

$$P_{macro} = \tfrac{1}{2} \cdot \left(\tfrac{3}{4} + 0\right) = \tfrac{3}{8}, R_{macro} = \tfrac{1}{2} \cdot \left(\tfrac{3}{3} + 0\right) = \tfrac{1}{2}, F_{macro} = \tfrac{1}{2} \cdot \left(\tfrac{6}{7} + 0\right) = \tfrac{3}{7}.$$

As micro performance gives all instances the same weight, the high number of correctly disambiguated mentions has more impact and we obtain a notably higher micro performance. Macro performance however clearly states that only half of the ground truth entities were correctly retrieved.

Further, to illustrate that performance is computed only over entities in $\boldsymbol{\mathcal{E}}_{\mathbf{M}}$, let us assume a prediction $\hat{e}(m_4) = e_3 \notin \boldsymbol{\mathcal{E}}_{\mathbf{M}}$. This prediction is counted as a false negative for e_2 and not as a false positive for e_3, and consequently we would have $R_{micro} = \frac{3}{4}$ and $P_{micro} = 1$. The corresponding values in macro performance are $R_{macro} = P_{macro} = \frac{1}{2}$.

To summarize, a low macro performance may induce that some mentions are more difficult to link than others or that an imbalance of ground truth targets has a negative impact on the model performance. But in the ideal case, macro performance should be close to micro performance.

Since we aim at representative models independent of the frequency of an entity, the usage of micro and macro performance is also mirrored in our dataset creation strategy. To avoid dominances and obtain diverse datasets, we used upper bounds on the number of examples per entity, but refrained from lower bounds on the minimum number of examples. The following section will describe dataset creation in detail.

Training and Evaluation Data from Wikipedia

At the time of publication, there was no dataset publicly available for person name disambiguation in German. Thus, inspired by Bunescu and Pasca [2006] we exploit Wikipedia's link structure to extract datasets of disambiguated entity mentions. As

described in Section 2.4, each Wikipedia link $l \in \mathbf{L}$ has an anchor text l_a corresponding to a mention m, and a link target l_t providing the referenced entity as ground truth assignment $e^+(m) = l_t$. Assuming the correctness of links in Wikipedia, this property allows the extraction of disambiguated datasets from Wikipedia references. This extraction is depicted in Alg. 1 and described in detail in the following.

First, when using Wikipedia for training and evaluation, we need to separate target entities from example contexts. Therefore we store a subset of Wikipedia entities in a *candidate pool* $\boldsymbol{\mathcal{W}}_c \subset \boldsymbol{\mathcal{W}}$ and use this pool as collection of target entities. The remaining articles in Wikipedia may then serve as resources for example contexts for the entities in $\boldsymbol{\mathcal{W}}_c$. This procedure ensures a clean separation among entities and example contexts. Furthermore, it also gives us control over the characteristics of the mentions we want to analyse. Depending on the focus of the entity linking model, we may generate candidate pools differently. For instance, if we focus on the disambiguation of person names, we may create a candidate pool containing only persons. We may also choose as candidate pool the subset of persons with ambiguous names, leaving out the majority of persons with unique name and therefore focusing on potentially more difficult tasks.

More specifically, the subset $\boldsymbol{\mathcal{W}}_c \subset \boldsymbol{\mathcal{W}}$ is the pool of entities for which we collect inlinks to extract disambiguated examples. For each entity $e_i \in \boldsymbol{\mathcal{W}}_c$, we extract a number of link sources $l_s \in \mathbf{L}_{in}(e_i)$ containing a reference (a link) to e_i and use these link sources to create example documents $\mathbf{D}_{e_i}$:

$$\mathbf{D}_{e_i} = \{l_s \in \mathbf{L}_{in}(e_i) \mid l_s \neq e_i, l_s \notin \boldsymbol{\mathcal{W}}_c\}\,. \tag{3.31}$$

Collecting grounded examples in this manner for all entities $e_i \in \boldsymbol{\mathcal{W}}_c$ results in a collection $\mathbf{D}$ which can be used to train and evaluate a linking model. Each document $d \in \mathbf{D}_{e_i}$ constitutes one example context of a mention of e_i. Even though the source l_s may contain other outlinks and thus other mentions, these are not considered and we treat only the mention for e_i in $d \in \mathbf{D}_{e_i}$. Therefore, the number of documents in $\mathbf{D}$ is equal to the number of mentions we evaluate. Through the link target l_t that is associated with the mention, each example is grounded with the true entity $e^+(m) = e_i$. The text of the example document d can be either the complete article $text(l_s)$ or a restricted window around the mention anchor.

Note that the constraint $l_s \notin \boldsymbol{\mathcal{W}}_c$ in Eq. 3.31 is necessary to avoid the mixture of example documents and candidate entities. For example, assume that an entity e_j is contained as link source in $\mathbf{L}_{in}(e_i)$ and therefore provides an example context for e_i. If on the other hand e_j would also be used as a candidate entity, there would be no clear distinction of example contexts and target entities and we would mix up knowledge base and input documents.

An additional aspect to be considered is the discrepancy in the number of examples per entity. Not all entities in $\boldsymbol{\mathcal{W}}_c$ need to have inlinks and in contrast some entities, especially popular ones, may have a very large number of inlinks. For instance, in

Algorithm 1: Extracting disambiguated examples from Wikipedia references

Input: candidate pool $\boldsymbol{\mathcal{W}}_c$, maximum number of examples per entity n, ratio of uncovered entities z.

Output: examples $\mathbf{D}$, adapted candidate pool $\boldsymbol{\mathcal{W}}_c$.

1 **for** $e_i \in \boldsymbol{\mathcal{W}}_c$ **do**
2 isNIL $\leftarrow$ false
3 $\mathbf{D}_{e_i} \leftarrow \emptyset$
4 **if** $i \pmod z = 0$ **then** `// mark every` z`th` $e \in \boldsymbol{\mathcal{W}}_c$ `as NIL`
5 isNIL $\leftarrow$ true
6 $\boldsymbol{\mathcal{W}}_c \leftarrow \boldsymbol{\mathcal{W}}_c \setminus \{e_i\}$ `// remove` e_i `from the candidate pool`
7 **while** $|\mathbf{D}_{e_i}| \leq n$ **do** `// collect at most` n `example references for` e_i
8 **for** $l_s \in \mathbf{L}_{in}(e_i)$ **do**
9 **if** $l_s \neq e$ ***and*** $l_s \notin \boldsymbol{\mathcal{W}}_c$ **then**
10 **if** isNIL **then** `// re-target` l_t `to NIL`
11 $l_t \leftarrow$ NIL
12 $d \leftarrow (text(l_s, l_a, l_t))$
13 $\mathbf{D}_{e_i} \leftarrow \mathbf{D}_{e_i} \cup \{d\}$
14 **return** $\mathbf{D} = \bigcup_{e_i} \mathbf{D}_{e_i}$, $\boldsymbol{\mathcal{W}}_c$

the English Version of Wikipedia, we observed that the number of inlinks may range from 1 to more than 250.000, the latter observed for the very popular entity United States. In such cases, a high model accuracy is achievable when all examples of the ambiguous name are linked against the popular entity since the few examples of the other, less popular entities have only minor influence on model accuracy. To avoid such pitfalls, we set a boundary on the number of examples per entity and use at most n randomly selected inlinks from the set $\mathbf{L}_{in}(e_i)$ as examples (line 7 in Alg. 1). Also, using all inlinks of an entity would result in a strong overlap of examples and entities in the candidate pool due to the strong interconnectivity of Wikipedia articles.

We simulate examples of uncovered entity mentions by marking every z-th entity in the candidate pool $\boldsymbol{\mathcal{W}}_c$ as NIL (line 4 ff. and line 10 ff. in Alg. 1). For example, a value of 5 for z means that 20% of the candidate entities in $\boldsymbol{\mathcal{W}}_c$ will be marked as NIL and therefore be removed from the candidate pool. Since the ground truth entity of the according link anchor text is changed to NIL, all examples of these entities are then examples for mentions of uncovered entities. This adaptable ratio is necessary to account for uncovered entities that will emerge frequently in non-Wikipedia texts such as newspaper articles.

Experiments

In Pilz and Paaß [2009], we compared word-topic correlation (WTC) and word-category correlation (WCC) for the disambiguation of German name phrases denoting persons. To create an evaluation corpus, we collect a set N of 500 ambiguous name phrases collectively corresponding to 1072 persons in the German Wikipedia. Here, the candidate pool $\boldsymbol{\mathcal{W}}_c \subset \boldsymbol{\mathcal{W}}$ consists of

$$\boldsymbol{\mathcal{W}}_c = \{e \in \boldsymbol{\mathcal{W}} \mid name(e) \in N\} \tag{3.32}$$

and contains all entities in $\boldsymbol{\mathcal{W}}$ whose name is contained in the list N. We used a simple candidate selection based on exact matches between entity names $name(e)$ and elements in N[1].

To obtain training and evaluation data, we extract the entity's references using the extraction scheme described in Alg. 1. Using $\boldsymbol{\mathcal{W}}_c$ as given in Eq. 3.32 and $n = 10$, we obtain 6513 disambiguated example contexts, each representing the context for one mention of an entity. We simulate uncovered entities by removing the true underlying entity from the candidate set for 10% of the extracted mention contexts. This is realized through a value of $z = 10$ in Alg. 1. The context of a mention is a window of 50 words around the mention, the context of a candidate entity is formed from the first 100 words appearing in its article text. We compare the WTC model to the cWCC approach (Eq. 3.6) on this dataset using 5441 of the above mention context for training and 1072 for testing. The test set contains 970 examples for covered entities and 102 examples for uncovered entity mentions.

The topic model used for WTC is the same as in Fig. 3.4, i.e. trained over 100k Wikipedia articles describing persons with $K = 200$, which we considered appropriate given the number of training articles. An empirical analysis of models with higher or lower granularity in topics revealed more volatile or less expressive topic clusters. Even though Wallach et al. [2009] basically state 'the more topics the better', we could not confirm this for our task.

We follow Bunescu and Pasca to learn a threshold for the detection of uncovered entities. We augment a mention's candidate set with a candidate representing NIL and represent the NIL-candidate by a vector that contains only the NIL-feature as in Eq. 3.19. We use a Ranking SVM to determine the right matching entity and to detect uncovered entities. The decision threshold is learned from the weight of the NIL-feature in a linear kernel.

For the implementation of cWCC, we need to extract categories from the German Wikipedia. Even if we could obtain the same categories as in Bunescu and Pasca [2006][2] we can not thoroughly align them with the German version. Further, the analyses of Wikipedia's category hierarchy is not a trivial task, as we

[1] More elaborate candidate selection methods are developed in Chapter 4.
[2] Bunescu and Pasca [2006] used the Wikipedia version from May 2005.

can encounter loops and other inconsistencies. Therefore, instead of analysing the category hierarchy to extract top-level categories, we used the categories that can be extracted by parsing the text of the Wikipedia article. These directly assigned categories are filtered with the same requirement regarding the minimum number of articles assigned to that category. We found 16201 different categories for the 198903 Wikipedia articles describing persons. Neglecting the 3996 categories that hold year of birth and year of death information, 12205 categories remain. From these, 2377 affect only one person. We are aware that we use by far more categories than the only 540 categories employed in Bunescu and Pasca [2006]. However, this is likely to result in more specific attributes with even more discriminative power. Also, while using more features is also unfair for the comparison with our method, we argue that this advantage is levelled by the less stringent semantic coherence among these categories. When referring to cWCC resp. WCC later on, we always mean this implementation with associated category selection scheme (dependent on the language version).

Interestingly, we obtained very similar results for both methods on the dataset described above. WTC achieves an F_{micro} of 97.76% resp. an F_{macro} of 96.70% which is very close to the result for cWCC with an F_{micro} of 98.60% resp. an F_{macro} of 98.10%. Also, we found that all entities simulated as uncovered were correctly linked to NIL. However, while the absolute difference between the different approaches is with 1 resp. 1.4 points in percentage very low, we should point out that we have a far lower error rate for cWCC. We assume that this is because cWCC has a very high dimensional and sparse feature vector representation that is prone to yield a clearer separability. While for WTC we used only 200 topics, there were more than 4000 categories available for cWCC. Consequently, the respective feature spaces differ notably in dimensionality and the maximum dimension of WTC is only one-twentieth of the maximum dimension for cWCC.

We also note that the results obtained for our implementation of Bunescu and Pasca's method are notably higher compared to the results originally published in Bunescu and Pasca [2006]. Clearly, we should not directly compare these figures since different datasets were used. However, we want to point out that for their implementation, Bunescu and Pasca reduced the number of categories, with the effect that more persons share categories and hence categories may be less distinctive. Apart from the difference in the dataset, this may be a reason for the notably higher performance obtained in our experiments compared to originally published accuracy of 84.8%.

There are some observations for the employed dataset we find worth noting. We manually investigated the model predictions, and, similar to the observations made in Cucerzan [2007], we observed links that were disambiguated correctly by our model but counted as errors since the ground truth annotation was incorrect. For instance, we found that the human annotators mixed up the two entities denoted by the name John Barber, e.g. the inventor of the gas turbine and an English

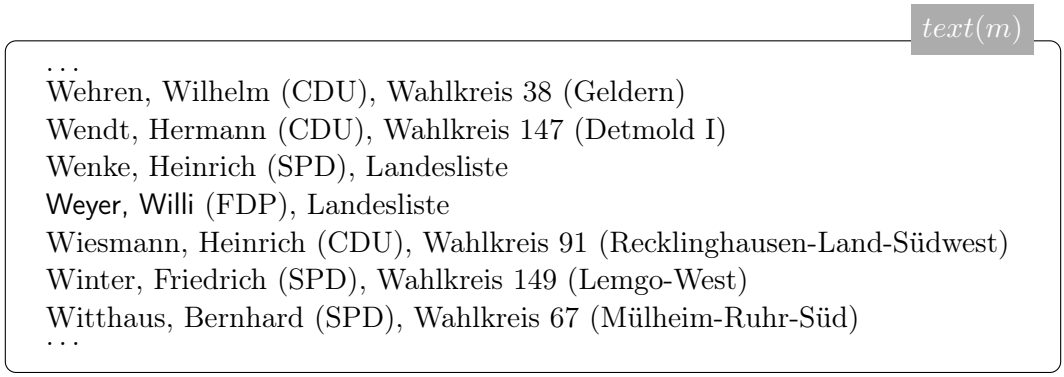
race driver, whereas the disambiguation model identified them correctly. Thus, unfortunately, we see that the assumption of correct links does not hold in general. Moreover, we found that the uniqueness of entity pages was not guaranteed: we observed two distinct articles Jens Jessen and Jens Jessen (Ökonom) describing the same entity. These two examples show that Wikipedia is not perfect for evaluation. However, it is unlikely to observe perfect inter-annotator agreement on other datasets. And, most importantly, Wikipedia is still the only source providing disambiguated examples in that quantity and multilingualism and we assume that the number of correct links easily surpasses the number of incorrect links.

In the last section, we discussed an entity linking model based on topic modelling and topic probability distributions $\mathcal{T}_e$ over candidate entity contexts. We have empirically shown that we do not need to rely on manually assigned Wikipedia categories, and that by replacing these categories with semantic information from topics we obtain comparably good results. Since expensive manual categorization is not required, our WTC model can thus potentially be applied to link entity mentions also to other textual knowledge bases that are not endowed with categorization.

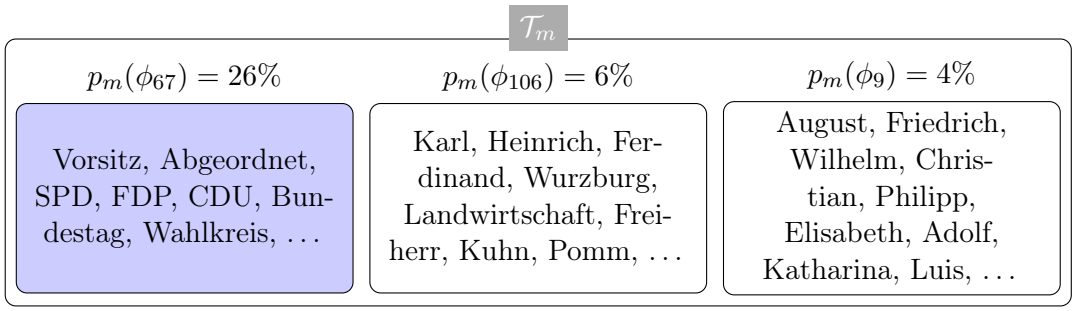
However, we can go a step further by observing that the WTC formulation did not exploit all of the available information. We used a restricted set of terms appearing jointly in the mention and entity context for the correlation with an entity's topics in order to learn a semantic overlap. Now, alternative terms may be used in mention and entity context to describe the same entity, but these terms would not be considered in WTC. To overcome this, we may infer topic distributions not only on an entity's context but also on a mention context and compare these two distributions directly.

To illustrate our motivation, Fig. 3.6 shows a context referring to the politician Willi Weyer (Politiker). This context is taken from a Wikipedia article on delegates in a German federal state. Note that even though the context is not a typical natural language text but a list-like enumeration, we may use LDA to infer a topic distribution $\mathcal{T}_m$ since LDA is as such independent of the text's structure. Fig. 3.6b summarizes the three topics with highest probability for the given context. The high probability of ϕ_{65} indicates a political topic in the context which clearly hints at the true underlying entity $e^+(m) =$ Willi Weyer (Politiker) and not at Willi Weyer (soccer player). Recall Fig. 3.4: the most prominent topic in $\mathcal{T}_e$ is the same for $e =$ Willi Weyer (Politiker).

Before we formulate thematic distances over mention and entity contexts, we should give some general remarks on how the characteristics of context and training corpus influence the topics inferred by LDA. First, the length of a context has direct impact on the inferred topic probability distribution. When estimating the topic distribution for a context, we sample a topic for each of the words in the context given all the topic assignments in the training corpus. Short contexts containing many terms related to one topic are prone to be assigned to one dominant topic with less probability mass on other topics. This effect can be observed for the context



(a) A context with a mention m with $name(m) =$ Weyer, Willi and $e^+(m) =$ Willi Weyer (Politiker).



(b) Topics for the mention context in 3.6a.

Figure 3.6: 3.6a shows a context mentioning Willi Weyer (Politiker), 3.6b shows the three most probable topics from the topic distribution $\mathcal{T}_m$ for this context. For each topic, we give the probability $p_m(\phi_k)$ and its most important words. The topic ϕ_{67} (shaded in blue) is also the most prominent topic in the topic distribution $\mathcal{T}_e$ for the article text of Willi Weyer (Politiker).

in Fig. 3.6a that contains many terms related to a political subject, e.g. political parties (*SPD*, *CDU*) or electoral constituencies (*Wahlkreis*). The abundance of these terms influences the sampling process towards topic clusters with political terms. Accordingly, we find that the probability of the most dominant topic ϕ_{67} is with $p_m(\phi_{67}) = 26\%$ about five times higher than that of the second most likely topic ϕ_{106} with $p_m(\phi_{106}) = 6\%$, a topic containing mostly person names.

Such *name topics* are typical for the nature of the training data. Name topics are not very informative for our task but will appear in most topic models trained over documents containing references of persons. Similar to news stories, articles in Wikipedia mention persons in relation to specific subjects. Consequently, we observe that names of politicians are associated with political topics and names of soccer players with sports topics. On the other hand, words that are equally distributed over the document collection do not exhibit specific co-occurrence schemes. For

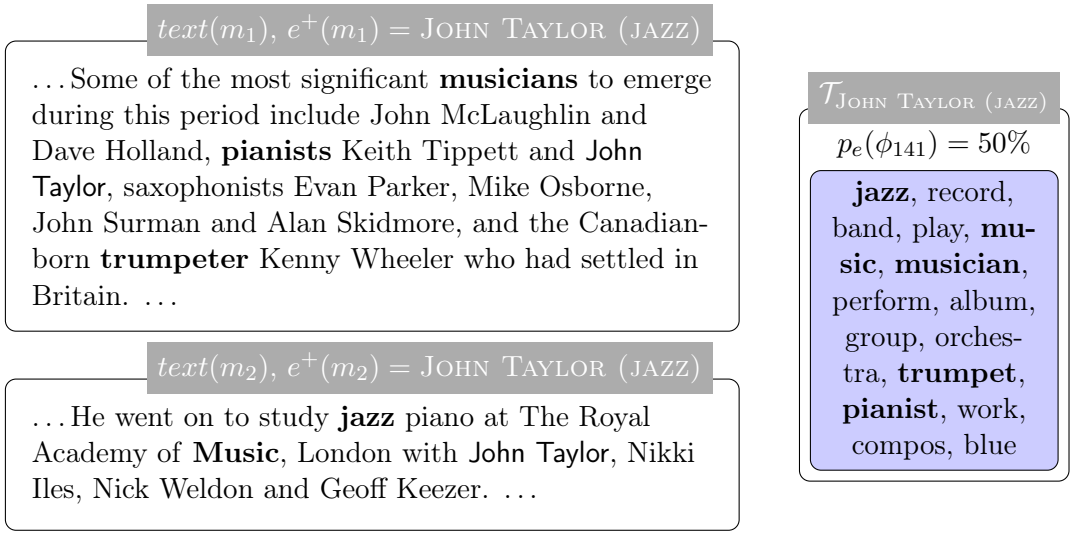


Figure 3.7: Two Wikipedia contexts mentioning the entity John Taylor (jazz) (left) and the entity's most important topic (right). While the word overlap between the two contexts is low, both contexts share terms (in bold) with the topic ϕ_{141} that has a high probability $p_e(\phi_{141})$ of 50% for the article text of John Taylor (jazz).

example, person names occurring across many diverse contexts will be clustered into name topics consisting mostly of first names and surnames. Similarly, function words such as stop words will be clustered into function word topics. As in the example context in Fig. 3.6a person names make up a high portion of the text, we also observe with ϕ_{106} and ϕ_9 two name topics for this context.

3.6 Thematic Context Distance

The comparison of word vectors over describing contexts has been quite successful in the literature. Thus, many approaches to name disambiguation estimate the identity of a mention by comparing its context with the words in the description of candidate entities. However, the performance of such a method is negatively affected when different words with similar meaning are used in the respective contexts. As indicated by the two contexts mentioning John Taylor (Jazz) in Fig. 3.7, contextual overlap based on words may be low even though the thematic subject is very similar and both contexts refer to the same person. Note that we can directly see the overlap of key terms for a musician when comparing the words in ϕ_{141} for John Taylor (Jazz) and the words in $text(m)$ but also that these terms indicate a specific genre, i.e. jazz. Topic models automatically disambiguate terms based on the co-occurrences with other terms.

The most frequently employed technique for word vector comparison is cosine

similarity, which was also used as baseline feature by Bunescu and Pasca [2006] to evaluate the similarity of mention and candidate context. As defined in Eq. 3.1, cosine similarity is a summation over common words, i.e. terms appearing simultaneously in two contexts, that is normalized through the respective contexts' lengths. Intuitively speaking, the larger this number the more similar the describing contexts and hence the more similar the entities denoted. However, cosine similarity summarizes all context information into one indicating scalar. While proven to be a powerful indicator in the literature, this aggregation can not handle world knowledge that may be expressed not explicitly but implicitly through semantic relations. A direct comparison over word vectors, even in a stemmed form, may fail to assign both contexts in Fig. 3.7 to JOHN TAYLOR (JAZZ), as such a method can not grasp that the other co-occurring persons also have a latent but strong relation to music. Therefore, we propose a new distance measure formulation where words are embedded into topics and distance is calculated over topic distributions instead of words. We will describe different candidates for this thematic context distances in the following.

3.6.1 Measures for Thematic Context Distance

To generalize from word based comparison, we propose to measure the *thematic context distance* between between mention and entity context to identify the underlying entity of a mention. To measure the thematic context distance for a mention-entity pair (m, e_i), we need to compare the topic probability distribution $\mathcal{T}_m$ over the mention context $text(m)$ with the topic distributions $\mathcal{T}_{e_i}$ over the candidate entity article text $text(e_i)$ for all candidates $e_i \in \mathbf{e}(m)$.

The topic modelling literature has evaluated a number of divergence measures for probability distributions with different weighting schemes. Here, we focus on their formulations as distances, i.e. the symmetric versions of divergence measures. We describe three popular distribution distance measures that, applied to topic probability distributions are here interpreted as thematic distances with respect to a given topic model. We evaluate each of these distances for their individual performance for the task of entity disambiguation (see Fig. 3.8 and Fig. 3.9).

Being very popular in the topic modelling literature, the first measure we describe is the Kullback-Leibler divergence. The Kullback-Leibler divergence was introduced by Kullback and Leibler [1951] as a divergence measure or relative entropy that for two probability distributions $\mathcal{T}_m$ and $\mathcal{T}_e$ is given by

$$\mathrm{KL}\left(\mathcal{T}_e, \mathcal{T}_m\right) = \sum_{k=1}^{K} p_e(\phi_k) \log\left(\frac{p_e(\phi_k)}{p_m(\phi_k)}\right) \in [0, \infty) \tag{3.33}$$

where $p_e(\phi_k)$ is the probability of topic ϕ_k in the context of entity e and $p_m(\phi_k)$ the probability of topic ϕ_k in the context of mention m. The Kullback-Leibler divergence

has a range of $[0, \infty)$ and is not symmetric. For all cases where $\mathcal{T}_e \neq \mathcal{T}_m$, we have $\mathrm{KL}(\mathcal{T}_e, \mathcal{T}_m) \neq \mathrm{KL}(\mathcal{T}_m, \mathcal{T}_e)$.

The symmetric version of Kullback-Leibler divergence is given by

$$\mathrm{sKL}(\mathcal{T}_e, \mathcal{T}_m) = \frac{1}{2} \sum_{k=1}^{K} \left(p_m(\phi_k) \log \left(\frac{p_m(\phi_k)}{p_e(\phi_k)} \right) + p_e(\phi_k) \log \left(\frac{p_e(\phi_k)}{p_m(\phi_k)} \right) \right) \in [0, \infty) \tag{3.34}$$

and yields $\mathrm{sKL}(\mathcal{T}_e, \mathcal{T}_m) = \mathrm{sKL}(\mathcal{T}_m, \mathcal{T}_e)$. Now, in our case, as the contexts $text(e)$ and $text(m)$ are interchangeable with respect to similarity, we assume the symmetric formulation to provide more interpretability.

Another distance measure is the Jensen-Shannon distance after Lin [1991]. This measure is similar to the symmetric Kullback-Leibler divergence in Eq. 3.34 but uses the average of two probabilities $r_k = 0.5 \cdot (p_m(\phi_k) + p_e(\phi_k))$ as denominator:

$$\mathrm{JS}(\mathcal{T}_e, \mathcal{T}_m) = \frac{1}{2} \sum_{k=1}^{K} \left(p_m(\phi_k) \log \left(\frac{p_m(\phi_k)}{r_k} \right) + p_e(\phi_k) \log \left(\frac{p_e(\phi_k)}{r_k} \right) \right) \in [0, 1]. \tag{3.35}$$

The bound of $0 \leq \mathrm{JS}(\mathcal{T}_e, \mathcal{T}_m) \leq 1$ holds, if the logarithm in Eq. 3.35 is to the base 2. If replaced with the natural logarithm, the upper bound is reduced to $\ln(2)$.

Blei and Lafferty [2009] proposed an adapted form of the Hellinger distance as another alternative measure for the similarity of probability distributions:

$$\mathrm{H}(\mathcal{T}_e, \mathcal{T}_m) = \sum_{k=1}^{K} \left(\sqrt{p_m(\phi_k)} - \sqrt{p_e(\phi_k)} \right)^2 \in [0, 1]. \tag{3.36}$$

This variant of the Hellinger distance has bounds $0 \leq \mathrm{H}(\mathcal{T}_e, \mathcal{T}_m) \leq 1$ and the upper bound of 1 is obtained when $p_m(\phi_k)$ assigns a probability value of zero to every event to which $p_e(\phi_k)$ assigns a positive probability, and vice versa.

All of the above distances and divergences may be used as a single scalar. However, the representation in a single scalar ignores a lot of information and entities appearing in similar contexts can be difficult to distinguish using such an aggregated measure. Therefore, instead of summing differences in probability values to a single value, we propose to use each difference term separately as a distinct feature. This allows for a better separability of the resulting data points and furthermore a classifier may evaluate correlations between these terms or learn weights individually for each thematic distance value. Thus, for the distances introduced above, we will

create one distinct distance term per topic index $k = 1, \dots K$, i.e.

$$\forall k = 1, \dots, K :$$

$$\mathrm{KL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_k = p_e(\phi_k) \log\left(\frac{p_e(\phi_k)}{p_m(\phi_k)}\right) \tag{3.37}$$

$$\mathrm{sKL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_k = \frac{1}{2}\left(p_m(\phi_k) \log\left(\frac{p_m(\phi_k)}{p_e(\phi_k)}\right) + p_e(\phi_k) \log\left(\frac{p_e(\phi_k)}{p_m(\phi_k)}\right)\right) \tag{3.38}$$

$$\mathrm{H}\left(\mathcal{T}_e, \mathcal{T}_m\right)_k = \left(\sqrt{p_m(\phi_k)} - \sqrt{p_e(\phi_k)}\right)^2 \tag{3.39}$$

$$\mathrm{JS}\left(\mathcal{T}_e, \mathcal{T}_m\right)_k = \frac{1}{2}\left(p_m(\phi_k) \log\left(\frac{p_m(\phi_k)}{r_k}\right) + p_e(\phi_k) \log\left(\frac{p_e(\phi_k)}{r_k}\right)\right). \tag{3.40}$$

As in Eqs. 3.33 to 3.36, $p_e(\phi_k)$ is the probability of topic ϕ_k in the context of entity e, $p_m(\phi_k)$ is the probability of topic ϕ_k in the context of mention m and $r_k = 0.5(p_m(\phi_k) + p_e(\phi_k))$ in Eq. 3.40.

Now, to experimentally find the best distance representation for entity linking, we evaluate all of the above distances with different kernels using an SVM classifier from SVM$^{\text{Light}}$ with standard parameters (the results are given in Section 3.6.4). To do so, we create for each thematic distance a feature vector $D_{(\cdot)}(\mathcal{T}_m, \mathcal{T}_e)$. To distinguish among the employed distance measure, we use as subscript the name of the distance and then have:

$$D_{\mathrm{KL}}(\mathcal{T}_m, \mathcal{T}_e) = \left[\mathrm{KL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_1, \dots, \mathrm{KL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_K\right] \in [0.01, 1]^K \tag{3.41}$$

$$D_{\mathrm{sKL}}(\mathcal{T}_m, \mathcal{T}_e) = \left[\mathrm{sKL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_1, \dots, \mathrm{sKL}\left(\mathcal{T}_e, \mathcal{T}_m\right)_K\right] \in [0.01, 1]^K \tag{3.42}$$

$$D_{\mathrm{JS}}(\mathcal{T}_m, \mathcal{T}_e) = \left[\mathrm{JS}\left(\mathcal{T}_e, \mathcal{T}_m\right)_1, \dots, \mathrm{JS}\left(\mathcal{T}_e, \mathcal{T}_m\right)_K\right] \in [0.01, 1]^K \tag{3.43}$$

$$D_{\mathrm{H}}(\mathcal{T}_m, \mathcal{T}_e) = \left[\mathrm{H}\left(\mathcal{T}_e, \mathcal{T}_m\right)_1, \dots, \mathrm{H}\left(\mathcal{T}_e, \mathcal{T}_m\right)_K\right] \in [0.01, 1]^K \tag{3.44}$$

The elements of the vectors $D_{(\cdot)}(\mathcal{T}_m, \mathcal{T}_e)$ in Eqs. 3.41 to 3.44 are computed according to Eqs. 3.37 to 3.40 respectively. As each of these feature vector representations computes distance terms between corresponding topic probabilities explicitly, we have a maximum dimension of K, according to the number of topics in the underlying topic model. So, technically, the range of each $D_{(\cdot)}(\mathcal{T}_m, \mathcal{T}_e)$ is $[0, 1]^K$. However, we here ignore a feature if both $p_m(\phi_k)$ and $p_c(\phi_k)$ are less than 0.01 and thus clamp the range of each $D_{(\cdot)}(\mathcal{T}_m, \mathcal{T}_e)$ to $[0.01, 1]^K$. This form of feature selection is based on the assumption that we don't need to spend modelling effort for unimportant topics since we don't give too much weight on the long tail of unimportant topics. It also has the side effect that the overall number of non-sparse features will be rather low, which speeds up the kernel computation in the SVM classifier.

In Pilz and Paaß [2011] we also evaluated a linear concatenation of the two probability distributions $\mathcal{T}_m$ and $\mathcal{T}_e$, i.e. $D(m, e) = [\mathcal{T}_m, \mathcal{T}_e] \in [0, 1]^{2K}$. In this representation, only the part in $\mathcal{T}_e$ varies over a given set of candidates. This formulation

showed by far the weakest performance compared to the other distance measures, even in a quadratic kernel that can model the interactions between the topics for e and m. Therefore, we give no further attention to this formulation and omit the obtained results here.

In our experiments, we follow Bunescu and Pasca [2006] and use as baseline feature the cosine similarity $\cos(m, e)$ (cf. Eq. 3.1). This baseline feature is used to evaluate directly matching words in the contexts of e and m in all of the following experiments.

Application in Multiple Languages

As the underlying algorithm of topic models does not dependent on the language of the corpus, topic models can be trained on corpora of different languages without the need to modify this algorithm. Hence, we may use topic modelling for entity linking in multiple languages as long as unique entity descriptions are available. We will empirically show that our proposed method generalizes to other languages, as its application for entity linking to the English, the German, and the French version of Wikipedia, currently the three largest Wikipedia versions, shows quite similar performance figures for all of these languages.

We build distinct topic models with 200 topics for each of these languages. To create training corpora, we extract 100k random documents, mostly articles articles describing persons, from the respective versions of Wikipedia and use the resulting collections as training documents for LDA. More specifically the English topic model is build on articles derived from the English Wikipedia, the German topic model on articles derived from the German version and analogously the French topic model is trained on French Wikipedia articles. We are aware of the chance that the LDA training corpus may contain some of the entity descriptions in the candidate pools $\boldsymbol{\mathcal{W}}_c$ or example references used for training and evaluation. However, we argue that this is not too harmful, since this overlap will be very small due to random sampling. Furthermore, a strict distinction between these datasets will not produce significantly different results since we may infer topic distributions also when a document contains previously unseen words. In the very unlikely case that no word is known to the model, no context based similarity measure will work.

Apart from some language specific adaptions, we use the same pre-processing techniques for all languages. We extract the plain text and stem it using the appropriate language settings for the Porter stemmer and change the respective stop word lists. Having trained a model, we use it to infer the topic probability distribution $\mathcal{T}_m$ for mention contexts as well as the topic probability distribution $\mathcal{T}_{e_i}$ for candidate contexts.

In preliminary experiments, we evaluated topic models with different values of K and different training corpora for the task of entity linking. That is, we varied K from 50 to 500 and increased the number of documents in training corpus. We also evaluated different combinations of topic models and formulated concate-

nated topic distributions through the concatenation of distributions derived from different models, i.e. $\mathcal{T}_m = [\mathcal{T}_{m,LDA_1}, \ldots, \mathcal{T}_{m,LDA_k}]$. However, we found no major difference in predictive performance when increasing the number of topics above 200 or varying training corpora or topic distribution representation. Considering the hyper-parameters α and β, no additional evaluation is necessary since the Mallet implementation of LDA automatically optimizes these parameters. So even when explicitly using different initial values of α and β, the learned models are more or less the same and yield the same or very similar performance.

In Pilz and Paaß [2009] and Pilz and Paaß [2012], we used a Ranking SVM to learn a linking model. To learn a linking model based on thematic distances, we use a classification method based on a standard SVM. We will next detail how entity linking is formulated as a classification problem.

3.6.2 Linking as a Classification Problem

Candidate entities in Wikipedia can be considered as labels for a mention. This labelling can be learned in a supervised classification task using disambiguated examples retrieved from Wikipedia's interlinkage. Assume a candidate entity $e(m)$ for a mention m and a mention-candidate pairing operator $x(m, e(m))$ describing the mutual relation of m and e. In our case, the operator $x(m, e(m))$ is a vector of n real-valued features, i.e. $x(m, e(m)) \in \mathbb{R}^n$. For instance, one feature in this vector may be the cosine similarity of the two describing contexts $text(m)$ and $text(e(m))$. Now, as stated in Section 2.2, a candidate entity $e(m)$ is either correct or not. In a binary classification setting with labels $\{y_-, y_+\} = \{-1, +1\}$, a collection of training instances

$$\mathcal{D} = \left\{ \left(x_i^{(k)}(m_i, e_k(m_i)), y_i^{(k)} \right) \mid x_i^{(k)} \in \mathbb{R}^n, y_i^{(k)} \in \{y_-, y_+\}, e_k(m_i) \in \mathbf{e}(m_i) \right\} \quad (3.45)$$

then contains for any mention m_i and its k candidate entities $e_k(m_i) \in \mathbf{e}(m_i)$ a descriptive vector $x_i^{(k)}(m_i, e_k(m_i))$ that has an associated label $y_i^{(k)} \in \{y_-, y_+\}$, where the label y_+ denotes a positive instance and y_- a negative instance. A positive instance $(x(m, e(m)), y_+)$ encodes that e is the correct underlying entity for the mention m. Analogously, a negative instance $(x(m, e'(m)), y_-)$ encodes that e' is not the correct underlying entity for the mention m. Given these training instances, we may learn an assignment function $f : x(m, e(m)) \mapsto \{y_-, y_+\}$ of the form

$$f(x(m, e(m))) = \begin{cases} y_+, & \text{if } e(m) = e^+(m) \\ y_-, & \text{if } e(m) \neq e^+(m). \end{cases} \quad (3.46)$$

In the inference step, we use the prediction value of $f(x(m, e(m)))$ to decide on the estimated or predicted target entity $\hat{e}$:

$$\hat{e}(m) = e(m) \text{ if } f(x(m, e(m))) = y_+. \quad (3.47)$$

In general we will observe collections $e_k(m) \in \mathbf{e}(m) \subset \boldsymbol{\mathcal{W}}$ of candidate targets for a mention m. Therefore, the function f is applied for each mention-candidate pairing $x^{(k)}(m, e_k(m))$

$$\forall e_k(m) \in \mathbf{e}(m) : f(x^{(k)}(m, e_k(m))) = y^{(k)} \in \{y_-, y_+\}, \tag{3.48}$$

resulting in a set of tupels $(y^{(k)}, e_k)$ of prediction value $y^{(k)}$ and candidate $e_k(m)$. Now, to determine the final prediction, we need to assign the mention m to uniquely one candidate entity e. However, this uniqueness is not inherently guaranteed using standard prediction algorithms such as a binary SVM that predicts labels $\{y_-, y_+\}$. As an example, we may observe two candidate entities from a very similar field, i.e. politicians from the same party, where descriptions may vary only slightly. The resulting mention-candidate-pairings may then both receive a positive label y_+. To circumvent this problem, we use the real valued prediction $y \in \mathbb{R}$ of the SVM instead of the binary labels $\{y_-, y_+\}$, i.e.

$$y^{(k)} = w^* \cdot x^{(k)} - b. \tag{3.49}$$

This real-valued prediction $y^{(k)}$ is the offset of the instance $x^{(k)}$ from the separating hyperplane whose parameters w^* and b are learned from the training instances $\mathcal{D}$ (Eq. 3.45). Then, we define the predicted entity $\hat{e}(m)$ to be the candidate $e_k(m)$ for which we obtain the maximum prediction value $y^{(k)}$ for the mention-candidate-pairing $x^{(k)}(m, e_k(m))$. The final assignment is then

$$\hat{e}(m) = \underset{e_k \in \mathbf{e}(m)}{\arg\max}\, y^{(k)}. \tag{3.50}$$

With Eq. 3.50 we generalized the binary classification to a rank-related classification by choosing the candidate with highest score $y^{(k)}$ among all candidates. This enables an overall assignment model in contrast to an "one-model-per-entity" approach. We also evaluated a Ranking SVM but, as we will show in the empirical evaluation, the results obtained were inferior to those using a classification method. The SVM classifier basically considers each instance individually for learning, whereas the ranking method considers groups of instances to learn the ranking of candidate entities towards a mention. We assume that the descriptive feature vectors for negative candidates, are too similar to each other and that this derogates the ranking approach.

For this classification approach with a standard SVM, we decided to not use artificial NIL candidates to learn a threshold for uncovered entities as in Bunescu and Pasca [2006], Pilz and Paaß [2009, 2012]. Instead, we use a threshold τ and define the prediction $\hat{e}(m) = \text{NIL}$ if the model predicts no score y greater than τ for any of the candidates $e_i(m) \in \mathbf{e}(m)$:

$$\hat{e}(m) = \text{NIL if } \max_{e_k \in \mathbf{e}(m)} y^{(k)} \leq \tau. \tag{3.51}$$

In initial experiments, we empirically determined the value of $\tau = 0$ to give the best results. The results obtained with this setting will be described in Section 3.6.4.

Alternative formulations are one-model-per-entity or multi-class-classification approaches. In the one-model-per-entity-approach, we would learn one separate model per entity. In the multi-class model, we would represent each entity as one distinct class. However, we consider both alternatives as not appropriate. This is because for a multi-class approach, the number of classes is proportional to the size of $\boldsymbol{\mathcal{W}}$ which would result in 3 million classes in the most general case. Then we can also expect a very skewed distribution of positive examples over the classes. For a more tractable formulation, subgroups would need to be determined which requires additional effort in model design. We argue that this is not necessary using the classification method described above. The one-model-per-entity approach is also practically difficult to realize since we would need to manage a huge number of models, for instance when naively storing one SVM model per entity. While probably suitable for smaller entity collections, we argue that our formulation of one model for all entities is more elegant since we can also exploit interactions between instances.

Having described an entity linking model based on a classifier with thematic distances as features, we will now detail the corpora used to evaluate this model.

3.6.3 Wikipedia Reference Datasets

Our aim is to build an entity linking model focused on persons that is applicable in more than one language. Even though recent work published a series of benchmark datasets, these datasets mostly consists of English documents. To the best of our knowledge, we are not aware of publicly available benchmark datasets with persons linked to Wikipedia for other languages such as German and French. Therefore, we resort to Wikipedia to provide disambiguated examples. While we are aware that Wikipedia documents are different from edited newspaper articles, regarding semantics, structure and topics, we assume that the model evaluated on this dataset can generalize to other corpora.

We aim at retrieving highly ambiguous datasets, i.e. datasets where mentions have many candidates in the candidate pool. To do so, we use two strategies to fill the candidate pools. This results in two different datasets, one consisting of mentions for persons, the other containing mentions of entities of diverse types. For the first strategy, we extract persons with ambiguous names by focusing on name phrases that refer to at least two distinct entities. This strategy enables a fair comparison with Bunescu and Pasca's method. For the second strategy, we widen the candidate pool by allowing partial matches for the common English surnames Jones, Taylor and Smith and removing the constraint that a candidate must be a person. Doing so, we obtain a broad set of entities that each contain at least one of these highly ambiguous seed names in their $name(e)$ but need not be of type person. Using this strategy, we empirically show that our method generalizes to other concepts apart

Table 3.1: Wikipedia evaluation datasets for English, German and French (indicated by subscript). The table shows for each dataset the number of entities in the candidate pool $\boldsymbol{\mathcal{W}}_c$, the number of extracted contexts $d \in \mathbf{D}$, the number of covered ($e^+(m) \in \boldsymbol{\mathcal{W}}_c$) and uncovered entity mentions ($e^+(m) = \text{NIL}$) and the average ambiguity per mention given by the average cardinality $|\mathbf{e}(m)|$ of candidates sets.

dataset	$\lvert\boldsymbol{\mathcal{W}}_c\rvert$	$\lvert\mathbf{D}\rvert$	$e^+(m) \in \boldsymbol{\mathcal{W}}_c$	$e^+(m) = \text{NIL}$	avg. $\lvert\mathbf{e}(m)\rvert$
WikiPersons$_\text{E}$	6213	16661	13593	3068	2.06
WikiMisc$_\text{E}$	10734	15481	13849	1632	26.76
WikiPersons$_\text{G}$	18024	44338	35367	8971	2.91
WikiPersons$_\text{F}$	7201	15159	12284	2875	1.88

from persons. Furthermore, the latter strategy accounts for cases where entities are referenced merely by the surname, which renders the distinction of candidates even more difficult.

Note that since all versions of Wikipedia are endowed with hyperlink structures, we may employ these strategies not only for the English version[1], but also for the German[2] and the French version[3]. From this, we obtain the datasets as summarized in Tab. 3.1. Their generation process will be further detailed next.

Using the first strategy, we start with the extraction of example mentions for persons with ambiguous names. We call the resulting corpus **WikiPersons** in the following and use an index to denote the language version of Wikipedia, i.e. **WikiPersons**$_\text{E}$ for the examples from the English version of Wikipedia and **WikiPersons**$_\text{G}$ for the German resp. **WikiPersons**$_\text{F}$ for the French version.

First, we need to identify articles describing persons. For this, we use both the type system of YAGO and Wikipedia categories. YAGO's type system provides the information whether an article describes a person which we use to determine person articles in our version of Wikipedia. Even though YAGO was built over a different version of Wikipedia, we may use it to determine persons in our version since older articles usually still exist and we may align them with our version via their unique titles. Articles in our version that previously not existed are consequently ignored and not used in the candidate pool for **WikiPersons**$_\text{E}$.

However, since YAGO is build over the English version of Wikipedia, we can not solely rely on it to detect all persons in the German and French versions via language links. Therefore we use simple heuristics such as the presence of the categories *Mann* or *Frau* to detect persons in the German Wikipedia and *Naissance* for the French version. While there are certainly more precise ways to determine persons

[1] `http://www.en.wikipedia.org`, retrieved on January 15, 2011.
[2] `http://www.de.wikipedia.org`, retrieved on January 31, 2011
[3] `http://www.fr.wikipedia.org`, retrieved on February 1st, 2011.

in other Wikipedia language versions, for example by analysing their individual category trees, this may require deeper understanding of these languages and further investigations that are not the focus of this thesis. For future work, we note that the multilingual entity taxonomy created in de Melo and Weikum [2010] may serve as a better alternative. As Tab. 3.1 shows, we could extract comparable number of persons in all versions of Wikipedia using YAGO types and our heuristics. The higher number of examples for the German version results from the comparably high amount of biographic entries in the German version.

From the Wikipedia articles describing persons, we select entities with ambiguous names. A person name is ambiguous in Wikipedia when at least two entities have the same name which is the title without disambiguation term. More specifically, when matching the name $name(e)$ against the name of other persons, we obtain at least one other person:

$$\begin{aligned} \boldsymbol{\mathcal{W}}_{Per} &= \{e \in \boldsymbol{\mathcal{W}} \mid c = \textit{''person''} \in \boldsymbol{c}(e)\} \\ \boldsymbol{\mathcal{W}}_{c} &= \{e \in \boldsymbol{\mathcal{W}}_{Per} \mid |\mathbf{e}(name(e)) \subseteq \boldsymbol{\mathcal{W}}_{Per}| \geq 2\}, \end{aligned} \tag{3.52}$$

where $\mathbf{e}(name(e)) \subseteq \boldsymbol{\mathcal{W}}_{Per}$ contains all persons in $\boldsymbol{\mathcal{W}}_{Per}$ whose name completely matches (case-insensitive equality) the name of e. For example, Jonas Taylor does not match John Taylor but John Taylor (jazz) does. The condition $c = \textit{''person''} \in \boldsymbol{c}(e)$ relies on the alignment with YAGO that provides this specific type and is used by us as a category. With a random selection on entities fulfilling these conditions, we arrive at a candidate pool $\boldsymbol{\mathcal{W}}_c$ with 6213 different entities for $\mathbf{WikiPersons}_{\mathrm{E}}$.

The dataset $\mathbf{WikiMisc}_{\mathrm{E}}$ is created using the second strategy. Here, we use no constraint on the entity type but focus on frequent names and alter the matching technique from exact matches to partial name matches. Given the seed names $\{\textit{jones}, \textit{taylor}, \textit{smith}\}$, an entity is added to $\boldsymbol{\mathcal{W}}_c$ if its name contains at least one of these names as a substring, i.e.:

$$\begin{aligned} \boldsymbol{\mathcal{W}}_{c} = \{e \in \boldsymbol{\mathcal{W}} \mid & \texttt{hasSubstring}(name(e), \textit{''smith''}) \\ & \vee \texttt{hasSubstring}(name(e), \textit{''taylor''}) \\ & \vee \texttt{hasSubstring}(name(e), \textit{''jones''})\} \end{aligned} \tag{3.53}$$

For instance, the entity Bruce Jones would be added to the candidate pool $\boldsymbol{\mathcal{W}}_c$ defined above, since $\texttt{hasSubstring}(\text{Bruce Jones}, \textit{''jones''})$ is true. Using again a random selection of the entities fulfilling these conditions, we arrive at a candidate pool $\boldsymbol{\mathcal{W}}_c$ with 10734 different entities for $\mathbf{WikiMisc}_{\mathrm{E}}$.

Aiming at high disambiguation performance not only for popular entities with many inlinks but also less popular entities with few inlinks, we set again a boundary on the number of examples per entity. This is achieved by using at most ten randomly selected inlinks from the set $\mathbf{L}_{in}(e)$ as examples, i.e. $n = 10$ in Alg. 1.

Again, we argue that this restriction allows a more balanced model over all entities in $\mathcal{W}_c$ and avoids over-fitting towards high popularity entities. Here, each of the example documents d contains the complete article text $text(l_s)$ of the link source l_s. Using the complete article text enables us to experimentally evaluate the influence of context width and we will discuss this in Section 3.6.4. In line with the corpus design described in Section 3.5.4, we treat only the mention of the entity from the candidate pool $\mathcal{W}_c$ and do not handle any other potentially appearing mention, i.e. any other link.

Given a mention m, we select its candidate entities in the same way we generate the candidate pool. In the case of **WikiPersons**, an entity e is considered as candidate if its name fully matches the surface form of the link anchor text l_a associated with the mention m, i.e. $l_a = name(e)$. In the case of $\textbf{WikiMisc}_{\text{E}}$, an entity e is considered as a candidate if this surface form is contained as a substring in the candidate's name, i.e. if $\texttt{hasSubstring}(l_a, name(e))$ is true. Using a partial match for candidate selection, the surface name Jones may then match Adam Jones or Catherine Zeta-Jones, but also Jones Soda or Jones, Oklahoma. This way we get on average more than 27 candidates per mention and thus a highly ambiguous dataset where references are not restricted to mentions of persons.

Apart from creating a different candidate pool $\mathcal{W}_c$ and using a different candidate selection method for $\textbf{WikiMisc}_{\text{E}}$, we also set a different boundary on the number of examples n. While example extraction is performed analogously to $\textbf{WikiPersons}_{\text{E}}$, we use at most $n = 5$ randomly selected inlinks per entity to arrive at datasets of comparable size. Otherwise, the number of example documents in $\textbf{WikiMisc}_{\text{E}}$ would be much higher, since the cardinality of the candidate pool here is nearly twice that of the candidate pool of $\textbf{WikiPersons}_{\text{E}}$.

For all datasets, we simulate mentions of uncovered entities by marking every fifth entity in the candidate pool $\mathcal{W}_c$ as uncovered, i.e. by setting $z = 5$ in Alg. 1. Tab. 3.1 shows that for $\textbf{WikiPersons}_{\text{E}}$, we arrive at 16661 example documents where 13593 are contexts of linkable mentions, i.e. $e^+(m) \in \mathcal{W}_c$ and 3068 are contexts for mentions with $e^+(m) = \text{NIL}$. The average ambiguity per mention is 2.06 which does not include the symbolic entity NIL and concerns only candidates $e \in \mathcal{W}_c$. For $\textbf{WikiPersons}_{\text{E}}$, the entity pool $\mathcal{W}_c$ contains 6213 different candidate entities. After the simulation of uncovered entities, we have a ratio of 1242 uncovered vs. 4971 covered entities. For $\textbf{WikiMisc}_{\text{E}}$, the entity pool $\mathcal{W}_c$ contains 10734 different candidate entities. After the simulation of uncovered entities, we have a ratio of 2146 uncovered vs. 8588 covered entities.

As the proposed method is in general language independent, we evaluate name disambiguation also on German and French datasets. To do so, we extract example contexts both from the German and the French version of Wikipedia using the same extraction technique as for $\textbf{WikiPersons}_{\text{E}}$ but adapt indicative categories. Then, both datasets contain references for persons with ambiguous names.

Particularly, for $\mathbf{WikiPersons}_{\mathrm{G}}$, we alter the category condition in Eq. 3.52 to

$$\boldsymbol{\mathcal{W}}_c = \{e \in \boldsymbol{\mathcal{W}} \mid (c = \textit{"Frau"} \in \boldsymbol{c}(e)) \vee (c = \textit{"Mann"} \in \boldsymbol{c}(e))\} \tag{3.54}$$

and arrive at candidate pool $\boldsymbol{\mathcal{W}}_c$ containing 18024 distinct, randomly selected entities fulfilling this condition. $\mathbf{WikiPersons}_{\mathrm{G}}$ then contains 44338 example documents, with 35367 contexts of linkable mentions, 8971 contexts of uncovered mentions and an average ambiguity of 2.91.

For $\mathbf{WikiPersons}_{\mathrm{F}}$ we alter the category condition in Eq. 3.52 to a partial match on category tags

$$\boldsymbol{\mathcal{W}}_c = \{e \in \boldsymbol{\mathcal{W}} \mid \exists c \in \boldsymbol{c}(e) : \texttt{hasSubstring}(c, \textit{"Naissance"})\}. \tag{3.55}$$

This means that it is sufficient that the word *"Naissance"* is contained as a substring in any of the category tags. For $\mathbf{WikiPersons}_{\mathrm{F}}$ we then have a candidate pool $\boldsymbol{\mathcal{W}}_c$ of 7201 different entities and a reference dataset of 15159 example documents, with 12284 contexts of linkable mentions, 2875 contexts of uncovered mentions and an average ambiguity of 1.88. Again, for both datasets, the average ambiguity does not include NIL as a candidate.

Analogously to the observations described in Section 3.5.4, we also find problematic links in these datasets. Some links are rather conceptional and point to a thematically related article, which does not imply identity. For example, the term client can be linked to the article Lawyer.

3.6.4 Evaluation

In Pilz and Paaß [2011] we proposed a splitting strategy for cross-validation that draws the instances in the cross-validation buckets not randomly from all examples but takes into account the ground truth target entities. In this *entity based* splitting, train and test folds are disjunct with respect to the ground truth target entities of the contained instances. The motivation for this splitting strategy is that it allows us to asses the models ability to generalize on new contexts *and* on new entities. We used this strategy for all of the corpora treated in Pilz and Paaß [2011] and reported the obtained results, finding that our model generalizes in both aspects. This important kind of evaluation was used in no other approach, which renders the results published in Pilz and Paaß [2011] unique. For the discussion in this thesis, we will also give results in this entity based splitting for $\mathbf{WikiPersons}_{\mathrm{G}}$ and $\mathbf{WikiPersons}_{\mathrm{F}}$, but will generally focus on the results obtained for the standard, instance based splitting strategy. We argue that these results are expressive enough to demonstrate the effectiveness of our methods and more directly comparable to results from the literature that generally uses instance based splitting.

We start the description of our evaluation with the experiments we conducted to find the best representation of thematic context distance and the most suitable selection of the mention's context.

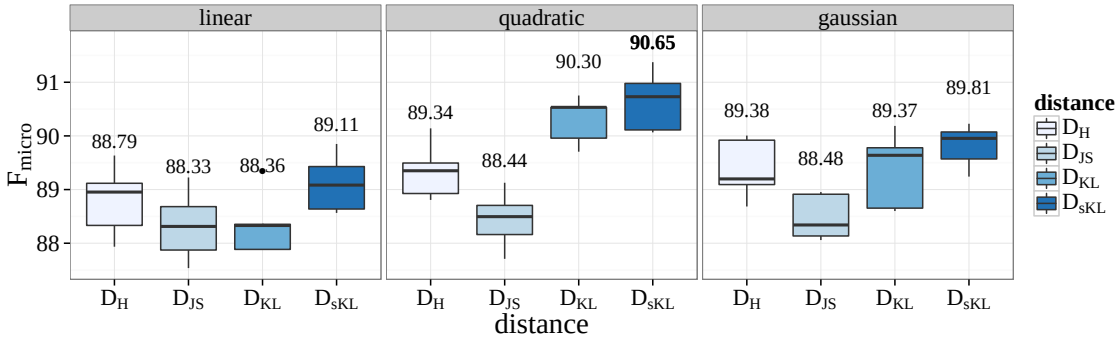


Figure 3.8: F_{micro} performance for entity linking on **WikiPersons**$_{\mathrm{E}}$ (all values in %). Here, uncovered entity mentions are simulated at random. We compare thematic distance representations in combination with different kernel types in 5-fold cross-validations. The mean of each sample is given above the boxes, the best performance (in bold) is obtained for D_{sKL} with a quadratic kernel.

Thematic Distances

To find the best distance representation, we evaluate the distances described in Section 3.6.1 with different kernels using SVM$^{\mathrm{Light}}$ standard parameters in five fold cross-validations on the dataset **WikiPersons**$_{\mathrm{E}}$. For a better comparison, we use here the results obtained with standard instance based splitting for cross-validation instead of the figures given in Pilz and Paaß [2011] that were obtained using the entity based splitting for cross-validation.

Fig. 3.8 visualizes the results obtained for each distance in a linear, a quadratic and a Gaussian kernel. On the first glance, there are no striking differences among the different combinations. To statistically compare the different representations, we therefore employ paired t-tests on the F_{micro} results over the cross-validation folds. This showed that the best result is obtained with the symmetric Kullback-Leibler distance D_{sKL} in a quadratic kernel.

With a linear kernel, the Hellinger distance D_{H} and the symmetric Kullback-Leibler distance D_{sKL} perform best. Using the more complex quadratic kernel increases performance for all distances, most notably for D_{sKL}. The Gaussian kernel with standard parameters however is not superior. Also, we find the symmetric Kullback-Leibler distance representation D_{sKL} superior to the asymmetric variant D_{KL} ($p < 0.02$) in all kernels. The same is true for the comparison of D_{sKL} with the Jensen-Shannon distance D_{JS} ($p < 0.01$), the latter giving the lowest performance in all cases. Comparing D_{sKL} with the Hellinger distance D_{H}, we find significantly better results for D_{sKL} ($p < 0.03$) in the linear and quadratic kernel, while there is no significant difference when using a Gaussian kernel. The asymmetric D_{KL} is inferior to D_{H} only for the linear kernel. From this evaluation we conclude that the

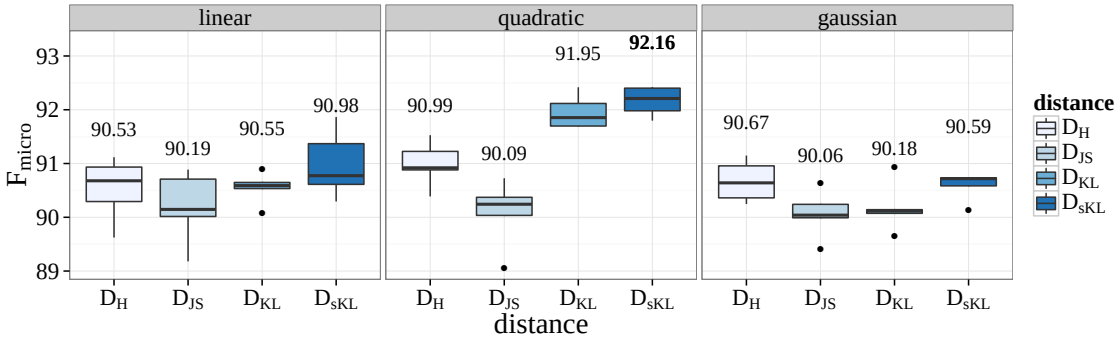


Figure 3.9: F_{micro} performance for entity linking on **WikiPersons**$_{\mathrm{E}}$ (all values in %). Here, uncovered entity mentions are simulated taking into account article text length. We compare thematic distance representations in combination with different kernel types in 5-fold cross-validations. The mean of each sample is given above the boxes, the best performance (in bold) is obtained for D_{sKL} with a quadratic kernel.

symmetric Kullback-Leibler distance D_{sKL} is most suitable for entity linking based on thematic distance. Since the Gaussian kernel does not provide superior results, we decide to not evaluate it further but present results in later experiments for the simpler representations in linear and quadratic kernels.

In this evaluation, we simulated uncovered entities for **WikiPersons**$_{\mathrm{E}}$ by removing every 5th entity from the candidate pool. Alternatively, we can simulate uncovered entity mentions taking into account article text length, for example by removing entities with short article texts where only few contextual information is available. In an additional experiment, we therefore simulate uncovered entity mentions by removing all the ground truth entities with an article text of less than 50 words. Following the described extraction strategy, we obtain less examples for uncovered entity mentions, 1674 instead of 3068, since short articles naturally tend to have fewer inlinks. The results obtained on this dataset are depicted in Fig. 3.9. The performance shows similar behaviour for both simulation strategies, even though the results are with one to two points in percentage slightly superior to those obtained with random simulation of uncovered entities. There are two possible reasons for this. First, when removing entities by article length, we can expect to find more well described candidates in the candidate pool and thus topic distributions are also more stable over their respective article texts. Since furthermore the number of mentions for uncovered entities is also lower, this dataset can be considered less difficult.

Since simulating uncovered entity mentions via article length puts a bias towards popular, well described entities and furthermore renders the linking problem artificially slightly easier, we decide on the random strategy for further experiments.

Context Properties

Bagga and Baldwin [1998], Gooi and Allan [2004] and Bunescu and Pasca [2006] observed that the words in a mention's close neighbourhood often contain most of the information necessary for its disambiguation. These authors therefore focus on localized context windows, usually with a width of 25 to 50 words centred around the mention, i.e. $text(m)_{-25,25}$ or $text(m)_{-50,50}$. In contrast, due to the sampling over words, topic distributions tend to be more representative when more context is available. Therefore, we evaluated different context widths for our method in initial experiments. We found that reducing the available context to local windows around the mention yields a slight decrease in predictive performance. This goes along with our assumption that we obtain a higher stability in the topic probability distribution when a larger context is used. Consequently, we use the full text of the link source l_s as mention context $text(m)$ to infer the topic distribution $\mathcal{T}_m$.

However, to put more emphasis on the local context, we propose a *local boosting*. Local boosting uses a context window around the mention and adds the terms from this window repeatedly to the overall document words. We found that boosting the ten word context window around the mention yields the best result. The terms from this window are then added five times to the words of the example document, i.e.

$$text(m) = text(l_s) \underbrace{\cup\, text(m)_{-10,10} \ldots \cup\, text(m)_{-10,10}}_{\times 5.} .$$

We found that boosting the local context in this manner increases performance significantly ($p < 0.05$) in comparison with the standard, non-boosted version. Hence, we use the local boosting on mention contexts for all experiments following in this chapter. Note that this does not affect the entity distributions $\mathcal{T}_e$, for those we use the full context $text(e)$ without boosting.

Having described the parameters for distances and contexts, we now evaluate thematic context distance for entity linking against the word-category correlation method proposed by Bunescu and Pasca [2006] (WCC was described in Section 3.5) on the English datasets $\mathbf{WikiPersons}_{\mathrm{E}}$ and $\mathbf{WikiMisc}_{\mathrm{E}}$. In Pilz and Paaß [2009] and Pilz and Paaß [2011], we compared against the version cWCC using only common words with the feature representation as in Eq. 3.6. For a more thorough comparison, we here additionally provide the results obtained for the original formulation as in Eq. 3.5. When referring to this method, we use WCC to denote the full and cWCC to denote the version restricted to common words. For the implementation of cWCC and WCC, we extract 5825 categories analogously to Section 3.5.4 from the English Wikipedia and furthermore always add the required candidate NIL that has no attributes apart from the NIL-feature as in Eq. 3.19.

We use $\mathrm{D}_{\mathrm{sKL}}$ to denote our proposed method which exploits thematic context distance through the symmetric Kullback-Leibler distance over the topic distributions $\mathcal{T}_m$ and $\mathcal{T}_e$. Since we also evaluate different kernels, we index this notation with

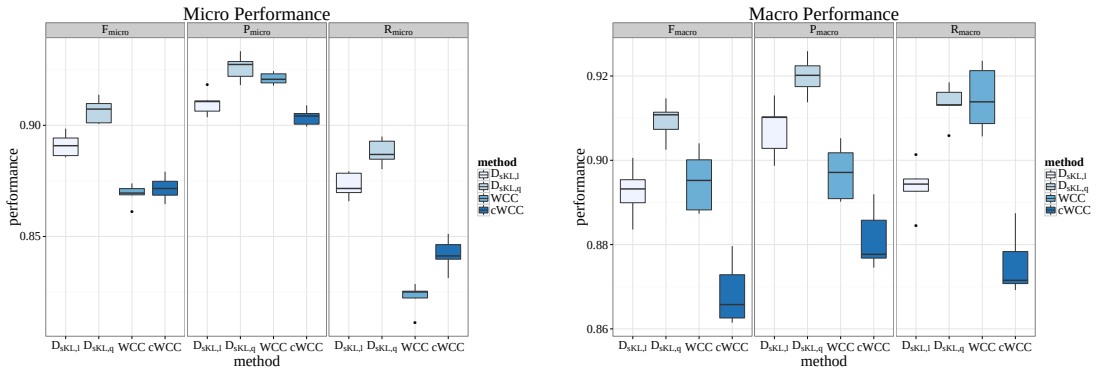


Figure 3.10: Micro and macro performance on **WikiPersons**$_{\mathrm{E}}$ for the methods $\mathrm{D}_{\mathrm{sKL,l}}$, $\mathrm{D}_{\mathrm{sKL,q}}$ and the competitor methods cWCC and WCC (all values in %).

$\mathrm{D}_{\mathrm{sKL,l}}$ to indicate the usage of a linear kernel and $\mathrm{D}_{\mathrm{sKL,q}}$ to indicate the usage of a quadratic kernel. As described previously and depicted in Fig. 3.8, the Gaussian kernel did not perform better, so we omitted experiments using this kernel.

All evaluations are performed in five-fold cross-validations with instance based splitting and compared for significant differences through paired t-tests with $p < 0.05$. We start with the results obtained on the dataset **WikiPersons**$_{\mathrm{E}}$ for which we evaluated our approach also in a Ranking SVM instead of a standard classification SVM but obtained remarkably inferior results.

Evaluation for Person Name Mentions

To emphasize the performance for uncovered entity mentions, we also report separate accuracy values for covered and uncovered entity mentions. The accuracy for covered entity mentions $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ is given by the ratio of mentions that were correctly assigned to an entity in $\boldsymbol{\mathcal{W}}_c$ and the overall number of covered entity mentions, i.e.

$$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c} = \frac{|\{\hat{e}(m) = e^+(m) \in \boldsymbol{\mathcal{W}}_c\}|}{|\{e^+(m) \in \boldsymbol{\mathcal{W}}_c\}|}. \tag{3.56}$$

Analogously, the accuracy for uncovered entity mentions $\mathrm{Accuracy}_{\mathrm{NIL}}$ is given by the ratio of mentions that were correctly assigned to NIL and the overall number of uncovered entity mentions, i.e.

$$\mathrm{Accuracy}_{\mathrm{NIL}} = \frac{|\{\hat{e}(m) = e^+(m) = \mathrm{NIL}\}|}{|\{e^+(m) = \mathrm{NIL}\}|}. \tag{3.57}$$

Fig. 3.10 visualizes the results obtained on the dataset **WikiPersons**$_{\mathrm{E}}$, the explicit figures are given in Tab. 3.2. In all cases, our methods using thematic context

Table 3.2: Results on **WikiPersons**$_{\mathrm{E}}$ (all values in %). The best result for each measure is in bold and marked with an asterisk if the difference towards the 2nd best method is significant ($p < 0.05$). As our methods $\mathrm{D}_{\mathrm{sKL,l}}$ and $\mathrm{D}_{\mathrm{sKL,q}}$ overall perform significantly better than cWCC ($p < 0.05$), we indicate differences only towards WCC for the sake of readability. In terms of $\mathrm{Accuracy}_{\mathrm{NIL}}$, the overall best result is obtained with $\mathrm{D}_{\mathrm{sKL,l}}$ in a Ranking SVM (significant superiority to $\mathrm{D}_{\mathrm{sKL,l}}$ in a standard SVM is indicated by †).

	Bunescu and Pasca		Thematic Context Distance		
			SVM		Ranking SVM
measure	cWCC	WCC	$\mathrm{D}_{\mathrm{sKL,l}}$	$\mathrm{D}_{\mathrm{sKL,q}}$	$\mathrm{D}_{\mathrm{sKL,l}}$
F_{micro}	87.17	86.90	89.11	**90.65***	83.19
P_{micro}	90.37	92.10	91.00	**92.59**	84.48
R_{micro}	84.20	82.25	87.30	**88.79***	81.95
F_{macro}	86.85	89.50	89.25	**90.93***	82.57
P_{macro}	88.13	89.70	90.75	**91.99***	85.60
R_{macro}	87.55	**91.46**	89.37	91.33	81.72
$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$	87.58	91.57	89.11	**92.14**	81.13
$\mathrm{Accuracy}_{\mathrm{NIL}}$	69.20	40.93	**79.30***	78.00	**85.08**†

in a symmetric Kullback-Leibler distance in a linear ($\mathrm{D}_{\mathrm{sKL,l}}$) or a quadratic kernel ($\mathrm{D}_{\mathrm{sKL,q}}$) perform significantly better than cWCC ($p < 0.05$). Comparing the linear and quadratic variant, we find that the quadratic variant is significantly ($p < 0.05$) superior to the linear variant in most cases. Only regarding the accuracy of uncovered entity mentions $\mathrm{Accuracy}_{\mathrm{NIL}}$, the linear variant $\mathrm{D}_{\mathrm{sKL,l}}$ is superior to the quadratic variant $\mathrm{D}_{\mathrm{sKL,q}}$. Interestingly, the full version WCC obtains with 40.93% a notably lower accuracy for uncovered entity mentions $\mathrm{Accuracy}_{\mathrm{NIL}}$ than the restricted version cWCC with 69.20%.

Comparing the full version WCC and the linear $\mathrm{D}_{\mathrm{sKL,l}}$, we find that WCC achieves a significantly ($p < 0.05$) higher P_{micro} of 92.10 % and R_{macro} of 91.46% compared to the respective values of P_{micro} of 91% and R_{macro} of 89.37% for $\mathrm{D}_{\mathrm{sKL,l}}$. However the difference in F_{macro} among these methods is not significant. The R_{macro} of $\mathrm{D}_{\mathrm{sKL,q}}$ is with 91.33% higher than that of $\mathrm{D}_{\mathrm{sKL,l}}$ and we find that then there is no more significant difference between $\mathrm{D}_{\mathrm{sKL,q}}$ and WCC. The same is true comparing the accuracy for covered entity mentions $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ for the two methods. $\mathrm{D}_{\mathrm{sKL,q}}$ achieves an $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ of 92.14% and WCC achieves an $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ of 91.57, but the difference in $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ among WCC and $\mathrm{D}_{\mathrm{sKL,q}}$ is not significant.

To summarize, our proposed method using thematic context distance over mention and entity contexts performs significantly ($p < 0.05$) better than the competitor method proposed in Bunescu and Pasca [2006] in most measures. We obtain with

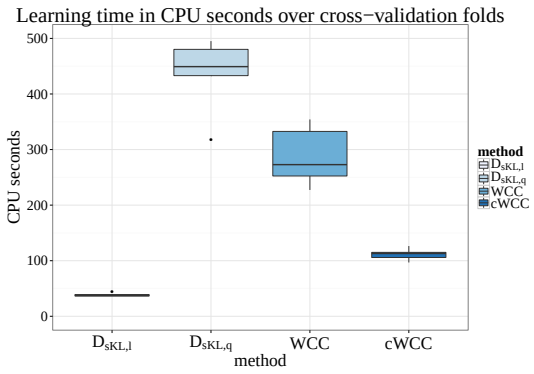


Figure 3.11: SVM learning time per method in CPU seconds, aggregated over cross-validation folds on **WikiPersons**$_{\mathrm{E}}$.

79.30% resp. 78% a significantly ($p < 0.01$) higher Accuracy$_{\mathrm{NIL}}$ for uncovered entity mentions, even though we did not learn an adapted threshold and used the empirically determined $\tau = 0$. The low R_{micro} for WCC and cWCC results from the low Accuracy$_{\mathrm{NIL}}$ for uncovered entity mentions that make up about 25% of all examples. WCC and cWCC both perform well for covered entity mentions. Therefore R_{macro} is notably higher as uncovered entity mentions are summarized by a NIL class which is again outweighed by the comparably high number of other entities. Since both D$_{\mathrm{sKL,l}}$ and D$_{\mathrm{sKL,q}}$ obtain a high Accuracy$_{\mathrm{NIL}}$ for uncovered entity mentions, R_{micro} and R_{macro} are consequently very close.

Tab. 3.2 also shows the results when we use a Ranking SVM instead of a standard SVM for our method D$_{\mathrm{sKL,l}}$. We see that using a Ranking SVM instead of a standard SVM results in notably lower performance. For the Ranking SVM, we used the same feature set as for the standard SVM but enabled threshold learning through NIL candidates in the same way as for cWCC and WCC. As a result, the Accuracy$_{\mathrm{NIL}}$ for uncovered entity mentions is notably higher. However, since this is the only measure for which we find an improvement, we argue that this learner is inferior to the standard SVM with this feature setting. For future work, it would be interesting to evaluate the Ranking SVM and standard SVM in a joint model, where the threshold is learned by the Ranking SVM but classification is performed with the standard SVM.

We also evaluated the average learning time for the methods cWCC, WCC, D$_{\mathrm{sKL,l}}$, and D$_{\mathrm{sKL,q}}$. For this, we record the SVM's computation time per cross-validation fold for each method on **WikiPersons**$_{\mathrm{E}}$ and depict the results in Fig. 3.11. This figure shows the SVM learning time for each method in CPU seconds, aggregated over the cross-validation folds. We see that the learning time is with an average of about 435 CPU seconds the longest for D$_{\mathrm{sKL,q}}$, i.e. the method using a quadratic kernel. The shortest learning time is with an average of 38.71 CPU seconds observed for

$D_{sKL,l}$, i.e. the method using a linear kernel. This is also about three times faster than the average learning time for cWCC and about seven times faster than the average learning time for WCC. Given that the full variant WCC has a far higher feature dimensionality than the restricted variant cWCC, WCC has a notably higher complexity and consequently also an increased learning time (about 2.5 times that of cWCC).

It is not surprising that the learning time of the quadratic kernel variant is notably longer than the linear variants as more parameters need to be estimated from the training data and the complexity increases. Given the good performance of the linear variant $D_{sKL,l}$ as detailed above and depicted in Tab. 3.2, we would thus recommend this variant for practical applications that have to obey certain time constraints.

Lastly, since we use cosine similarity as a baseline feature for all methods, we also evaluated this feature alone in preliminary experiments on **WikiPersons**$_{\mathrm{E}}$. With a linear kernel, the SVM classifier could not determine an optimum value and aborted optimization. In that case, the cosine similarity consequently showed a poor performance of only 18.27% in F_{micro}. We assume that a linear kernel can not separate the feature vectors described by cosine similarity alone. In contrast, with a quadratic kernel, we obtained an F_{micro} of 78.24% using only this baseline feature.

Evaluation for General Entity Mentions

Unfortunately, there was a mistake in the experiment reported for the dataset **WikiMisc**$_{\mathrm{E}}$ in Pilz and Paaß [2011]. We wrongly set a parameter of SVM$^{\mathrm{Light}}$ and, instead of a Ranking SVM, we used a standard SVM as classifier for Bunescu and Pasca's method. At the time of publication we were not aware of this and reported the obtained results to the best of our knowledge. When re-running experiments, this error became obvious and we accordingly report the correct results here in Tab. 3.3.

The high ambiguity and the more diverse entity types in **WikiMisc**$_{\mathrm{E}}$ render this dataset more demanding for all methods. The high number of candidates results also in a high number of negative examples, which was approached through automatic cost ratio adaption for all methods. Nevertheless, we find notably lower performance on this dataset for all methods. While we find P_{micro} to be comparable for our methods $D_{sKL,l}$ and $D_{sKL,q}$, all other measures drop by about 3 to 6 points in percentage. However, the decline in performance is with about 20 points in percentage far stronger for cWCC and WCC. In contrast to the dataset **WikiPersons**$_{\mathrm{E}}$, we also find that the restricted version cWCC is significantly ($p < 0.05$) superior to the full version WCC. The high value of WCC for Accuracy$_{\mathrm{NIL}}$ in Tab. 3.3 must be interpreted taking into account the accuracy for covered entity mentions in order to avoid misleading interpretations. Basically, the method predicted NIL in most cases and therefore the accuracy for uncovered entity mentions Accuracy$_{\mathrm{NIL}}$ is high, whereas the accuracy for covered entity mentions Accuracy$_{\boldsymbol{\mathcal{W}}_c}$ is rather low. Also,

Table 3.3: Results on **WikiMisc**$_{\mathrm{E}}$ (all values in %). The best result for each measure is in bold and marked with an asterisk if the difference towards the 2nd best method is significant ($p < 0.05$). Our methods $\mathrm{D}_{\mathrm{sKL,l}}$ and $\mathrm{D}_{\mathrm{sKL,q}}$ are significantly ($p < 0.05$) superior to cWCC and WCC for all measures apart from $\mathrm{Accuracy}_{\mathrm{NIL}}$.

	Bunescu and Pasca		Thematic Context Distance	
measure	cWCC	WCC	$\mathrm{D}_{\mathrm{sKL,l}}$	$\mathrm{D}_{\mathrm{sKL,q}}$
F_{micro}	67.02	64.38	86.74	**87.12***
P_{micro}	69.32	66.58	91.91	**92.43***
R_{micro}	64.87	62.33	82.13	**82.39***
F_{macro}	65.44	62.80	86.35	**86.83***
P_{macro}	68.51	66.08	87.06	**87.45***
R_{macro}	64.35	61.50	86.66	**87.27***
$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$	63.67	60.82	86.91	**87.50***
$\mathrm{Accuracy}_{\mathrm{NIL}}$	74.17	**75.13**	41.53	39.04

there is no significant difference in this measure for cWCC and WCC.

Even though our proposed methods show a decline in performance on the dataset **WikiMisc**$_{\mathrm{E}}$, we see that they are more favourable for entity linking than the competitor methods. Apart from the accuracy for uncovered entity mentions, neither measure drops below 86%, a figure that can be satisfactory in most use cases and applications. However, on this dataset, the threshold τ was not appropriate since the accuracy for uncovered entity mentions dropped significantly for our methods. Again, we assume that the earlier proposed combination with the Ranking SVM to learn a threshold may promise more satisfying results.

We conclude that the proposed thematic context distance is a very good method for the disambiguation of name phrases but more suitable for the disambiguation of person names. Due to the often biographic nature of person descriptions, the thematic overlap with their reference contexts tends to be higher than for other entity types that may be mentioned off-topic (e.g. locations as geographic anchors of events in news documents).

As a side effect, our mistake using the 'wrong' learner allows for an interesting observation, namely that WCC is very sensitive regarding the machine learning method. Using a standard SVM results in an average performance of about 16%, whereas a Ranking SVM as learner results in notably higher values of more than 60%. In contrast, our method dropped only by about 10 points in percentage on **WikiPersons**$_{\mathrm{E}}$ when substituting the standard SVM with a Ranking SVM.

Table 3.4: WTC on **WikiPersons**$_{\mathrm{E}}$ and **WikiMisc**$_{\mathrm{E}}$ using a standard SVM and a Ranking SVM as learner (all values in %). Values significantly ($p < 0.05$) higher than those obtained with $\mathrm{D}_{\mathrm{sKL,q}}$ in a standard SVM are marked in bold.

	Word-topic correlation (WTC)			
	WikiPersons$_{\mathrm{E}}$		**WikiMisc**$_{\mathrm{E}}$	
measure	SVM	Ranking SVM	SVM	Ranking SVM
F_{micro}	87.48	**91.88**	74.92	80.30
P_{micro}	88.49	**93.40**	76.38	82.47
R_{micro}	86.49	**90.42**	73.51	78.24
F_{macro}	86.03	91.08	73.41	79.00
P_{macro}	87.89	91.85	75.73	80.63
R_{macro}	85.72	91.55	72.70	78.82
$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$	86.16	91.88	72.73	78.82
$\mathrm{Accuracy}_{\mathrm{NIL}}$	**87.99**	**83.94**	**80.07**	**73.26**

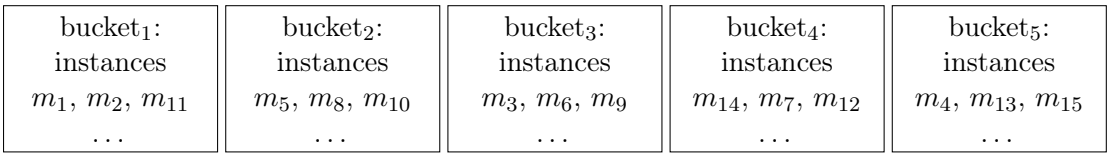
Comparison with Word-Topic-Correlation

To show that thematic context distance is superior to the word-topic correlation approach WTC proposed in Section 3.5, we evaluated the latter also on the datasets **WikiPersons**$_{\mathrm{E}}$ and **WikiMisc**$_{\mathrm{E}}$. The results are given in Tab. 3.4.

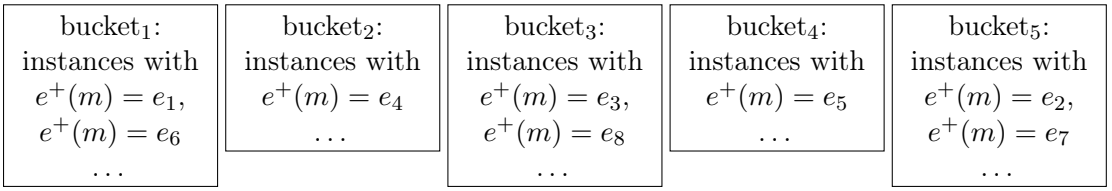
Using a standard SVM classifier, the results obtained with WTC on **WikiPersons**$_{\mathrm{E}}$ are on average about 3 points in percentage (pp) lower than those obtained using $\mathrm{D}_{\mathrm{sKL,q}}$ (cf. Tab. 3.2). More specifically, WTC achieved an F_{micro} of 87.48% and an F_{macro} of 86.03% on **WikiPersons**$_{\mathrm{E}}$ compared to the F_{micro} of 90.65% and F_{macro} of 90.93% for $\mathrm{D}_{\mathrm{sKL,q}}$, the latter also in a standard SVM.

Replacing the learner with a Ranking SVM however could increase the performance to an F_{micro} of 91.88% and an F_{macro} of 91.08%. As we see in Tab. 3.4, the performance is then also superior to $\mathrm{D}_{\mathrm{sKL,q}}$, however significantly ($p < 0.05$) only in micro performance values. On **WikiPersons**$_{\mathrm{E}}$, WTC is significantly superior to $\mathrm{D}_{\mathrm{sKL,q}}$ in $\mathrm{Accuracy}_{\mathrm{NIL}}$ for both learners. Considering other performance measures on **WikiPersons**$_{\mathrm{E}}$, WTC is significantly superior to $\mathrm{D}_{\mathrm{sKL,q}}$ only in micro performance and then only with the Ranking SVM as learner. Comparing WTC to WCC and cWCC we find comparable performance on **WikiPersons**$_{\mathrm{E}}$, again only the variant using the Ranking SVM stands out.

On **WikiMisc**$_{\mathrm{E}}$, we find again notably lower results with an F_{micro} of 74.92% and an F_{macro} of 73.41% using the standard SVM as learner, and an F_{micro} of 80.30% resp. F_{macro} of 79% using the Ranking SVM as learner. Comparing to the F_{micro} of 87.12% and the F_{macro} of 86.83% obtained for $\mathrm{D}_{\mathrm{sKL,q}}$ with a standard SVM, we find a notable and significant difference of consistently more than 4 pp. In any measure



(a) Buckets in a five-fold cross-validation with instance based splitting.



(b) Buckets in a five-fold cross-validation with entity based splitting.

Figure 3.12: Instance based (3.12a) and entity based (3.12b) splitting for a five fold cross-validation. The varying rectangle sizes for the entity based splitting (3.12b) indicate different bucket sizes.

apart from $\text{Accuracy}_{\text{NIL}}$, WTC is significantly inferior to $\text{D}_{\text{sKL,q}}$ on $\textbf{WikiMisc}_{\text{E}}$, either using a standard SVM or a Ranking SVM.

Therefore, even if the results with WTC are slightly higher on one dataset, we conclude that both topic based linking models are effective for person name disambiguation but that the more explicit variant over thematic distances is superior in general.

Additional Results from Entity-Based Splitting in Cross-Validation

The results published in Pilz and Paaß [2011] contain cross-validation results with entity-based splitting. Fig. 3.12 depicts cross-validation buckets created from instance based splitting and entity based splitting. Instance based splitting, as depicted in Fig. 3.12a, is the standard procedure for cross-validation. It distributes instances randomly over the buckets with the sole constraint that all buckets are of (approximately) the same size. For entity based splitting, as illustrated in Fig. 3.12b, we consider the ground truth target entities referenced in the instances and create a splitting over the ground truth entity set resulting from the complete dataset. Even though this can result in unbalanced bucket sizes, as the number of example contexts per entity varies and examples for one ground truth entity, especially NIL, all fall into one bucket, this strategy allows for additional interpretation of the model's ability to generalize, as testing instances are example contexts of entities that were not seen during training.

Due to threshold setting, this splitting strategy is not as problematic for our classification model as it is for the WCC ranking method. This method learns the

threshold for uncovered mentions from examples. However, in the entity-based splitting, documents are split by ground truth entity and examples with NIL as ground truth entity are then contained either in the training or in the test bucket. Thus, WCC can not have learned the required threshold from training data when confronted with such examples in the test data and consequently fails in the correct prediction for these entity mentions. One may argue that this renders the comparison of results for entity-based splitting somewhat unfair. On the other hand, it is a hard test on the models ability to generalize. We observed that the performance of WCC is also notably lower than that of our approach in folds without NIL instances and therefore argue that our approach has better generalization properties. For more details, we refer the interested reader to Pilz and Paaß [2011] and put here more emphasis on results obtained with entity based splitting for the corpora **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$ where we did not compare to Bunescu and Pasca's method.

Evaluation for Person Name Linking in German and French

As already stated, topic models can be build over document collections in all languages. This allows us to formulate entity linking models over thematic context distance in other languages apart from English. This can be done in a plug&play fashion, since the only necessary steps here are the creation of datasets to train the topic models and to evaluate the linking method. In the following we therefore evaluate entity linking using thematic context distance also for German and French. These are with about one million articles each two of the largest versions of Wikipedia[1]. Here, we focus on entities of type person for these two datasets **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$. To detail the obtained results, we start with the evaluation of kernel types and then give results for different splitting strategies for cross-validation. Since we observed that the quadratic kernel is superior to the linear kernel using thematic context distance $\mathrm{D}_{\mathrm{sKL}}$ on the dataset **WikiPersons**$_{\mathrm{E}}$, we also compare these two variants for the German and French datasets **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$. The obtained results are given in Tab. 3.5.

With the linear variant, we obtain averaged micro and macro F-Measures of 82.92% in F_{micro} and 82.75% in F_{macro} for the German dataset **WikiPersons**$_{\mathrm{G}}$. The corresponding values for the quadratic kernel are very similar. In fact, on this dataset, we find that the quadratic kernel results in significantly ($p < 0.05$) higher results only for macro Recall R_{macro}, micro Precision P_{micro} and the accuracy for covered entity mentions Accuracy$_{\boldsymbol{\mathcal{W}}_c}$, but significantly ($p < 0.05$) lower accuracy for uncovered entity mentions. The differences in the remaining measures are not significant. In contrast, on the smaller French data **WikiPersons**$_{\mathrm{F}}$ the averaged micro and macro F-Measures of 83.78% in F_{micro} and 83.14% in F_{macro} for the lin-

[1]Figures retrieved in July, 2013.

Table 3.5: Results on **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$ for cross-validation with instance based splitting (all values in %). The given figures are obtained using thematic context distance in a linear ($\mathrm{D}_{\mathrm{sKL,l}}$) and a quadratic kernel ($\mathrm{D}_{\mathrm{sKL,q}}$). The best value for each kernel is marked in bold and with an asterisk if the difference is significant.

	WikiPersons$_{\mathrm{G}}$		**WikiPersons**$_{\mathrm{F}}$	
measure	$\mathrm{D}_{\mathrm{sKL,l}}$	$\mathrm{D}_{\mathrm{sKL,q}}$	$\mathrm{D}_{\mathrm{sKL,l}}$	$\mathrm{D}_{\mathrm{sKL,q}}$
F_{micro}	82.92	**83.66**	83.78	**87.47***
P_{micro}	86.81	**88.61***	87.55	**91.13***
R_{micro}	**79.37**	79.23	80.32	**84.10***
F_{macro}	82.75	**83.15**	83.14	**86.94***
P_{macro}	**84.42**	84.01	84.93	**88.19***
R_{macro}	82.88	**83.95***	83.48	**87.59***
$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$	83.12	**84.65***	83.83	**87.76***
$\mathrm{Accuracy}_{\mathrm{NIL}}$	**64.56***	57.51	65.39	**67.63***

ear kernel are significantly ($p < 0.05$) smaller compared to 87.47% in F_{micro} and 86.94% in F_{macro} for the quadratic variant. Thus, the increase in performance using a quadratic kernel is only remarkable for the smaller French dataset. The observation from **WikiPersons**$_{\mathrm{G}}$ that the linear variant obtains a higher accuracy for uncovered entity mentions does not hold on **WikiPersons**$_{\mathrm{F}}$.

Generally, these results are lower compared to those obtained for the English person dataset **WikiPersons**$_{\mathrm{E}}$. We argue that this is because **WikiPersons**$_{\mathrm{G}}$ is about 2.6 times larger than **WikiPersons**$_{\mathrm{E}}$. Again, we find that the usage of a quadratic kernel ($\mathrm{D}_{\mathrm{sKL,q}}$) can yield superior results to the linear kernel ($\mathrm{D}_{\mathrm{sKL,l}}$) but this is mostly true for the smaller French dataset. Further, especially for the larger German dataset, the learning time for a linear kernel is far lower compared to the learning time for a quadratic kernel.

For both corpora our method achieves an F-measure well above 80%. We find that although we did not spend additional efforts on the specific characteristics of these languages, we can very accurately assign name phrases to the corresponding entities in Wikipedia. Thematic context distance derived from Latent Dirichlet Allocation is a language independent method: we have shown that the same approach to measure thematic context distance yields very good results for different source languages. Note that apart from the training of the LDA model, which is unsupervised, no other language specific adaptations needed to be made.

Finally, we give in Tab. 3.6 the results on the datasets **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$ that were obtained in a cross-validation with entity based splitting using the linear variant of thematic context distance $\mathrm{D}_{\mathrm{sKL,l}}$. Note that in fact

Table 3.6: Results on **WikiPersons**$_{\mathrm{G}}$ and **WikiPersons**$_{\mathrm{F}}$ obtained in cross-validation with entity based splitting and thematic context distance in a linear kernel $\mathrm{D}_{\mathrm{sKL,l}}$ (all values in %).

measure	**WikiPersons**$_{\mathrm{G}}$	**WikiPersons**$_{\mathrm{F}}$
F_{micro}	87.96	88.01
P_{micro}	97.48	97.48
R_{micro}	80.15	80.25
F_{macro}	84.91	84.86
P_{macro}	90.22	90.34
R_{macro}	82.79	82.68
$\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$	82.97	83.14
$\mathrm{Accuracy}_{\mathrm{NIL}}$	59.87	59.03

we can not directly compare these results to the variant using instance based splitting as the samples are not paired. Still, comparing Tab. 3.5 with Tab. 3.6, we see that for both datasets micro and macro Precision values are much higher in Tab. 3.6, i.e. about 10 pp for P_{micro} and about 6 pp for P_{macro}. While the accuracy for covered entities $\mathrm{Accuracy}_{\boldsymbol{\mathcal{W}}_c}$ remains more or less the same, the accuracy for uncovered entity mentions $\mathrm{Accuracy}_{\mathrm{NIL}}$ drops by about 5 pp. In this splitting strategy, we observe a higher standard deviation of about 5% for most measures over the folds, whereas for the instance based splitting this figure is consistently less than 1%. This is because examples for uncovered entity mentions are all contained in one fold and since these examples are more difficult to handle than covered entities, performance drops significantly on this fold whereas the folds containing only examples of covered entity mentions show a consistently higher performance. This is noteworthy to avoid misleading interpretation of the results. To summarize this experiment, we conclude that the performance of our method is not affected when confronted with completely new entity sets. This is an important results since it empirically proves our method's ability to generalize.

Srinivasan et al. [2009] generate words according to a topic model and then add these to the respective feature vector to overcome the synonymy problem. In contrast, our approach relies on the overall topic probability distribution of a document, thus using a completely different feature vector representation based on topic clusters instead of words.

The methods proposed and evaluated in this section rely on the textual content provided by Wikipedia. Since depending only on observed words, topic models are not language specific and can naturally be employed to estimate topic distributions in various languages. The proposed method is also independent of the Wikipedia specific category system and depends only on corpus size. Further, while filtering

categories requires understanding of the language, training a topic model does not, only when aiming for a manual inspection and interpretation of the formed word clusters. If a specific version of Wikipedia does not provide enough textual content, we might acquire content by crawling news pages in the respective language. However, in that case we cannot reason on the performance of the model since the topic model might then not reflect the articles in the Wikipedia version of interest.

For future work, it will be interesting to exploit more sophisticated variants of LDA. Some variants allow the incorporation of background knowledge to account for additional structures and priors over words and documents Andrzejewski et al. [2009], Steyvers et al. [2011], Newman et al. [2011]. Polylingual topic models (Mimno et al. [2009]) might be useful for knowledge transfer among different Wikipedias. Other variants of LDA, such as the method proposed in Wahabzada et al. [2011], allow faster learning over larger datasets, an asset that may be useful when handling more diverse reference contexts. Alternatively to LDA, we note that the continuous word representations recently proposed by Mikolov et al. [2013] should also be investigated. These word representations are computed from the hidden layers in a neural network and belong to the deep learning techniques that have recently achieved enormous attention in NLP and various other fields. For instance, they may be used to further enhance context representation but also to learn a new form of entity profiles.

Having evaluated and discussed our approach to person name linking, we will now give a brief overview on approaches to Named Entity Linking. Since we directly generalize to arbitrary entity types in the following chapter, this section serves as a connection and highlights the major findings of the relevant approaches. Important aspects will be discussed again in the following chapter.

3.7 An Excursion into Named Entity Linking

In this chapter we have focussed mainly on the linking of person names. Although we have shown in our experiments that thematic context distance can achieve superior results on a dataset containing other types of entity names, i.e. **WikiMisc**$_{\text{E}}$, we have not directly evaluated the performance for entities different than persons. One natural next step would therefore be named entity linking. Named entity linking extends person name linking and usually includes locations and organizations. Named entity linking has been widely studied in recent years and has also been one focus of the Knowledge Base Population shared tasks at the Text Analysis Conferences (TAC) since 2009 (McNamee and Dang [2009], Ji et al. [2010, 2011]).

Hachey et al. [2013] thoroughly compared the most successful approaches of 2009 (Varma et al. [2009]) and 2010 (Lehmann et al. [2010]) against those of Bunescu and Pasca [2006] and Cucerzan [2007]. Since we have no access to the either of

the employed datasets[1], we here summarize the findings of Hachey et al.'s overview concerning the dataset from 2010. The TAC 2010 dataset is a collection of Reuters news articles and web pages containing named entity mentions that are to be linked against a snapshot of Wikipedia articles (from 2008) or to NIL if the underlying entity is not covered in this snapshot.

Varma et al. [2009] achieved the best results in the TAC 2009 challenge. They used a carefully constructed candidate selection method with in-document co-reference resolution for acronym expansion in combination with a rather simple candidate consolidation method that maximizes the cosine of mention context and candidate context. Their candidate selection method uses an inverted index over alias names against which mentions are matched both token- as well as phrase-wise. Lehmann et al. [2010] use a similar technique but achieve a higher candidate recall. The presumably most important difference is that Lehmann et al. [2010] also use alias information derived from links which gives them not only more aliases but also enables the usage of priors such as entity-mention probability. The candidate consolidation of Lehmann et al. [2010] is a heuristic ranking over features such as alias trustworthiness, the similarity between mention and candidate name and the matching of mention and candidate type. It also includes a supervised binary logistic classifier used for NIL detection. Employing a separate classifier for NIL detection was also reported by Zheng et al. [2010] to slightly increase the results of Varma et al. [2009] on the TAC 2009 corpus.

Interestingly, Hachey et al. found the re-implementation of Cucerzan [2007] superior to that of Varma et al. [2009] (81.6% vs. 84.5% accuracy), as the former achieved a much higher accuracy for covered entity mentions. We assume that this is due to the collective nature of Cucerzan's approach which can be superior to the simpler contextual similarity method of Varma et al.. Hachey et al. argue that this can also result from the nature of the dataset. Varma et al. specialised in organisation and acronym handling but the number of respective mentions is far lower on the TAC 2010 dataset, i.e. 21% in the dataset from 2009 vs. 15% in the dataset from 2010. However, both methods gain from in-document co-reference resolution both for acronyms as well as other mentions. This may also explain why Bunescu and Pasca's method showed with an accuracy of 80.8% the weakest overall performance. Bunescu and Pasca neither use in-document co-reference resolution nor candidate selection methods as elaborate as Cucerzan or Varma et al.. Furthermore, as described in this chapter, Bunescu and Pasca's approach was originally designed for person name linking, while the TAC dataset also includes other entity types. Hachey et al. tried to account for this by generalizing the employed category set for their implementation to different top-level category sets.

[1]The datasets are available only to the participants of the challenge. When asking for the data, the consortium would give allowance if we participate in the upcoming challenge which at that time was unfortunately not possible.

Hachey et al. compared all methods against two baselines. The first is a title&redirect baseline that uses exact matches of mentions against Wikipedia titles and redirects. The second is a NIL-baseline that assigns every mention to NIL. Interestingly, the title&redirect baseline was found to achieve an overall accuracy of 79.4% and the NIL-baseline arrived at an accuracy of 54.7%. The latter is due to the even distribution of covered and uncovered entity mentions in this dataset. For person mentions in news texts, the title&redirect baseline was found to achieve a nearly perfect accuracy of 97.0%. Hachey et al. attribute this to editorial standards in news, which lead to entity mentions in their most common form and thus mentions close to Wikipedia titles. However, most of these mentions also truly referred to an entity in Wikipedia. In contrast, this baseline showed with 82% a far weaker accuracy for person mentions in web texts. Unfortunately, since we miss important statistics and also don't know the average ambiguity of these person mentions we can not further judge these results.

Hachey et al.'s evaluation allows for several important insights that go along with the experimental results obtained in this thesis. The performance of linking approaches need not generalize across datasets and may strongly depend on the number of uncovered entity mentions but also the distribution over entity types of mentions. This also concerns the number of examples for uncovered entity mentions in the training dataset since this fraction usually influences model parameters and thus also the performance on test datasets.

In an analysis of the systems performance broken down by entity type, Hachey et al. found that all systems perform best for persons, with remarkably lower results for organizations and geopolitical entities (about 20% lower accuracy). As no approach was found to perform consistently superior across document type (news or blog) and entity types, the authors suggest the combination of entity specific models or voting combinations.

Locations and organisations as well as their mentions have different characteristics than persons. First off, locations may be mentioned off-topic as geographic anchors, e.g. as a reference in a news article reporting on some sports event. In such cases, the reference context may not provide enough evidence to distinguish among mentions of Lincoln, Ontario, Lincoln, Alabama or Lincoln, Kansas. Furthermore, the article texts of locations usually describe historical, geographical or political characteristics and do in most cases not thematically relate to reference contexts. Notably, a mention Lincoln may also refer to a person (Abraham Lincoln), an educational institution (University of Lincoln), an English football club (Lincoln City F.C.) or many more candidate entities. Approaches restricted to specific entity types (e.g. persons) may then further suffer from potential errors of NER models.

Similar characteristics apply to organisations and probably most difficult are sports associations for which we often find not only natural language text but also many tables in the article text. Tables are inherently relational and we will approach

them in a more relational approach that treats relations more explicitly than the LDA topic modelling technique.

Instead of focussing on named entity's, we will investigate general entity linking in the next chapter. General entity linking covers named entity linking but goes a step further by treating all kinds of entities, i.e. concepts usually denoted by noun phrases. We aim for a collective approach that may gain from interactions among and across all kinds of entities.

The candidate retrieval methods of Varma et al. [2009] and Lehmann et al. [2010] have been an inspiration for the method we will present in the following chapter. For general entity linking, we will extend them with relational information derived from co-occurrences of entities in ensemble queries treating all mentions in a document.

3.8 Summary

In this chapter, we focussed on persons as an entity type with highly ambiguous names and proposed entity linking models using topic models. We evaluated topics as semantic labels for the disambiguation of person names in German and, inspired by the promising results, generalized the usage of topic models to derive thematic context distances over describing contexts. Relying on the distance over topic distributions instead of descriptive word-vectors, this method can inherently handle synonymy and polysemy which is not the case for methods based on direct word comparison. While overly sparse text representations such as WTC or WCC may often perform well, such approaches can not grasp the similarity between terms like *splendid* and *terrific* and also often have a longer learning time.

We evaluated our method on reference data from Wikipedia in English, German and French and showed that similarity measures computed over latent topics are especially suitable to link mentions of persons to their underlying entities. Being more general than word based distances, the proposed thematic distances allow to exploit the thematic overlap between referring contexts and the biographic content of articles describing persons.

We have compared our approach to the most related method of Bunescu and Pasca [2006] and shown in detail that our method can significantly ($p < 0.05$) increase performance and improve the assignment of name mentions to the underlying articles in Wikipedia. Treating also mentions of entities that are not covered by an article in Wikipedia, we have shown that our method can handle this problem very accurately. This is a crucial aspect: When we retrieve information for a known entity, we don't want to assign false facts to it. Comparing to the Wikipedia category based approach of Bunescu and Pasca [2006] or Cucerzan [2007], our approach is furthermore more flexible and applicable to different languages without expensive manual category analysis. At the time of publication, this method was the first to approach entity linking in multiple languages.

In this chapter, we focused on person name disambiguation in a purely contextual approach with simple matching techniques for candidate retrieval. As described in the overview on named entity linking, a straightforward match of mentions against Wikipedia titles or redirects can yield more than satisfactory results. Especially in edited news paper articles, persons are often mentioned with canonical names which may render candidate retrieval less crucial for persons. However, when generalizing to other entities we need more elaborate candidate retrieval techniques, for example to handle abbreviations. This is the subject of the next chapter where we will extend from a context based approach to a more collective, relational method.

Chapter 4

Local and Global Search for Entity Linking

Outline

In the previous chapter we focused on the consolidation part of entity linking, especially for mentions of persons, and treated each mention instance individually. In this chapter, we generalize entity linking to arbitrary entity types and introduce a *global* view on the *document level* by collectively linking the mentions in a document and doing so, focus more on the candidate retrieval part of entity linking.

We first introduce general entity linking that considers both named entities as well as abstract concepts (Section 4.1) and give an overview of related work with focus on recent collective approaches that investigate linking to Wikipedia (Section 4.2). We then describe our approach, a data driven method that exploits the structured and unstructured information encoded in Wikipedia by a carefully constructed search index (Section 4.3). The description of the proposed multi-stage algorithm starts with a brief summary (Section 4.4) that outlines the subsequent sections. Having described how mentions are enriched with various attributes used for linking (Section 4.5), we detail the stages of our entity linking algorithm. We propose a novel candidate retrieval method that collectively uses all mentions in a document and exploits the co-occurrence of links in Wikipedia. We assess relatedness through the collective fitness of candidate entities in the document in a novel coherence measure. Based on this coherence, we compute the best fitting candidate for each mention and combine this prioritization with local, contextual information in a second stage (Section 4.6). Finally, candidates are consolidated by a supervised ranking SVM (Section 4.7). The method is evaluated in an unsupervised (Section 4.8.2) as well as in a supervised variant (Section 4.8.4) on five different benchmark corpora.

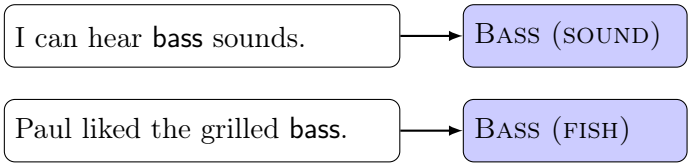
This chapter covers the ideas and findings published in Pilz and Paaß [2012] and provides additional experimental evaluation to demonstrate the performance of the proposed method.

4.1 General Entity Linking

In this chapter, we aim at linking mentions of both concrete *named entities* as well as *abstract entities* or *concepts*. Doing so, we generalize from named entity linking or person name disambiguation to general entity linking. Note that even though conceptual entities are usually referenced by proper nouns, this task overlaps closely with word sense disambiguation. The latter aims at resolving ambiguity for all common words in text, e.g. adjectives, verbs and nouns, but does not necessarily include proper nouns and named entities.

Mallery [1988] termed word sense disambiguation an AI-complete problem that requires not only deep linguistic knowledge but often also world knowledge. For illustration, we give the following example, a modified version of the one given in Navigli [2009].

Example 8 (Word sense disambiguation)
Take the following two mentions of bass that denote two distinct concepts:



In the first sentence, the mention bass denotes low-frequency tones, i.e. the concept Bass (sound). The second mention refers to a type of fish, i.e. Bass (fish).

For a human reader, the hints provided in these short sentences above are sufficient to grasp the intended meaning of each mention. The respective sense of each mention is implied through the co-occurring context terms: *hear* and *sounds* hint at the concept Bass (sound), *grilled* hints at the concept Bass (fish). However, for the automatic inference of the intended senses, the available contextual evidence is rather poor. Some model would be required to reason on the relation between *hear*, *sounds* and *bass* to infer the concept of sound, likewise for *grilled* and the concept of fish. These relations are not explicitly given in the text but need to be inferred or learned from background or world knowledge, for example from statistics over co-occurring terms.

Mihalcea and Csomai [2007] define word sense disambiguation as the automatic assignment of the most appropriate sense to a word within a given context. This sense is taken from an inventory that is often assumed to be complete. Originally, the major sense inventory in word sense disambiguation was WordNet and ambiguity was resolved by assigning a word to a specific set of synonyms (i.e. a *synset*) in WordNet

(Navigli [2009]). Note that for entity linking, the assumption of completeness is not appropriate since for instance there is no inventory covering all persons in the world. Also, while there may be multiple senses for verbs or nouns, the degree of polysemy of person names is notably higher. For example, the average number of candidates for polysemous nouns in WordNet is 2.79[1], whereas the number of candidates for person names can easily exceed 20 (as shown in Tab. 1.1).

Furthermore, in word sense disambiguation, a mention may refer to a conceptual entity such as BASS (FISH) that subsumes all individuals belonging to this species of fish. These individuals are usually not distinguishable by a rigid designator. For instance, in the example above, the mention bass refers to a specific, existing real-world entity: the bass on Paul's plate. The given context distinguishes the fish on Paul's plate from all other fishes of species BASS and therefore defines it's uniqueness. However, this entity has no rigid designator and in practice we cannot distinguish among all individuals of the BASS (FISH) species. Thus, instead of generating a unique pseudo-identifier such as *grilled_bass_on_pauls_plate_281282* to ground the mention bass, we propose to link it to the conceptual entity BASS (FISH) that comprises all individuals of this species. We argue that this link resolves the ambiguity of the mention as it distinguishes it from the abstract entity BASS (SOUND).

While not explicitly excluding adjectives or verbs, we here focus on entities or concepts usually denoted by noun phrases. This is more general than named entity disambiguation since we aim at linking mentions independent of their type while still taking into account uncovered entity mentions.

Having introduced word sense disambiguation and general entity linking, we will now give an overview of the relevant related work.

4.2 Related Work: General Entity Linking

In this section, we will give an overview of the most relevant related work in entity linking. We will also introduce the benchmark corpora published by those approaches and simultaneously describe and discuss the employed evaluation techniques. Most of these corpora consist of English newspaper articles from different time periods where the major difference lays in the annotation scheme. Depending on the intentions of the authors, some corpora are annotated with mentions of various entity types including uncovered entity mentions, others contain only covered named entity mentions. While all approaches propose accuracy related measures, there are different aspects of interest, such as the averaged accuracy per mention, the averaged accuracy per entity or the accuracy towards uncovered entities. This results in a small variety of performance measures which we need to discuss in order to lay the ground for a better comparability of the results presented in this thesis.

[1]Retrieved in July, 2014 for WordNet 3.0, `http://wordnet.princeton.edu/wordnet`

We start with Mihalcea and Csomai [2007], who presented *Wikify!*, the first Wikipedia based word sense disambiguation approach. *Wikify!* detects and links keyphrases, where a keyphrase can be any kind of term. To detect candidates for link anchor texts, the authors introduce *link probability*. Link probability approximates the probability of a phrase m being used as a link anchor text l_a through the ratio of articles containing m as link anchor text l_a and the overall number of articles containing m, i.e. the document frequency of m. The best linking performance is obtained with a Naive Bayes classifier (for details on Naive Bayes see Russell and Norvig [2003]) that uses the following features: the candidate phrase together with a local context of three words around it, the part of speech tags of these words and other sense specific keywords that often co-occur with a link target candidate. This method is reported to achieve an F-measure of 88% in an evaluation on 7286 links extracted from Wikipedia.

Milne and Witten [2008b] extend the approach of Mihalcea and Csomai [2007] through the incorporation of semantic relatedness among candidate entities. This relatedness is the averaged SRL (Milne and Witten [2008a], cf. Eq. 2.2) of an ambiguous candidate entity towards other, unambiguous candidate entities in the document, weighted by their individual link probability. To compute relatedness, the authors compare each possible candidate with its surrounding context formed from the other candidates in the document. To eliminate context terms that do not relate to the central subject of the document, they calculate its average semantic relatedness to all other context terms, using the above relatedness measure. The sum of the weights previously assigned to each context term is used as context quality feature. The relatedness among candidates accounts for their coherence but is restricted to unambiguous candidates. Assuming that a sufficiently long text contains a certain amount of unambiguous terms and in order to avoid cycles, the authors compute relatedness of ambiguous candidates only towards unambiguous candidates. Using EMP (cf. Eq. 2.7), the above described context quality and the relatedness of each candidate as features, the authors evaluate different classifiers for candidate selection. Comparing Naive Bayes, C4.5 (Quinlan [1993]) and SVM as potential classification algorithms they found C4.5 to give the best result even though it should be noted that the individual F-measures were not strikingly different. The authors argue that Naive Bayes performs worst because of the inter-dependencies of the used features. The classifier is trained and evaluated on Wikipedia references[1]. With an F-measure of 97.1%, the proposed method using C4.5 is found to be superior to a maximum prior baseline (90.7% in F-measure) as well the heuristic baseline by Medelyan et al. [2008] (92.9% in F-measure). The authors argue that their approach is superior since the system described in Medelyan et al. [2008] uses no machine learning and no context weighting.

The approach is also evaluated on the **AQUAINT** corpus that was made publicly

[1]Version of November 20, 2007.

available by the authors. **AQUAINT** is a collection of 50 documents from the AQUAINT corpus of English news-wire stories[1]. As the authors annotated only the first mention of important or interesting entities in the document, the annotation scheme of this corpus is similar to that in Wikipedia. This amounts to 727 mentions of both named as well as conceptual entities. Notably, since the authors do not focus on uncovered entities, this news corpus also contains no mentions with uncovered entities as ground truth targets. On **AQUAINT**, Milne and Witten [2008b] report a linking accuracy of 76.4% for their proposed method.

Ratinov et al. [2011] found superior performance on this corpus for their proposed model GLOW. Similar to the approach of Milne and Witten [2008b], GLOW is an approximation to joint disambiguation and collective information is again encoded in the semantic relatedness among candidates. To emphasise the coherence among candidates, the authors extract additional named entity mentions and noun phrases from the document that were found to be previously used as link anchor texts in Wikipedia. Augmenting the given query mentions with this set, candidates are then retrieved by querying an anchor-title index that maps each link target in Wikipedia to its various link anchor texts and vice versa. Using this overall candidate set, the best candidate is then predicted individually per mention by a Ranking SVM. This model is trained on Wikipedia links and uses textual similarity weights, EMP (Eq. 2.7) and popularity prior (Eq. 2.8) as local features. Additional global features are in- and outlink based relatedness of the candidates, i.e. SRL (Eq. 2.2) and SRL_{out} (Eq. 2.5), in different weighting schemes. Since the prediction of the Ranking SVM is always a Wikipedia entity, the authors additionally employ a linear SVM to decide whether the Ranking SVM's prediction should be switched to NIL. Both models are trained on Wikipedia references. The second classifier is trained on the predictions of the Ranking SVM on this corpus, using as features the confidence of the Ranking SVM, a boolean value encoding whether a mention m is a named entity and link statistics of m.

The authors report superior performance for GLOW to the API version of Milne and Witten's system on the corpora **AQUAINT**, **ACE** and **MSNBC**. **ACE** is a selection of 36 documents from the ACE co-reference dataset[2], where named entity mentions are manually annotated with Wikipedia targets or NIL. The corpus contains mostly mentions of locations but also persons and organizations as well as a substantial amount of conceptual entities.

Originally published by Cucerzan [2007], **MSNBC** is a collection of 20 English news stories of various topics, covering among others business, health and sports topics. **MSNBC** is annotated with ground truth entities for all mentions of persons, locations, organizations, and entities of miscellaneous and conceptual type using the Wikipedia version from April 2, 2006. Here, not only the first mention is annotated

[1]The full corpus is under license at `https://catalog.ldc.upenn.edu/LDC2002T31`.
[2]The full corpus is under license at `https://catalog.ldc.upenn.edu/LDC2005T33`.

but all subsequent ones.

The main difference among **MSNBC**, **ACE** and **AQUAINT** is that the first two corpora contain notably more mentions per document. This is presumably helpful for GLOW and its candidate retrieval model and may explain why the model performs remarkably better than that of Milne and Witten [2008b]. Also, Ratinov et al. [2011] spend much more effort on NIL detection. This can be the reason that GLOW performs better on corpora containing such mentions, whereas performance is comparable on **AQUAINT**, a corpus that contains no uncovered entity mentions.

Unfortunately, it is not possible to directly compare the figures published for GLOW with those of Cucerzan [2007] since different evaluation measures are used. For the sake of completeness, we note that, taking into account both covered and uncovered mentions, Cucerzan [2007] reported an accuracy of 91.4% for their collective method on **MSNBC**.

Ratinov et al. [2011] used **Bag-of-Titles** (**BoT**) evaluation. This evaluation method compares the predicted set of entities with the ground truth set of entities, ignoring duplicates in either set, and further utilizes standard Precision, Recall, and F-measure that we denote with P_{BoT}, R_{BoT} and F_{BoT} respectively.

For illustration, we take the example from their paper.

Example 9 (BoT evaluation)
The collection of ground truth annotations

$$e^+(m_1 = \mathsf{China}) = \textsc{People's Republic of China}$$
$$e^+(m_2 = \mathsf{Taiwan}) = \textsc{Taiwan}$$
$$e^+(m_3 = \mathsf{Jiangsu}) = \textsc{Jiangsu}$$

has gold-BoT = {People's Republic of China, Taiwan, Jiangsu}.
According to Ratinov et al., the set of predictions

$$\hat{e}(m_1 = \mathsf{China}) = \textsc{People's Republic of China}$$
$$\hat{e}(m_2 = \mathsf{Taiwan}) = \text{NIL}$$
$$\hat{e}(m_3 = \mathsf{Jiangsu}) = \textsc{Jiangsu}$$
$$\hat{e}(m_4 = \mathsf{China}) = \textsc{History of China}$$

has BoT = {People's Republic of China, History of China, Jiangsu} and Precision and Recall of $P_{\text{BoT}} = R_{\text{BoT}} = 0.66$. This calculates from two true positives for People's Rep. of China and Jiangsu and the false positive prediction History of China for the additional mention m_4 of China. The latter is taken into account since the associated ground truth entity China appears in the gold-BoT. Note that the predicted BoT does not include the NIL prediction which is consequently counted only as a false negative for Taiwan.

Technically, this measure corresponds to the micro performance described in Section 3.5.4 so long as each entity appears only once in the ground truth set. The ignorance of duplicate entities however is here necessary because GLOW extends the ground truth set with additional mentions and their respective entity predictions. Therefore, the predicted BoT may contain more entities from which only those are taken into account that appear in the ground truth set. Consequently, BoT also ignores the frequency of ground truth entities. This may thus obscure both erroneous as well as correct predictions. For instance, if an entity appears five times in the gold annotation and the disambiguation model fails to resolve it correctly, the number of false negatives is only one in BoT, whereas it would be five if all instances were considered. As this holds analogously also for the number of true positives, this measure accounts for the overall accuracy of all mentions and, similar to micro performance, treats all entities equally, independent of their frequency. Also, incorrect predictions of NIL are not counted as false positives. For the remainder, we assume that BoT takes the sequential order of ground truth entities into account and penalises any change in order when aligning predictions with the ground truth set.

GLOW is the most similar to the method we propose in this chapter, especially considering the usage of inverted indices and the combination of local and global information. A variant of GLOW achieved the fourth place in the TAC 2011 KBP entity linking challenge (Ratinov and Roth [2011]). We show in the experimental section of this chapter that our method outperforms GLOW on all of the above benchmark corpora.

Shen et al. [2012] present LINDEN, a system that links given named entity mentions to YAGO. Along with previous work, the authors investigate coherence among possible candidate entities. Similar to Milne and Witten and Ratinov et al., they use EMP and a variant of SRL as features and propose two new features. The first feature is the semantic similarity of candidates with respect to the types in the YAGO ontology. This feature assumes a tree structure of categories applying to candidate senses. Note that such a feature can only be obtained from well processed knowledge bases with a strict type hierarchy such as YAGO. In contrast, the original Wikipedia category system is not a tree but may contain cycles. The second new feature is the global coherence of candidates for mentions in the document, where the global coherence of one candidate is the average SRL to other candidates. These four features are used in a linear SVM that is evaluated in cross-validations on a variant of the **MSNBC** corpus and data from the TAC 2009 knowledge base population task (McNamee and Dang [2009]). Thus, for each corpus the model parameters are determined individually. As the TAC 2009 corpus contains many documents with only one mention, the coherence feature was deemed useless and consequently removed. For **MSNBC**, the authors found EMP along with the link based SRL feature to be the most influential.

However, it is not fair to compare the performance reported by Shen et al. [2012]

with that of other methods tackling **MSNBC**, since Shen et al. removed documents as well as 18% of the given mentions to be linked and thus ignored many linking decisions. This unfortunately applies also to the results reported by Dredze et al. [2010] for **MSNBC**, as these are obtained after removing 297 mentions from **MSNBC** that do not refer to named entities.

Shen et al. also approach uncovered entity mentions. According to the authors, the system returns NIL if no candidate can be retrieved. If there is only one candidate, this candidate is set as prediction. If the number of candidates exceeds one, the authors use the ranking based on the features described above and choose as prediction the candidate with maximum score. This score needs to exceed a threshold τ, otherwise the prediction is set to NIL. Unfortunately, the authors do not state how the proclaimed learning of the threshold τ is performed or give any empirically determined value.

The **MSNBC** corpus was also used in Kulkarni et al. [2009], who formulate collective entity linking as a joint optimization problem in a probabilistic graphical model. Based on the pairwise relatedness among all candidates for a given mention, they aim at assigning entities to mentions such that the mention-entity compatibility and global entity-entity coherence is maximized. Since estimating the maximum a posteriori joint probability distribution is shown to be computationally too expensive, the authors propose a linear programming and local hill-climbing relaxations for optimization. The method yields favourable results on **MSNBC** as well as on the dataset **IITB**, the latter created and published by the authors. **IITB** is composed of 104 web documents and richly annotated, aiming for aggressive linkage. It contains named entity mentions and with about 85% a large number of conceptual entity mentions. It contains no ground truth annotations resolving to NIL.

On both datasets, Kulkarni et al. compared against the API version of Milne and Witten's algorithm and an implementation of Cucerzan's method and found their collective method superior. Interestingly, they found that a local model using only contextual similarity without collective inference performed better than the prior approaches of Milne and Witten and Cucerzan on the **IITB** dataset. However, it should be noted that the results reported for Kulkarni et al.'s implementation of Cucerzan's algorithm gave remarkably different results on **MSNBC** compared to the originally published ones. Also, unfortunately, the reported statistics on the **MSNBC** dataset differ from those given in the original publication and also from the statistics we extracted for this dataset. Documents as well as mentions seem to be missing, which reduces the comparability of the reported results.

The evaluation scheme used in Kulkarni et al. [2009] is comparable to BoT but accounts more explicitly for false positive NIL-predictions in Precision. To distinguish this evaluation scheme from BoT, we will use BoT* as subscript. More specifically, let $\{\hat{e} = e^+ \in \boldsymbol{\mathcal{W}}\}$ and $\{\hat{e} \neq e^+ \in \boldsymbol{\mathcal{W}}\}$ denote the sets of correct respectively incorrect predictions for covered entities and $\{\hat{e} = \text{NIL}\}$ denote the set of assignments to

uncovered entities. Then, Kulkarni et al. define

$$P_{\text{BoT}^*} = \frac{|\{\hat{e} = e^+ \in \boldsymbol{\mathcal{W}}\}|}{|\{\hat{e} = e^+ \in \boldsymbol{\mathcal{W}}\}| + |\{\hat{e} \neq e^+ \in \boldsymbol{\mathcal{W}}\}| + |\{\hat{e} = \text{NIL}\}|} \tag{4.1}$$

$$R_{\text{BoT}^*} = \frac{|\{\hat{e} = e^+ \in \boldsymbol{\mathcal{W}}\}|}{|\{e^+ \in \boldsymbol{\mathcal{W}}\}|} \tag{4.2}$$

$$F_{\text{BoT}^*} = \frac{2 \cdot P_{\text{BoT}^*} \cdot R_{\text{BoT}^*}}{P_{\text{BoT}^*} + R_{\text{BoT}^*}}. \tag{4.3}$$

Aiming at aggressive linkage for covered entities, the focus is here on the Recall for covered entities as implied by the denominator in the Recall formula (Eq. 4.2). Kulkarni et al. also do not focus on the models accuracy concerning the detection of uncovered entity mentions as implied both by the evaluation scheme and the fact that none of the treated datasets contains NIL as ground truth target. Thus, the motivation behind this approach somewhat differs from ours. Nevertheless, we will compare our method to it given that this approach is one of the first and most cited for general entity linking against Wikipedia.

Han et al. [2011] propose a graph-based collective entity linking method that exploits the global interdependence between different entity linking decisions. In this graphical model, mentions and entities are nodes connected via weighted edges. The edges between entities are weighted by their SRL, the edges between mentions and entities are weighted through the cosine similarity of the local mention context, 50 words window, and the candidate entity's article texts. Instead of Kulkarni et al.'s pairwise relatedness, the authors estimate truly on the document level, assigning an entity to a mention based on the product of local compatibility, i.e. text similarity, and evidence scores from all other related assignments. These evidence scores are inferred in a Personalized PageRank (Haveliwala [2002]) using the product of an initial vector of context similarities and a transition or evidence propagation matrix capturing textual similarity and relatedness among all candidates. Since the transition matrix needs to be inverted, candidate selection is crucial to avoid the inversion of a large matrix. Here, candidates are selected from link anchor text information.

For evaluation, the authors state that the TAC 2009 corpus is unsuitable, since there is only one mention per document to be linked and collective approaches are thus deemed useless. Instead, they compare to Kulkarni et al. [2009] on **IITB** and find an improvement of 4% in F_{BoT^*} compared to the result of 69% reported by Kulkarni et al.. They demonstrated performance improvements, however, they did not take into account uncovered entities and evaluate performance using only the name mentions whose underlying entities are contained in Wikipedia.

Using a generative entity topic model that integrates topic coherence, assuming one topic per document, and local context compatibility, the authors could increase the F_{BoT^*} on this corpus to 80% (Han and Sun [2012]). Again, only mentions with underlying entities contained in Wikipedia are evaluated.

Hoffart et al. [2011b] introduce AIDA, a system that links named entity mentions towards the entity catalogue YAGO2 (Hoffart et al. [2011a]), an extension of YAGO, by maximizing the weighted sum of prior probability, contextual similarity and candidate coherence in a greedy search strategy over related candidate entities. To do so, they build an undirected mention-entity graph with on demand computed edge weights. Similar to Han et al. [2011], edges between mention and entity nodes are weighted by context similarity and EMP (Eq. 2.7), edges between entity nodes are weighted by their coherence. The coherence among candidates is based on SRL (Eq. 2.2) and context similarity is computed by matching the mention context against weighted entity keyphrases derived, among others, from article text, categories, inlinks and external references.

From this graph, they iteratively remove the entity node and all its incident edges with the smallest weighted degree, until there is no more removable entity node. This maximizes the overall objective function as the weighted degree of a node is the total weight of its incident edges. In a post-processing step, they remove remaining edges relating one mention to several entities. This greedy algorithm identifies a dense sub-graph with exactly one edge per mention-entity pair and is assumed to yield the most likely entity predictions.

The weighting components are selectively used based on some heuristics and automated prior and coherence tests. The thresholds for the tests are determined together with their weights in an objective sum using line search on development data. Interestingly, similarity and popularity prior receive with 0.43 and 0.47 the highest weights, compared to the far less influential weight of 0.1 for coherence. The heuristics state that the prior is only used when all candidates for every mention have popularity prior higher than 0.9 and that coherence is only used when the sum of dis-agreements between the popularity priors and the context similarities for all mention-entity pairs exceeds a learned threshold of 0.9, otherwise only the entity with maximum prior and context similarity is added as a node to the graph.

The method is trained and evaluated on named entity mentions in the CoNLL 2003 corpus, a collection of 1393 Reuters news articles[1]. The link annotations of the corpus were made publicly available by the authors, however the text is part of the CoNLL 2003 shared task (Sang and Meulder [2003]) and has restricted access. Due to previous work on NER, we had access to these documents and, in line with Hoffart et al. [2011b], we will consider the 228 documents from the test set **CoNLLb**. The annotation strategy of the authors is that only those named entity mentions that can be automatically mapped to YAGO2, are linked[2]. Abbreviations such as EU that are not handled properly by this mapping are assigned NIL and ignored in evaluation, even though the presumably correct entity would be European Union. For **CoNLLb** this results in a gold annotation set of 4363 named

[1]The corpus is under restricted access available at `http://www.cnts.ua.ac.be/conll2003/ner`

[2]Since YAGO2 is derived from Wikipedia, these link targets are equivalent to Wikipedia targets.

entity mentions with persons (977), locations (1388), organisations (1458) and entities of miscellaneousness type (540). Interestingly, the inter-annotator agreement on the full corpus, as reported by the authors, was with 78.9% notably below the best performance measure reported by the authors.

To discuss results, we will first describe one of the used performance measures. Originally a measure from Information Retrieval, the authors use Mean Average Precision (MAP) that is defined as

$$\text{MAP} = \frac{1}{m} \sum_{i=1}^{m} p@\frac{i}{m}, \tag{4.4}$$

where $p@\frac{i}{m}$ is the Precision at a specific Recall level. Here, Recall is related to the position in the output list and not the number of false negatives. This position is computed from the model output and ranked according to some confidence score s. This means that highly confidential assignments of entities to mentions are ranked at leading positions and predictions with low confidence at late positions. Assuming that incorrect predictions have in general a low confidence, MAP thus shuffles erroneous predictions to the end of the ranked output list. As a consequence, the sum is dominated by correct predictions with presumably high confidence at the top of the ranking, which are propagated through the whole list. This is can be of great importance when the number of mentions in a document is especially large.

For further illustration, we will compare MAP to BoT in the following example and show that even for a small number of mentions we may observe notable differences.

Example 10 (MAP evaluation)
Assume a list of predictions

$$\{s(\hat{e}(m_3)) = 0.9, s(\hat{e}(m_2)) = 0.8, s(\hat{e}(m_1)) = 0.2\}$$

sorted by the magnitude of a confidence value s instead of order of appearance. Assume all predictions to be correct apart from $\hat{e}(m_1) \neq e^+(m_1)$. The summands in Eq. 4.4 are computed using the ranking induced by s. This leads to

$$\begin{aligned} s(\hat{e}(m_3)) = 0.9 &\rightarrow p@1 = 1/1 \\ s(\hat{e}(m_2)) = 0.8 &\rightarrow p@2 = 2/2 \\ s(\hat{e}(m_1)) = 0.2 &\rightarrow p@3 = 2/3. \end{aligned}$$

As by Eq. 4.4, the MAP is the sum of these values divided by the number of mentions, i.e.

$$\text{MAP} = 1/3\,(1/1 + 2/2 + 2/3) = 8/9.$$

The BoT performance for this example is $P_{\text{BoT}} = R_{\text{BoT}} = F_{\text{BoT}} = 2/3$, and in this case corresponds to standard accuracy. Also, note that when we follow the

> sequential input order, i.e. ignore the sorting induced by confidence s and count only the number of correct predictions, the result is
>
> $$\text{MAP}_{\text{order}} = 1/3\,(0 + 1/2 + 2/3) = 7/9.$$

In the example above, we have for one predicted outcome three possible performance values ranging from 66% to 88%, depending on the calculation rule. From this we see that the comparison of methods using different performance measures is not straightforward. We carefully need to take into account seemingly minor differences that result, at least on the first glance, in sometimes strikingly different values. Now, having described MAP evaluation in detail, we will use this measure for comparison with Hoffart et al. [2011b]. The authors also proposed other measures but unfortunately their computation was not thoroughly described and remained vague.

Hoffart et al. [2011b] compared AIDA on **CoNLLb** to the methods proposed by Kulkarni et al. [2009] and Cucerzan [2007] and reported with a MAP of 87.31% superior performance to that of their re-implementations of Kulkarni et al. (85.44% in MAP) and Cucerzan (40.06% in MAP). This is obtained with the best configuration of AIDA that uses popularity prior with robustness test, keyphrase based similarity and graph coherence with robustness test. Interestingly, a popularity prior baseline was reported to achieve a MAP value of 86.63% on **CoNLLb**. To summarize, AIDA performs favourably for named entity mentions. However, it completely ignores mentions of uncovered entities and also mentions of covered entities for which their mapping based on link anchor texts, redirects and disambiguation pages fails to retrieve a candidate.

Unfortunately, most of the described approaches used different performance measures. This is presumably motivated by the intrinsic design of the approaches but renders comparison difficult. We have discussed some drawbacks of the specific measures in this section but can not state which of them is the most suitable for the task at hand. Generally we would argue that the macro performance introduced in Section 3.5.4 is the most appropriate and expressive measure. But since we decided to compare against the figures as published by the authors, we have to use the respective performance measures for a fair comparison. We will discuss this decision further in our evaluation where we use Ratinov et al.'s BoT-evaluation as reference measure but also provide results in Hoffart et al.'s MAP (Eq. 4.4) and Kulkarni et al.'s variant of BoT (Eq. 4.3).

To summarize, joint entity linking aims at maximizing the coherence of candidates often in probabilistic graphical models. Such methods can be computationally expensive (Kulkarni et al. [2009], Han et al. [2011]) or may require the design of logical predicates (Fahrni and Strube [2012]). In contrast, most collective entity linking approaches do not explicitly model the joint distribution over all candidates. Such approaches link each mention individually, independent of other, potentially inter-

dependent linking decisions. The joint nature of such approaches is realised in the usage of collective coherence attributes, e.g. semantic relatedness, that are computed on the document level and used as indicative features for candidate retrieval and/or candidate consolidation models (Milne and Witten [2008b], Ratinov et al. [2011], Hoffart et al. [2011b]).

In contrast to probabilistic graph based approaches that can be computationally expensive (Kulkarni et al. [2009], Han et al. [2011]) or may require the design of logical predicates (Fahrni and Strube [2012]), we propose a data driven approach that encodes the huge amount of structured and unstructured information stored in Wikipedia in a carefully constructed *search index*.

Inverted indices have become a favoured means to create alias dictionaries for entity linking. Different surface forms collected from Wikipedia titles, redirects, disambiguation pages and link anchor texts can be stored and retrieved on a per mention basis (Ratinov et al. [2011], Hachey et al. [2013]). For instance, Ratinov et al. [2011] retrieve candidates by querying their anchor-title index individually per mention m and keeping the 20 entities that have been most frequently used as target for the anchor text m. Varma et al. [2009] match the mention context against indexed entity contexts and use as prediction the top scoring entity. Here, we will go a step further and use relational information encoded in the co-occurrences of links for candidate retrieval.

The usefulness of search indices was also observed by Song and Heflin [2011] in the context of entity resolution in structured data. However, in structured data, exploitable attributes are very different from the attributes in unstructured data, the focus of this thesis. Attributes in databases often carry an inherent distinctiveness due to the creation process of the database, textual information needs first be made understandable, i.e. processable and searchable. The DBpedia project is one approach of structuring Wikipedia in a consistent database and could have been used here. However, we decided against it, since representing Wikipedia in our own index gives us more control about the kind of information used and the manner it is stored and retrieved. These aspects will be the main subjects of the following sections and we will start with a brief description of the used framework.

4.3 Wikipedia in an Inverted Index

Inverted indices allow the fast retrieval of documents relevant to a search query by mapping terms to their occurrences in the collection of indexed documents. In the context of entity linking, this can be used to retrieve all relevant candidate entities for a given mention. This is achieved by first encoding each Wikipedia entity in a document stored in such an index and then executing a carefully constructed search query. Depending on the entity attributes we index, a search query will retrieve most relevant candidate entities, where relevance is measured in terms of contextual

overlap but also semantic relatedness.

More specifically, we create two indices, namely an *entity index* and a *link index*. The entity index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ represents Wikipedia[1] and stores information for each entity $e \in \boldsymbol{\mathcal{W}}$. It stores textual data, alias information, YAGO type information as well as useful information from outlinks. The link index $\boldsymbol{\mathcal{I}_{L}}$ represents Wikipedias's hyperlink graph and stores pairs of Wikipedia link anchor texts and link targets. The entity index is used in candidate retrieval and consolidation, the link index is used to compute important figures such as EMP (Eq. 2.7) on the fly. Technically, both indices can be created from any Wikipedia dump in any language providing link anchor texts along with the respective targets, article texts and redirects.

Before we describe the details of these indices and the queries we may formulate for these indices, we give a brief description of Lucene and its scoring function, the framework of our candidate retrieval method.

4.3.1 Lucene as Indexing Framework

We use Apache's search engine Lucene[2] as framework, a fast and memory efficient open source implementation that allows the creation of inverted indices and facilitates the search in large scale text collections in a structured way. Each document in a Lucene index can have a multitude of distinct *fields* that are used to store specific types of information or content. A *search* in an index is performed using a *query* consisting of one or more *query terms* that are matched against the fields of the indexed documents. The result of the search is a ranked list of indexed documents where the ranking basically states how well each document in this ranking fits the query.

Fields and Queries

Each field in a Lucene index is qualified by a name and the value it stores. To indicate the difference, we use specific fonts for `field names`. Typically, the value of a field is a string or a collection of strings but it can also be a number. For instance, a field may store the title of an entity as one keyword, another may store all words from an article text as a collection of strings. A field may also be added several times to a document with multiple different values, for instance to store all synonyms of an entity[3].

Fields are the targets for search queries and named fields allow the placement of dedicated query terms. In Lucene, a query term $q_f(x)$ is associated with a specific field f and some value x. In the following we use the simplified notation as follows when referring to query terms on specific fields.

[1] `http://www.en.wikipedia.org`, version from September 1st, 2011.
[2] `http://lucene.apache.org/`
[3] Internally, Lucene handles this as a concatenation of all values associated with that field.

Notation (Query terms)
For a query term $q_f(x)$ the argument x is the value that is to be matched against the field f.

For example, the query term $q_{\texttt{title}}(\texttt{"Apple"})$ is associated with a field `title` and matches any field `title` that contains the value "Apple". A search query q is then formulated through a conjunction of one or more query terms, i.e.

$$q = q_{f_1}(x_1) \wedge \ldots \wedge q_{f_n}(x_l), \tag{4.5}$$

where each query term $q_{f_i}(x_j)$ is associated with a field $f_i, i = 1, \ldots, n$ and some value $x_j, j = 1, \ldots, l$ that may be the same or different for each query term.

Each query term can be characterized as either optional, mandatory or excluding. Mandatory terms must appear in an indexed document in order for the document to be retrieved, excluding terms effectively rule out all documents containing this term. Optional terms should appear in the document and if so will increase the query related score of a document, but in a conjunction of optional terms not all terms need to be present. Each optional term can be endowed with a weight to emphasise its importance. The same holds for a sequence of terms and the combination of different queries.

Example 11 (Queries and Query Terms)
Assume an index with fields `A`, `B`, `C`, and `D` each storing some string value. An exemplary query q to search this index is

$$q = +q_{\texttt{A}}(\,''a''\,) \wedge\, !q_{\texttt{B}}(\,''b''\,) \;\wedge\; q_{\texttt{C}}(\,''x''\,) \wedge\; w \cdot q_{\texttt{D}}(\,''y''\,).$$

The query q is a conjunction of the following query terms:

- a mandatory term $q_{\texttt{A}}(\,''a''\,)$ (indicated by $+$)
- an excluding term $q_{\texttt{B}}(\,''b''\,)$ (indicated by !)
- two optional terms $q_{\texttt{C}}(\,''x''\,)$ and $q_{\texttt{D}}(\,''y''\,)$, the latter weighted by a factor w.

The query q will retrieve only those documents that contain the value $''a''$ in a field `A`, but do not contain the value $''b''$ in any field `B`. The documents fulfilling these constraints are ranked according to the number of matches on the fields `A`, `C` and `D`, matches on the latter are weighted by some factor w. If for instance the term $q_{\texttt{D}}(\,''y''\,)$ is three times more important than $q_{\texttt{C}}(\,''x''\,)$, we would use $w = 3$.

The concept of fields allows us to encode distinct entity attributes in specific fields that may have different importance for entity linking. As already stated, the importance of an attribute can be emphasised or *boosted* through the usage

of weights. For instance, we may formulate a query that has a higher weight for exact matches between mention name and entity name and a lower weight for partial matches. This weighting will then be used in Lucene's scoring function. This scoring function associates each document retrieved from a search with an individual ranking score.

Scoring in Lucene

According to Hatcher et al. [2010], the score $s_{\boldsymbol{\mathcal{I}_\mathcal{W}}}(q, d)$ of a document d for a search with a query q in an index $\boldsymbol{\mathcal{I}_\mathcal{W}}$ is given by:

$$s_{\boldsymbol{\mathcal{I}_\mathcal{W}}}(q, d) = norm(q) \cdot c(q, d) \sum_{q_f(x) \in q} tf_{q_f(x),d} \cdot idf^2_{q_f(x)} \cdot w_{q_f(x),d} \cdot norm(q_f(x), d). \quad (4.6)$$

The quantity $tf_{q_f(x),d} \in \mathbb{N}^{\geq 0}$ denotes the frequency of term $q_f(x)$ in d which is the number of times the value x appears in any field f in document d. The factor $idf_{q_f(x)} \in \mathbb{R}^{\geq 0}$ is the inverse document frequency as in Eq. 3.2 and reflects how may documents contain the value x in any field f. The factor $w_{q_f(x)} \in \mathbb{R}^{>0}$ is the weight on a specific query term $q_f(x)$. Later, we will use this factor to emphasize matches on dedicated fields.

The normalization factor $norm(q)$ in Eq. 4.6 is the same for all documents and used internally by Lucene to compare different queries. It is given by the sum of squared weights of each of its terms

$$norm(q) = \frac{1}{\sqrt{w_q^2 \cdot \sum_{q_f(x) \in q} (idf_{q_f(x)} \cdot w_{q_f(x)})^2}}, \quad (4.7)$$

where the additional factor $w_q \in \mathbb{R}^{>0}$ denotes the weight on conjunctions of terms in the query q.

The coordination factor $c(q, d)$ in Eq. 4.6 is based on the number of terms $q_f(x)$ a document contains. It rewards a document containing many query terms by increasing the document's score over those of documents containing less terms. The last factor $norm(q_f(x), d)$ in Eq. 4.6 is a field-length norm and computed over the values in a field f. Comparable to a length norm, it is used to give a higher score to fields with few values matching the query compared to fields with many values matching a query. For instance in the context of entity linking, this field-length norm makes sure that entities with short article text are treated similarly to entities with longer article text.

Having described the general aspects for index or search based entity retrieval, we will now describe the underlying indices. We start with the link index $\boldsymbol{\mathcal{I}_L}$ since this index is also created first.

4.3.2 Link Index

In Ratinov et al. [2011], EMP was experimentally found to be a very competitive baseline for disambiguation. We follow this approach and employ this figure as a dedicated feature for entity linking. To efficiently estimate this value for any mention-candidate pair, we create the link index $\boldsymbol{\mathcal{I}_L}$ over all link anchor texts and link targets similar to Ratinov et al. [2011]. During the creation of this index we also collect valuable alias information from link anchor texts. This information is subsequently stored in designated fields in the entity index $\boldsymbol{\mathcal{I}_W}$. Therefore we create the link index first.

More specifically, we iterate over all articles in Wikipedia and collect the link attributes link target $l_t \in \boldsymbol{\mathcal{W}}$ and link anchor text l_a. We neglect the source $l_s \in \boldsymbol{\mathcal{W}}$ and store each pair (l_t, l_a) as a distinct document in $\boldsymbol{\mathcal{I}_L}$ to record its frequency of occurrence. Each indexed link is represented by the two fields `linkText` and `linkTo`. The field `linkText` stores the link anchor text l_a used to reference the link target l_t, the field `linkTo` stores the title $title(l_t)$ of the link target l_t.

> **Example 12**
> A link $l = (\cdot, l_t = \text{JOHN TAYLOR (BASS GUITARIST)}, l_a = \text{John Taylor})$ is represented by the index document with fields (`linkTo`, JOHN TAYLOR (BASS GUITARIST)) and (`linkText`, John Taylor).

Then, to estimate EMP, we only need to query the link index. For a specific mention m and a candidate entity e we create a query with two mandatory terms (`linkTo`, e) and (`linkText`, $name(m)$) that matches only indexed links containing both values. Dividing the number of returned indexed links by the overall frequency of the field (`linkText`,$name(m)$) in the link index then yields the EMP $p(e|m)$ according to Eq. 2.7. This trick allows the fast computation of EMP on demand and has low memory costs since we do not need to keep a probability table in memory.

Via the unique link target values $title(l_t), l_t \in \boldsymbol{\mathcal{W}}$, the link index $\boldsymbol{\mathcal{I}_L}$ is aligned with the entity index $\boldsymbol{\mathcal{I}_W}$ that we will describe next.

4.3.3 Entity Index

The entity index $\boldsymbol{\mathcal{I}_W}$ contains all Wikipedia entities and each indexed entity corresponds to a specific article in Wikipedia. Since we consider the purpose of entity linking as the assignment of a mention to a unique entity in Wikipedia, we specifically excluded disambiguation pages. In contrast to Kulkarni et al. [2009], we argue that a link to a disambiguation page does not solve the task of name disambiguation since these pages are merely listings of potential candidates. Therefore we do not

consider them as valid target entities and consequently do not index them. Furthermore, we also exclude designated meta pages such as category pages or other administrative entries.

Apart from intentionally deprecated articles, there may also be articles missing unintentionally. We used our own parser implementation that, due to the high variance in Wikipedia's markup language, could unfortunately not extract all articles correctly from the Wikipedia dump. These articles, i.e. the associated entities are not contained in the entity index $\mathcal{I}_{\mathcal{W}}$. For a precise treatment, we introduce the notation of *missing entities*.

Notation (Missing entities)
Entities that are originally covered in Wikipedia but erroneously missing in $\mathcal{I}_{\mathcal{W}}$ are, analogously to uncovered entities, denoted with NIL*.

Thus, we distinguish between covered entities contained in the index, uncovered entities NIL originally not covered in Wikipedia and missing entities NIL*. This distinction is necessary for evaluation: we do not assume our index to be a perfect representation of Wikipedia and thus want to account for potential errors resulting from missing entities.

Now, each indexed document in the entity index $\mathcal{I}_{\mathcal{W}}$ holds both unstructured textual information, as well as semi-structured attributes such as type information, popularity priors, outlinks and aliases. Each of these attributes is stored in dedicated fields that we will detail next. These fields allow the dedicated placement of queries against specific attributes that can furthermore be emphasised in importance using the weights introduced in Section 4.3.1.

Text Fields

The first field we describe stores the information from Wikipedia article texts $text(e)$ and is accordingly named `text`. We remove all Wikipedia markup language as well as all stop words and store the remaining article text in tokenized and stemmed form using Lucene's internal stemmer for English. We also make use of Lucene's term vector representations. Storing each article text as a vector of words, where each entry corresponds to the word's frequency in the article text, allows us to later efficiently compute important words, i.e. keywords, on a TF-IDF basis. This enables the usage of contextual similarity between mention context $text(m)$ and entity context $text(e)$ stored in this field. Then, we may either match the full context of a mention against all indexed entities, or formulate more specific queries of the form (`text`, *"word"*), where *"word"* may be the mention's surface form or any other important term in the mention context.

Notation (Contextual queries)
Queries against text fields are denoted by $q_{\text{text}}(x)$.

Ratinov et al. [2011] proposed to extend the article text with context terms extracted from referencing entities. To compute the textual similarity weights for their candidate selection, the authors represent the context of a candidate in two TF-IDF ranked word vectors, one obtained from the article text, the other from internal references in Wikipedia. The authors also evaluate different weighting schemes and find that weighting terms with respect to candidate contexts yields slightly better results compared to standard TF-IDF weighting with Wikipedia as background corpus for IDF. Similar weighting schemes have been shown to be superior to standard TF-IDF representation also by Mendes et al. [2011] in the context of entity linking and by Joachims [1997] in the context of text classification.

While reporting the results of different context weighting schemes, Ratinov et al. unfortunately did not report the effect of the context extension alone. We investigated this technique in initial experiments but couldn't find it useful. Instead, following the results obtained in the previous chapter, we will use the article text as is for the inference of topic distributions and employ the derived information as attribute in candidate consolidation.

Type Fields

Next to textual information, we also store type information, e.g. if the entity is a person or a location. To do so, we try to automatically align all entities in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ with YAGO using article URLs. The purpose of this alignment is to obtain entity types from YAGO that can be used as type attributes for named entities. Technically, YAGO covers more than 50 relations such as `happenedIn`(*"x"*) or `isCitizenOf`(*"y"*) that are mostly extracted from Wikipedia infoboxes but also generated from the alignment with WordNet. Here, we use the "type" relation that has predicates extracted from Wikipedia categories and WordNet. From the more than 60k predicates, we use here only the WordNet types "person", "location", "organization", "association", "team" and "club". The first two correspond to the named entity types *person* and *location*, the last four are subsumed under the named entity type *organization*. So for all entities that can be aligned with YAGO, we add this entity type information provided by YAGO. If a mention is endowed with such a type by an NER model, we may use this additional information to place a more distinctive query, for example a query $q_{\texttt{type}}(\textit{"person"})$.

Notation (Type based queries)
Queries against type fields are denoted by $q_{\texttt{type}}(x)$.

Queries against type fields allow the usage of entity type information. Since we store context and type information in separate fields, these fields can be queried separately and we will show the influence of type information in our experiments on search coverage.

Note that we will never use type queries as mandatory terms. In contrast to Hoffart et al. [2011b], we refrain from relying too strongly on YAGO's type system, since we want to avoid error propagation from mistakes made by named entity recognition models. Again, it is worth noting that all techniques proposed in this chapter are language independent and might also be applied for other languages. Relying strongly on the prediction of NER models would exclude languages where no such model is available.

Link Fields

Each indexed entity also holds information from Wikipedia's hyperlink graph. We store all outlinks $\mathbf{L}_{out}(e) = \{l \in \mathbf{L} | (l_s = e, l_t, l_a)\}$ of an entity e in designated link fields of the respective index document in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$. For each $l \in \mathbf{L}_{out}(e)$, we create two fields to store both the link target $title(l_t)$ as well as the link anchor text l_a. As in the link index $\boldsymbol{\mathcal{I}_{L}}$, a link $l = (l_s = e, l_t = e', l_a = \text{''}m\text{''})$ is stored in fields (`linkText`, $\text{''}m\text{''}$) and (`linkTo`, $title(e')$) where `linkText` holds the link anchor text $\text{''}m\text{''}$ and `linkTo` the title $title(e')$ of the link target entity e'. We denote queries against link fields in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ as follows.

Notation (Link queries)
Queries against link anchor text fields `linkText` are denoted by $q_{l_a}(x)$, queries against link targets fields `linkTo` are denoted by $q_{l_t}(x)$.

In the entity index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$, these link fields are associated directly with the source entity l_s from which they originate. This design provides the basis for our collective search that will be described in Section 4.6.1.

Prior Fields

Furthermore, we store popularity priors from inlinks in a prior field. The prior field is the only numeric field in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ and holds the total number of inlinks of the respective entity. Mandatory queries against this field serve as threshold function on the minimum or maximum numbers of inlinks. For example, a mandatory query term $q_{p_{in}>5}$ excludes all entities from retrieval that have less than 5 inlinks. We use these priors for candidate retrieval where we initially require each candidate to have at least 5 inlinks in order to filter out rarely referenced entities. Importantly, this threshold is adaptive and our implementation is designed in such a way that it can automatically be lowered or even omitted.

Alias Fields

Next to the decision on the true underlying entity, the retrieval of candidates is one of the most important aspects of successful entity linking. Thus, aliases are one of the

most valuable resources in entity linking. They are crucial for candidate retrieval that needs to return all relevant candidates for a given mention. Approaches to word sense disambiguation may benefit from WordNet's synsets that contain the common synonyms of words (Miller [1995]). For entity linking, we first need to create an analogous resource. Especially for named entities this is complicated due to the common usage of nicknames, abbreviations, translations, spelling variations etc. Here, we extract and generate aliases from Wikipedia.

To enable high candidate recall, we require a comprehensive alias resource that should provide all the possible names for an entity, e.g. its synonyms such as nicknames or acronyms. At the same time it should reflect that a mention may be polysemous. This knowledge is encoded in Wikipedia's hyperlink graph that provides the mapping between mentions and different target entities.

We store all the aliases we can retrieve or generate in the entity index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ along with the entity they belong to. Then, alias fields subsume all known as well as possible names of an entity. To enable the emphasis of matches on specific fields through weighted query terms, we create for each alias type, e.g. redirect or abbreviation, distinct alias fields.

This naturally also includes the title of an entity. Noting that the title of an entity is usually its most commonly used name, we store for each entity a unique field `title` that holds the title $title(e)$ of the associated Wikipedia article. Then, for instance to account for the canonical usage of entity names in news paper articles, we may match a mention directly against all Wikipedia titles.

Notation (Title queries)
Queries against title fields are denoted by $q_{\text{title}}(x)$.

Additionally, we also store the title without disambiguation term, i.e. the name of the entity $name(e)$, in the field `name`. To account for synonymy and polysemy, we extract all redirects from the Wikipedia redirect dump and then add all redirects of an entity as distinct `redirect` fields to the indexed document in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$. Even though the usage of redirects may lead to errors (we gave examples in Section 2.3.3), we consider all redirects assuming that erroneous redirects are the minority.

Now, since these resources need not reflect all possible name variations, we also artificially generate new variations for entity names. This is supposed to increase candidate recall especially for mentions that were not used in Wikipedia. To do so, we use a simple heuristic to create abbreviations and acronyms for names consisting of more than one word, i.e. phrases. More specifically, we split a phrase $name(e)$ into distinct tokens and use each possible combination of initial letter and token as an abbreviation. For example, for the phrase Michael Jordan we obtain the abbreviations *M. Jordan*, *Michael J.*, *M. J.* as well as the acronym *MJ*. We assume acronyms to be especially useful for entities with long names that tend to be referenced to by their acronyms, e.g. *BSE* or *DNA*. The generated abbreviations and acronyms are then stored in dedicated `abbreviation` fields.

Finally, we use link anchor texts as alias resource. We collect all link anchor texts during the creation of the link index $\boldsymbol{\mathcal{I}_L}$ and then create fields `meantBy` storing the link anchor texts of all $\mathbf{L}_{in}(e)$. Similar to redirects, these fields provide alternative names and entity aliases that may not be found in the article text itself.

Importantly, we obtain the EMP for each pair of link target and anchor text from the creation of $\boldsymbol{\mathcal{I}_L}$. These probability values are used in $\boldsymbol{\mathcal{I}_\mathcal{W}}$ as weighting factors on the associated fields `meantBy`. For example, for two entities e and e' and some link anchor text m, we may find $p(e|m) = 0.8$ and $p(e'|m) = 0.2$. We then give the field (`meantBy`, m) for the entity e a higher weight by using a boost factor of $p(e|m) = 0.8$. To reflect that e' has a lower probability to be referenced by m, we give the field (`meantBy`, m) a lower weight using a boost of $p(e'|m) = 0.2$. As described in Section 4.3.1 these weights are used internally in Lucene's scoring function, and basically weigh a match on the field (`meantBy`, m) for the indexed entity e with the factor 0.8 and for the indexed entity e' with the factor 0.2. This procedure enables allows us to implicitly exploit EMP during query time without the need for re-computation.

For simplicity, all of the above introduced fields are henceforth subsumed as *alias* fields. As recap, the following example illustrates the alias fields we generate for Michael Jordan.

Example 13 (Alias fields for Michael Jordan)
The entity in $\boldsymbol{\mathcal{I}_\mathcal{W}}$ representing the basketball player Michael Jordan has the following alias fields, where each field is a tupel of field name and stored value, i.e. (`fieldName`,"content").

name: (`name`, "Michael Jordan")

abbreviations: (`abbreviation`, "M. Jordan"), (`abbreviation`, "Michael J."), (`abbreviation`, "MJ") ...

redirects: (`redirect`, "His Airness"), (`redirect`, "Michael Jeffrey Jordan"), (`redirect`, "Michael Jeffery Jordan"), ...

link anchor texts: (`meantBy`, "jordanesque"), (`meantBy`, "american basketballer of the same name"), ...

We store all of the described alias fields in two forms. The tokenized and stemmed form allows for fuzzy, indirect matches which is intended to increase candidate recall. This includes for example the capability to handle the insertion of middle names in mentions or the matching of verbs against their respective noun forms, e.g. writing and Writer. The other form stores the field values in their original form, i.e. as a single string. This allows for direct matches based on string equality. Exact matches often induce the underlying entity of a mention and may therefore be prioritized.

Table 4.1: Fields in the entity index $\mathcal{I}_{\mathcal{W}}$. Alias fields are marked with an asterisk. Apart from `text`, all fields are stored in tokenized and stemmed form to allow fuzzy matches as well as in their original form that is not processed and allows exact matches. The field `text` is stored only in the former variant.

field name	content
`title`	the unique title $title(e)$
`text`	the article text $text(e)$
`type`	the named entity type (derived from YAGO)
`linkText`	the link anchor texts of all outlinks $l \in \mathbf{L}_{out}(e)$
`linkTo`	the titles of all link targets of the outlinks $l \in \mathbf{L}_{out}(e)$
*`name`	the $name(e)$, i.e. the title without disambiguation term
*`meantBy`	the link anchor texts of all inlinks $l \in \mathbf{L}_{in}(e)$
*`redirect`	the redirects $\boldsymbol{r}(e)$
*`abbreviation`	the abbreviations and acronyms generated from $name(e)$

Now, alias fields allow queries of the form $q_{\texttt{name}}(m)$, $q_{\texttt{redirect}}(m)$ or $q_{\texttt{meantBy}}(m)$. We may use them for direct matches of unambiguous mention names, but also to retrieve candidate entities from $\mathcal{I}_{\mathcal{W}}$ that are referenced trough abbreviations or synonyms.

Notation (Alias queries)
Queries against all alias fields are collectively denoted by $q_{\text{alias}}(x)$.

To demonstrate the value of alias resources, we will experimentally evaluate entity linking using only alias fields without contextual or other information.

For a better overview, all of the fields introduced in this section are summarized in Tab. 4.1. Having defined the basis of our entity linking method with the entity index $\mathcal{I}_{\mathcal{W}}$ and the link index $\mathcal{I}_{\boldsymbol{L}}$, we will now give a brief overview of the proposed model. This method involves a two stage candidate retrieval process and a final step for candidate consolidation.

4.4 Overview: Entity Linking via Search and Ranking

We propose a multi-stage entity linking model. This linking model uses the index $\mathcal{I}_{\mathcal{W}}$ to generate candidate entities, as well as a supervised Ranking SVM to adjust the ranking of these candidates and to detect mentions of uncovered entities. For a better overview, we give here a short description of the involved steps that we will describe in more detail in the following sections.

Mention Enrichment First, we extract all mentions $\mathbf{M} = \{m_1, \ldots, m_k\}$ from the input document, perform in-document co-reference resolution for name expansion and enrich each mention with attributes derived from context and, if available, entity type information. More details on these steps will be given in Section 4.5.

Candidate Retrieval Second, we perform a two-stage candidate retrieval using global and local information. In the first stage, we collectively use all mentions $\mathbf{M} = \{m_1, \ldots, m_k\}$ from the input document and exploit the co-occurrence of links in Wikipedia to arrive at candidate sets with strong inter-relatedness. Using an **ensemble query** that jointly treats all mentions $\mathbf{M}$ (Eq. 4.8), we perform a **collective search** in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ that retrieves source entities referencing many of the mentions $\mathbf{M}$. From the outlinks of these source entities, we retrieve at most k intermediate candidate sets $\mathbf{e}_1^c(m_1), \ldots, \mathbf{e}_k^c(m_k)$ where each set $\mathbf{e}_i^c(m_i) = \{e(m_i)\} \subset \boldsymbol{\mathcal{I}_{\mathcal{W}}}$ holds potential candidates for the mention m_i. On these candidate sets $\mathbf{e}_1^c(m_1), \ldots, \mathbf{e}_k^c(m_k)$, we compute the **cross coherence** $coh_{\times}$ (Eq. 4.10) of individual candidates to arrive at a coherent set of semantically related entities. The cross coherence accounts for the collective fitness of candidates and weighs candidates according to their coherence. Based on this coherence, we determine the best fitting global candidate $e^{coh}(m_i) \in \boldsymbol{\mathcal{W}}$ for each mention $m_i \in \mathbf{M}$ (Eq. 4.16).

In the second stage, we perform another search in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ for candidate retrieval but now we use a prioritization on the global candidates $e^{coh}(m_i)$. This is additionally combined with local, contextual information for each individual mention m_i. From this prioritized search, we obtain improved, ranked candidate sets $\mathbf{e}_1^*(m_1), \ldots, \mathbf{e}_k^*(m_k) \subset \boldsymbol{\mathcal{I}_{\mathcal{W}}}$. These two stages of candidate retrieval are described in detail in Section 4.6.

Candidate Consolidation Lastly, the retrieved candidates $\mathbf{e}_i^*(m_i)$ are consolidated through a supervised Ranking SVM. We apply the Ranking SVM to all candidates $\mathbf{e}_1^*(m_1), \ldots, \mathbf{e}_k^*(m_k)$ for **re-ranking** and **detection of uncovered entities**. From this we obtain the final predicted entity $\hat{e}(m_i) \in \mathbf{e}_i^*(m_i) \cup \{\text{NIL}\}$ for each mention m_i. Candidate consolidation is described in detail in Section 4.7.

Following this outline, we start with the first step, i.e. mention enrichment. To simplify re-implementation of the proposed algorithms, we give them in Section 4.6.2 and in Appendix A.

4.5 Mention Enrichment

We assume the input to our linking model to be a natural language text document with a collection of mentions $\mathbf{M} = \{m_1, \ldots, m_k\}$ to link. These mentions can either be given, as is the case for the benchmark corpora described in Section 4.2, or they can be provided by a chunker or a NER model. Note that in contrast to other

approaches such as Hoffart et al. [2011b], we do not restrict to mentions of named entities but also treat conceptual entities such as BANK or TREE. We may evaluate the available type information in candidate retrieval, but we do not thoroughly rely on it. Instead, we will mainly use it in our heuristic name expansion algorithm.

4.5.1 Name Expansion

Entities and typically persons are usually mentioned only once with their full name in a document. Subsequent mentions are often abbreviations and use for instance only the last name of a person. This can be misleading for candidate retrieval as it can notably increase the number of candidates and may also distract from the correct candidate when a different entity has a notably higher entity-mention probability EMP. For instance, this would be the case for a mention Wilhelm Busch that is later in the document abbreviated to Busch.

To account for this, we propose *name expansion* with a simple, heuristic in-document co-reference resolution. For this, we first apply the publicly available Apache OpenNLP NER model[1] to infer entity types for the mentions in a document. Then we collect all mentions from the document and use each mention's surface form along with its type information (if present) for name expansion. More specifically, we iterate over the mention sequence $\mathbf{M} = \{m_1, \ldots, m_k\}$ and search for mentions that are partially or token-wise contained in a preceding mention, i.e. $name(m_i) \subset name(m_{i-j})$ for any $i = 1, \ldots, k-1$ and $0 < j < i$. If such a match is found and the type of the corresponding two mentions is the same, e.g. $type(m_i) = type(m_{i-j}) =$ person, the shorter mention is expanded to the longer one.

Example 14 (Name expansion)
For the mention collection

$$\mathbf{M} = \{(\text{Al Gore}, person), (\text{Gore}, person), (\text{Gore Bay}, location)\}$$

name expansion yields

$$\mathbf{M}_{exp} = \{(\text{Al Gore}, person), (\text{Al Gore}, person), (\text{Gore Bay}, location)\}.$$

This name expansion based on co-reference resolution is similar to Shen et al. [2012] but additionally incorporates the type information. Cucerzan [2007] used a similar method but required mentions to have the same type for expansion. We relax this assumption when encountering mentions of unknown entity type. Assume that the type of Gore in Example 14 would be unknown because the NER model failed to recognize it as a person mention. Then, we still assume that it resolves to the person

[1] `http://opennlp.apache.org/`

mention Al Gore, since the abbreviation of person names is supposedly much more common than the abbreviation of location names. Thus, since this name expansion does not fully depend on type information, mentions without type assignment are also handled.

Given that our matching is token-based and not character-based, we do not expand acronyms as done by Cucerzan [2007] or Varma et al. [2009]. This would certainly be a point worth investigating for future work, particularly because Hachey et al. [2013] report that the performance of Cucerzan [2007] drops by about 5% when this expansion is omitted. The authors report similar decrease in performance when the acronym expander of Varma et al. [2009] is omitted. Here, we also experimentally evaluated the effect of name expansion and found that is has a positive impact on linking performance. We will detail our findings further in Section 4.8.2.

4.5.2 Context Representation

We use different context representations for candidate retrieval and candidate consolidation. For candidate retrieval, we emphasize on the local, mention specific context. For candidate consolidation we will use document-level information and also latent topics, the details will be described in Section 4.7. Here we describe the mention context representation used in candidate retrieval.

Since the disambiguating quality of the mention context is important for entity linking, we extract context not only on the document level but also from local, mention specific context features. To do so, we first extract all PoS tags in the document using the Apache OpenNLP PoS tagging tool. Then, local context words are the two nouns left and right of the mention. This is motivated by the idea that noun phrases, named entities or conceptual entities that imply the sense of an ambiguous mention usually appear close to this mention.

Additionally, we extract the top 20 TF-IDF ranked keywords from the document text as document-level keywords. For this, we use Wikipedia as background corpus for IDF computation. Since short documents may contain a lower number of important words, we refrain from using a threshold and simply use the 20 words with highest TF-IDF score. The document-level keyword set is then localized for each mention. From the joint set of local context words and document keywords, we keep only those terms that relate to any of the mention's candidates. More specifically, the candidate specific terms are those words that appear at least once in the text of any candidate entity whose title matches the surface form of the mention. This candidate dependent keyword selection aims at using specifically those terms that are discriminative for entities. In the same way, we compute keywords from the headline of the input document, assuming that headline information is especially important. Since we refrain from tuning extraction methods to specific corpora, we use a simple headline extraction method that assumes the headline to be the first line in the document followed by a line break.

Arguably, the extraction of these context features is presumably more complex than the context representation of other approaches, for instance that of Ratinov et al. [2011] who use a window of 100 TF-IDF weighted words as mention context. But as also shown by Mendes et al. [2011], entity specific context weighting can be superior to standard TF-IDF based context representations. Mendes et al. [2011] propose a candidate-set specific term weighting that captures the importance of a word for a specific candidate set. Evaluating the influence of contextual similarity, the authors found that linking a mention to the entity with highest contextual overlap results in a notably higher accuracy of 73.39% when context terms are weighted with the proposed candidate-set specific term weighting in contrast to standard TF-IDF weighting that was found to achieve an accuracy of only 55.91%.

Now, as name, type and context attributes of the mentions can be matched to the associated fields in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using specific queries, we can evaluate searches of different coverage, i.e. search based entity linking using cumulatively more of the above attributes. We will present the influence of these attributes also in combination with the prioritization on the global candidates retrieved from collective search in our experimental evaluation (Section 4.8.2).

We will now come to the second step in our entity linking model and describe the two-stage candidate retrieval approach.

4.6 Candidate Retrieval

In editorial texts such as newspaper articles, persons are often referenced by their canonical name. In such cases the simple string matching techniques we used for person name linking in Chapter 3 are often sufficient for candidate retrieval. In this chapter we want to generalize to other entity types and to do so we propose a more elaborate candidate retrieval method. We will use all of the described alias sources stored in our entity index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ but extend them with a relational candidate retrieval method based on coherence. In the literature, coherence is often motivated from co-occurrence: the joint mention of "Queen" and "Mercury" indicates that the document refers to the rock band and its singer, rather than the British queen and the planet. This co-occurrence information is explicitly reflected in Wikipedia's hyperlink graph and consequently measures such as SRL (Eq. 2.2) are the typical means for the definition of a coherence measure. Here, we also exploit co-occurrence but are the first to match all mentions in a document *collectively* against the hyperlink graph to arrive at coherent candidate sets. To do so, we rely on the link information stored in the entity index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$.

4.6.1 Collective Search

Collective search is motivated by the assumption that Wikipedia entities referencing many of the given mentions are likely to have a similar subject as the input document. Then, from the *outlink target entities* these entities provide, we can automatically generate intermediate but reliable candidate entity sets that will in many cases contain the correct underlying entities for the given mentions. To find the best source entities of these outlink targets, we create an *ensemble query* over all the given mentions. This ensemble query is then matched against the link information encoded in our index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ and thus also implicitly against the full hyperlink graph of Wikipedia.

Ensemble Query Generation

To exploit the co-occurrence of mentions as link anchor texts in Wikipedia, we create an ensemble query $q_{\mathbf{M}}$ that jointly treats the names $name(m_i)$ of all mentions $m_i \in \mathbf{M} = \{m_1, \ldots, m_k\}$. This query then contains one link anchor text query term $q_{l_a}(m_i)$ per mention $m_i \in \mathbf{M}$ and, according to Eq. 4.5, is formed as a conjunction over these terms:

$$q_{\mathbf{M}} = q_{l_a}(name(m_1)) \wedge \ldots \wedge q_{l_a}(name(m_k)) \tag{4.8}$$

Importantly, we use no mandatory terms to state that a specific mention must appear. First, this would require prior knowledge on the importance of mentions. Second, if a mandatory mention was never observed as a link anchor text in Wikipedia, a search in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using this query would always return zero results. Also, we do not use weights on specific terms and thus treat all mentions equally. For future work, it would be worth investigating the existence of seed entities, i.e. entities that should be weighted higher in such a query because they are more influential for the document-level entity distribution.

Note that such an ensemble query can also be used to approximate the probability of joint occurrence of the mentions $\mathbf{M}$: few hits indicate a low joint probability, many hits indicate a high joint probability.

Candidates retrieved from Ensemble Queries

Now, to retrieve the aforementioned source entities, we search $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using the ensemble query $q_{\mathbf{M}}$ and obtain a ranked list of source entities $\mathbf{S}_{q_{\mathbf{M}}} \in \boldsymbol{\mathcal{W}}$ that *collectively* contain a high number of the input mentions m_i as values in their link text fields. To avoid noise, we restrict the number of returned hits and use at most 30 source entities. As described in Section 4.3.1, Lucene ranks each source entity $e_{q_{\mathbf{M}}} \in \mathbf{S}_{q_{\mathbf{M}}}$ with a score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}$ that is based on the number of matches of the mention m_i on the link text fields (`linkText`, m_i) of $e_{q_{\mathbf{M}}}$. According to Eq. 4.6, this score relates to

document text

[...] Shepard, Glenn set first two milestones
May 5, 1961: Alan Shepard becomes first American in space.
Feb. 20, 1962: John Glenn becomes first American in orbit.
Jan. 27, 1967: Gus Grissom, Edward White II and Roger Chaffee die in Apollo 1 spacecraft fire on launch pad.
July 20, 1969: Apollo 11's Neil Armstrong and Buzz Aldrin land on moon.
July 17, 1975: American Apollo and Soviet Soyuz spacecraft link in orbit.
April 12, 1981: Columbia soars on first space shuttle flight.
June 18, 1983: Sally Ride becomes first American woman in space.
Jan. 28, 1986: Challenger explodes, killing all seven on board.
April 25, 1990: Hubble Space Telescope is released into orbit.
Dec. 2, 1993: First Hubble repair mission is launched.
March 14, 1995: Norman Thagard is first American to be launched on a Russian rocket. Two days later, he becomes first American to visit Mir.
June 29, 1995: Atlantis docks with Mir in first shuttle-station hookup.
Sept. 26, 1996: Shannon Lucid returns to Earth after 188-day Mir mission, a U.S. space endurance record and a world record for women.
Nov. 19, 1996: Story Musgrave, at age 61, becomes oldest man in space.
Oct. 29, 1998: Discovery is scheduled to blast off, carrying 77-year-old John Glenn back into orbit and making him oldest man in space. [...]

Figure 4.1: Excerpt from a document in the **AQUAINT** corpus. The mentions to be linked to Wikipedia are highlighted, here this is only the first mention of an entity in the document.

the TF-IDF of a mention m_i for a source entity $e_{q_\mathbf{M}}$ where the operating figures are computed over the index fields l_a. Consequently, the more mentions an entity $e_{q_\mathbf{M}}$ contains as link anchor text l_a, the higher the ranking score $s_{\mathcal{I}_\mathcal{W}}(q_\mathbf{M}, e_{q_\mathbf{M}})$.

Now, since each $e_{q_\mathbf{M}}$ also provides the link targets $l_t \in \mathcal{W}$ for the link anchor texts l_a in the fields `linkTo`, we can extract all outlink targets $l_t \in \mathcal{W}$ from all source entities in $\mathbf{S}_{q_\mathbf{M}}$, i.e. all $e = l_t \in \mathbf{L}_{out}(\mathbf{S}_{q_\mathbf{M}})$. We endow each of these outlink targets $e = l_t \in \mathbf{L}_{out}(\mathbf{S}_{q_\mathbf{M}})$ with a relevance weight $w_r(e)$. This weight is the sum over the scores $s_{\mathcal{I}_\mathcal{W}}(q_\mathbf{M}, e_{q_\mathbf{M}})$ for the different source entities $e_{q_\mathbf{M}} \in \mathbf{S}_{q_\mathbf{M}}$ that contain e as an outlink target $e \in \mathbf{L}_{out}(e_{q_\mathbf{M}})$:

$$w_r(e) = \sum_{e_{q_\mathbf{M}} \in \mathbf{S}_{q_\mathbf{M}}} \delta_e s_{\mathcal{I}_\mathcal{W}}(q_\mathbf{M}, e_{q_\mathbf{M}}), \quad \delta_e = \begin{cases} 1 & \text{iff } e \in \mathbf{L}_{out}(e_{q_\mathbf{M}}), \\ 0 & \text{else.} \end{cases} \tag{4.9}$$

These weights are interpreted as the relevance of a candidate and we use them to remove less relevant candidates from the overall set $\mathbf{L}_{out}(\mathbf{S}_{q_\mathbf{M}})$ that may easily contain more than a thousand of different entities appearing only once as outlink target. Therefore, we keep only a reduced set $\mathbf{L}^*_{out}(\mathbf{S}_{q_\mathbf{M}})$ of the top 100 candidate entities in $\mathbf{L}_{out}(\mathbf{S}_{q_\mathbf{M}})$ that have the highest relevance weights $w_r(e)$.

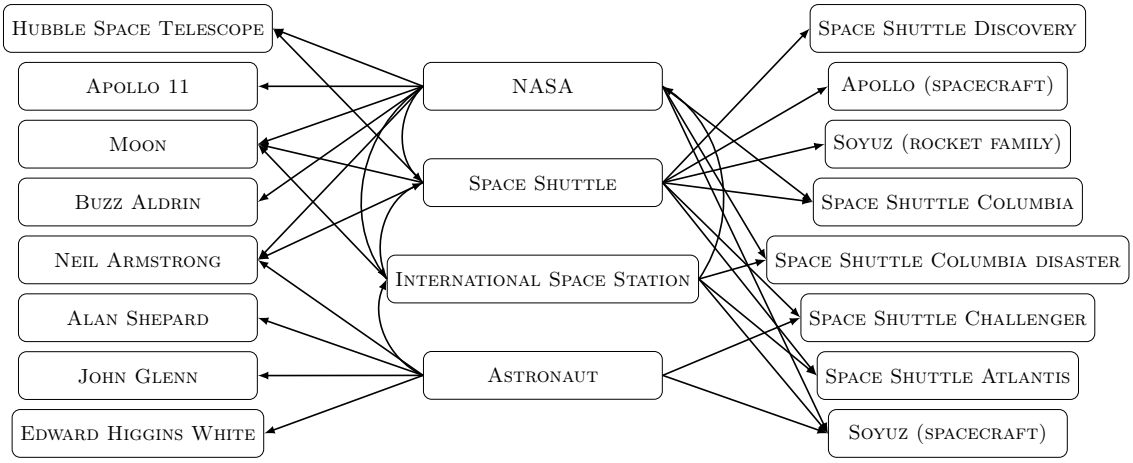


Figure 4.2: Illustration of collective search results (Example 15). The figure shows a reduced link network for the top ranked source entities $\mathbf{S}_{q_\mathbf{M}}$ (middle) retrieved with an ensemble query for the mentions depicted in Fig. 4.1. The entities NASA, Space Shuttle and International Space Station have highest rank as they contain many link anchor texts matching the given mentions, e.g. Challenger, Columbia and Atlantis. For simplicity we depict only outlinks and each link only once. Indeed, each link may appear multiple times in the article of a source entity, e.g. we find five outlinks from NASA to Space Shuttle.

To illustrate the described process of collective search, we give the following example using a document from the **AQUAINT** corpus as depicted in Fig. 4.1. For convenience, we reduce to a subset of the mentions contained in the document and show only a selection of source entities and outlink targets in Fig. 4.2.

Example 15 (Collective Search)
Take a list of mentions contained in the document as depicted in Fig. 4.1:

$$\mathbf{M} = \{\text{Apollo 11}, \text{spacecraft}, \text{Columbia}, \text{Challenger}, \text{Atlantis}, \text{Discovery}, \ldots\}.$$

Following Eq. 4.8, the ensemble query created for $\mathbf{M}$ is

$$q_\mathbf{M} = q_{l_a}(\text{"Apollo 11"}) \wedge q_{l_a}(\text{"spacecraft"}) \wedge q_{l_a}(\text{"Columbia"}) \\ \wedge q_{l_a}(\text{"Challenger"}) \wedge q_{l_a}(\text{"Atlantis"}) \wedge q_{l_a}(\text{"Discovery"}) \wedge \ldots.$$

A search in $\boldsymbol{\mathcal{I}_\mathcal{W}}$ using this query $q_\mathbf{M}$ returns 30 ranked sources entities $\mathbf{S}_{q_\mathbf{M}}$. For illustration, we show here only the highlights:

1. NASA, $s_{\boldsymbol{\mathcal{I}_\mathcal{W}}} = 143.76$
2. Space Shuttle, $s_{\boldsymbol{\mathcal{I}_\mathcal{W}}} = 120.98$
3. International Space Station, $s_{\boldsymbol{\mathcal{I}_\mathcal{W}}} = 119.56$

4. Apollo program, $s_{\mathcal{I}_\mathcal{W}} = 89.13$
⋮
8. Moon, $s_{\mathcal{I}_\mathcal{W}} = 56.51$
9. Astronaut, $s_{\mathcal{I}_\mathcal{W}} = 53.78$
⋮
20. Space Shuttle Atlantis, $s_{\mathcal{I}_\mathcal{W}} = 34.29$
21. Space Shuttle Columbia disaster, $s_{\mathcal{I}_\mathcal{W}} = 32.23$
⋮

Note that the scores $s_{\mathcal{I}_\mathcal{W}}(q_\mathrm{M}, e_{q_\mathrm{M}})$ are given here only for illustration and can not be interpreted without the context of this example. As we see from this ranked list, the retrieved source entities are thematically very related to the content of the document (cf. Fig. 4.1). This is also illustrated through the dense linkage in Fig. 4.2: the entities NASA, International Space Station and Space Shuttle are ranked highest as they contain most of the given mentions, e.g. Challenger, Columbia and Atlantis, as link anchor texts, albeit with different link targets. The entity Apollo Program is already more specific and, containing less mentions as link anchor text, receives a notably lower score $s_{\mathcal{I}_\mathcal{W}}$.

Since these entities contain the required link anchor texts, they consequently also contain many outlink targets that indeed correspond to the ground truth entities of the given mentions. As we see in Fig. 4.2, the set of outlink targets $\mathbf{L}_{out}(\mathbf{S}_{q_\mathrm{M}})$ of the source entities NASA, Space Shuttle etc. contains

$$\begin{aligned}\mathbf{L}_{out}(\mathbf{S}_{q_\mathrm{M}}) = \{&\text{Space Shuttle Columbia}, \ldots,\\ &\text{Space Shuttle Challenger}, \ldots,\\ &\text{Space Shuttle Atlantis}, \ldots,\\ &\text{Space Shuttle Discovery}, \ldots\}\end{aligned}$$

Following Eq. 4.9, these outlink targets are weighted with relevance weights $w_r(e)$ based on the score of their respective source entities in $\mathbf{S}_{q_\mathrm{M}}$. The higher this weight, the more often the entity e is an outlink target of any source entity $e_{q_\mathrm{M}} \in \mathbf{S}_{q_\mathrm{M}}$. The highest weight would be obtained for an entity that appears in all outlink target sets of the source entities containing at the same time many mentions as link anchor text.

At this point, the retrieved candidates form a set of potential targets that is not yet related to the input mentions. To link the elements in the target entity set $\mathbf{L}^*_{out}(\mathbf{S}_{q_\mathrm{M}})$ with the input mentions, we use their respective title and redirect index fields. More specifically, we analyse for each $e \in \mathbf{L}^*_{out}(\mathbf{S}_{q_\mathrm{M}})$ if either the title or the redirect of e contains any $name(m_i)$. If so, we add e to the candidate set $\mathbf{e}^c_i(m_i)$ for mention m_i. Note that one e can then be contained in multiple candidate sets. Alternatively, when this candidate-mention association yields no result, no collective

search candidate can be assigned.

The result of the collective search and the above candidate assignment is the collection $\mathbf{e}_1^c(m_1), \ldots, \mathbf{e}_k^c(m_k)$, where each set $\mathbf{e}_i^c(m_i)$ is a set of candidate entities for mention m_i.

Cross Coherence among Candidate Entities

Our intuition is that entities mentioned jointly in a document should be related. Following other approaches, we use the SRL* measure over inlinks (cf. Eq. 2.4) to compute the relatedness among Wikipedia entities. Now, SRL* is usually used to compute the pairwise relatedness of two entities. Here, we introduce *cross coherence* to account for the *collective* fitness of a set of entities.

Cross coherence states how well a specific candidate entity $e_{ij} \in \mathbf{e}_i^c(m_i)$ *fits* to the other candidate entities $\{\mathbf{e}_l^c\}_{l=1,l\neq i}^{|\mathbf{M}|}$. More formally, we define the cross coherence $coh_\times$ of a candidate $e_{ij} \in \mathbf{e}_i^c$ towards a collection of other candidates $\{\mathbf{e}_l^c\}_{l=1,l\neq i}^{|\mathbf{M}|}$ as:

$$coh_\times(e_{ij}, \{\mathbf{e}_l^c\}_{\substack{l=1\\l\neq i}}^{|\mathbf{M}|}) := \frac{1}{|\mathbf{M}|} \sum_{\substack{l=1\\l\neq i\\\mathbf{e}_i^c\neq\mathbf{e}_l^c}}^{|\mathbf{M}|} \frac{1}{|\mathbf{e}_l^c|} \sum_{\substack{e'\in\mathbf{e}_l^c\\e_{ij}\neq e'}} \Delta \cdot \text{SRL}^*(e_{ij}, e'). \tag{4.10}$$

Here, $|\mathbf{M}|$ is the total number of mentions in the document, i the index over these mentions and j the index over the candidates for a mention m_i. The second sum in Eq. 4.10 computes the averaged pairwise SRL* of candidate e_{ij} for mention m_i and the candidates in another candidate set $\mathbf{e}_l^c$ for another mention m_l. This is weighted by the factor Δ, a real-value scalar that we use to account for contextual similarity and describe in more detail in the following. The weighted averaged relatedness is then again averaged over all candidate sets for all mentions by the first sum in Eq. 4.10.

With the definition above, cross coherence can be interpreted as the average distance of an entity to a collection of entities and has range $[0, 1]$. This range is also preserved through the weighting factor Δ in Eq. 4.10. This factor serves as an additional weight of relatedness and may be the EMP of a candidate (Milne and Witten [2008b]) or a binary value indicating that the two candidates link to each other (Ratinov et al. [2011]). Since the first variant may erroneously prioritize high popularity candidates and the second is somewhat restrictive, we here propose factors that constitute contextual similarity weights.

Cross Coherence Weight Factors

To evaluate the effect of the different weighting factors, we will compare against a baseline that uses no weight factor for semantic relatedness by omitting the term Δ

in Eq. 4.10. For better distinction, we will denote this baseline with coh_{SRL^*} and then have

$$coh_{\text{SRL}^*}(e_{ij}, \{\mathbf{e}_l^c\}_{\substack{l=1 \\ l\neq i}}^{|\mathbf{M}|}) := \frac{1}{|\mathbf{M}|} \sum_{\substack{l=1 \\ l\neq i \\ \mathbf{e}_i^c \neq \mathbf{e}_l^c}}^{|\mathbf{M}|} \frac{1}{|\mathbf{e}_l^c|} \sum_{\substack{e' \in \mathbf{e}_l^c \\ e_{ij} \neq e'}} \text{SRL}^*(e_{ij}, e'). \tag{4.11}$$

The first weight that we evaluate is based on the cosine similarity of candidate contexts $\cos(text(e), text(e'))$ (cf. Eq. 3.1). Analogously to Eq. 4.11, we replace the factor Δ in Eq. 4.10 and arrive at

$$coh_{\cos \text{SRL}^*}(e_{ij}, \{\mathbf{e}_l^c\}_{\substack{l=1 \\ l\neq i}}^{|\mathbf{M}|}) := \frac{1}{|\mathbf{M}|} \sum_{\substack{l=1 \\ l\neq i \\ \mathbf{e}_i^c \neq \mathbf{e}_l^c}}^{|\mathbf{M}|} \frac{1}{|\mathbf{e}_l^c|} \sum_{\substack{e' \in \mathbf{e}_l^c \\ e_{ij} \neq e'}} \cos(text(e_{ij}), text(e')) \cdot \text{SRL}^*(e_{ij}, e'). \tag{4.12}$$

We will also evaluate cross coherence using only cosine similarity. Then, cross coherence is purely context based and uses no semantic relatedness from links. This is achieved by omitting the SRL^* term in Eq. 4.12, i.e.

$$coh_{\cos}(e_{ij}, \{\mathbf{e}_l^c\}_{\substack{l=1 \\ l\neq i}}^{|\mathbf{M}|}) := \frac{1}{|\mathbf{M}|} \sum_{\substack{l=1 \\ l\neq i \\ \mathbf{e}_i^c \neq \mathbf{e}_l^c}}^{|\mathbf{M}|} \frac{1}{|\mathbf{e}_l^c|} \sum_{\substack{e' \in \mathbf{e}_l^c \\ e_{ij} \neq e'}} \cos(text(e_{ij}), text(e')). \tag{4.13}$$

Following the results obtained in the previous chapter, we also introduce a coherence weight based on topic distributions. In contrast to the previous chapter, topic distributions are not inferred on article texts but, to emphasise the co-occurrence of entities, on link anchor texts. More specifically, the documents used to train the topic model then consist of the concatenation of all link anchor texts l_a contained in the outlink collection $\mathbf{L}_{out}(e)$ of an entity e. Based on this, we introduce as thematic weight the Hellinger distance of the topic probability distributions inferred over the concatenation of link anchor texts $l_a \in \mathbf{L}_{out}(e)$ and $l_a \in \mathbf{L}_{out}(e')$

$$H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_{e'_{l_a}}) = \sum_{k=1}^{K} \sqrt{p_{\mathbf{L}_{out}(e)}(\phi_k) p_{\mathbf{L}_{out}(e')}(\phi_k)}, \tag{4.14}$$

with K the number of topics in the LDA model and $p_{\mathbf{L}_{out}(e)}$ and $p_{\mathbf{L}_{out}(e')}$ the topic probability distribution vectors for the concatenation of link texts $\{l_a\} \in \mathbf{L}_{out}(e)$ resp. $\{l_a\} \in \mathbf{L}_{out}(e')$ of the entities e and e'. The formulation of Hellinger distance in Eq. 4.14 is an alternative to that in Eq. 3.36 but it can be shown that they are equivalent. The thematic weight $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_{e'_{l_a}})$ then constitutes the thematic distance

over two link text collections and we use this weight as a replacement for the cosine similarity in Eq. 4.12, i.e.

$$coh_{\tau\mathrm{SRL}^*}(e_{ij}, \{\mathbf{e}_l^c\}_{\substack{l=1\\ l\neq i}}^{|\mathbf{M}|}) := \frac{1}{|\mathbf{M}|} \sum_{\substack{l=1\\ l\neq i\\ \mathbf{e}_i^c \neq \mathbf{e}_l^c}}^{|\mathbf{M}|} \frac{1}{|\mathbf{e}_l^c|} \sum_{\substack{e'\in \mathbf{e}_l^c\\ e_{ij}\neq e'}} H(\mathcal{T}_{e_{ij_{l_a}}}, \mathcal{T}_{e'_{l_a}}) \cdot \mathrm{SRL}^*(e_{ij}, e'). \quad (4.15)$$

To train the required topic model, we randomly chose 90k entities that have at least 10 outlinks, extracted all their link anchor texts and then trained a topic model with 500 topics on the generated documents.

So far we have defined the collective search procedure and the weighting of the retrieved candidates. Since still the sets $\mathbf{e}_i^c$ contain more than one candidate for each mention m_i, we now describe how we choose the *best fitting* candidate from this set.

Selection of Candidates from Collective Search

The selection of the best fitting candidate is the final result of the collective search procedure and determines prioritized candidate entities $e^{coh}(m_i)$. More specifically, from the collectively retrieved candidate entities $\mathbf{e}_i^c(m_i)$ we select one candidate entity $e^{coh}(m_i)$ for each mention m_i based on the product of collective search relevance weight w_r (Eq. 4.9) and cross coherence $coh_\times$ (Eq. 4.10):

$$e^{coh}(m_i) := \underset{e_i(m_i)\in \mathbf{e}_i^c(m_i)}{\arg\max} \; (w_r(e_i(m_i)) \cdot coh_\times(e_i(m_i), \{\mathbf{e}_l^c(m_l)\}_{\substack{l=1\\ l\neq i}}^{|\mathbf{M}|}). \quad (4.16)$$

That is, among all candidates $\mathbf{e}_i^c(m_i)$ for a mention m_i, we choose the entity that has maximum value for the product of collective search relevance weight $w_r(e_i)$ (Eq. 4.9) and cross coherence $coh_\times$ (Eq. 4.10). We use $coh_\times$ in a product with w_r to reduce the dominating effect of w_r as the latter is usually several orders of magnitudes higher. Importantly, note that we can only assign such a candidate $e^{coh}(m_i)$ to a mention m_i, if the set of $\mathbf{e}_i^c(m_i)$ is not empty. In the other case, we have no such candidate.

In an alternative formulation, we might incorporate the EMP of a candidate. But then popular candidates will dominate in most cases, even if their coherence is low. For instance, Shen et al. [2012] propose a similar global coherence measure, but, instead of computing the global coherence over all candidates as proposed here, the authors use a strong simplification and compute the global coherence only over those candidates that have highest EMP, no matter how well they fit to the context. Thus, less prominent entities are completely ignored. In contrast, we consider all candidates and investigate different weighting schemes as described above. When evaluating different cross coherence weights, we also determine the candidate $e^{coh}(m_i)$ using the specific weight for cross coherence computation. Then we use either coh_{SRL^*} (Eq. 4.11), $coh_{\cos\mathrm{SRL}^*}$ (Eq. 4.12), $coh_{\tau\mathrm{SRL}^*}$ (Eq. 4.15) or the purely contextual form $coh_{\cos}$ (Eq. 4.13) that omits SRL^*.

4.6.2 Prioritized Candidate Retrieval

In the first stage of candidate retrieval, we collectively treated all mentions $\mathbf{M}$ in the document. In the second stage of candidate retrieval, we treat each mention $m_i \in \mathbf{M}$ individually and create local, mention specific queries. More specifically, for each mention $m_i \in \mathbf{M}$, we combine the collectively retrieved candidate $e^{coh}(m_i)$, the local mention attributes as described in Section 4.5 and a title&redirect baseline. The title&redirect baseline is a candidate $e^{loc}(m_i)$ that we retrieve by matching the (expanded) name of a mention against the fields `title` and `redirect` in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ and then choosing the returned entity with highest score. Here, we require an exact albeit case-insensitive match.

Baselines using only title and redirect information have been reported to often yield excellent linking performance (Hachey et al. [2013]). Here, we combine this baseline candidate $e^{loc}(m_i)$ directly into our model and treat it similarly to the $e^{coh}(m_i)$ candidate. Also note that, as already stated, not all mentions need have an e^{coh} candidate. In such cases, the title&redirect baseline may serve as a reliable baseline. Then, to combine global document level information and local mention specific information, we will place a query against $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ that covers the candidates $e^{coh}(m_i)$ and $e^{loc}(m_i)$, if available, as well as the attributes of a mention m_i. This query is formed as follows and depicted in Alg. 2.

First, if either $e^{loc}(m_i)$ or $e^{coh}(m_i)$ exist, we emphasise a match on the title of either candidate by adding boosted query terms on title fields (line 2 and line 4 in Alg. 2). This is comparable to the usage of a prior, therefore we also call this prioritized candidate retrieval. Note that adding boosted query terms pushes the returned result towards specific candidate entities that also receive a higher score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}$. Here, we imply that the two candidates e^{coh} and e^{loc} are especially important and give the respective query terms a five times higher weight than the remaining terms treating local contextual attributes. Naturally, if either of these candidates does not exist, the respective query terms are omitted.

Now, since both candidate priors are not fail safe, we use contextual information as additional evidence and add the local attributes of the mention to our query. We add query terms on all alias fields using the name of the mention (line 5), terms on type fields using its type (line 7) and finally terms on contextual fields using the context of the mention (line 9). As depicted in Alg. 2, the creation of this query is parametrized through *search coverage* flags. Using always the name of a mention as well as the title&redirect baseline candidate e^{loc}, this allows us to experimentally evaluate the influence of different attributes by either adding or omitting the respective query terms. We will demonstrate the influence of search coverage as well as prioritization on e^{coh} candidates in Section 4.8.2.

Then we use this query to search $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ to obtain a ranked set of candidate entities $\mathbf{e}_1^*(m_i) \subset \boldsymbol{\mathcal{I}_{\mathcal{W}}}$ for each mention m_i (line 11). We use a limit of three on the cardinality of each $\mathbf{e}_i^*(m_i)$ (line 12), a figure we experimentally found to be suffi-

Algorithm 2: Candidate retrieval (Stage 2)

Input: A mention m_i with attributes $name(m_i)$, $type(m_i)$, $text(m_i)$, candidates $e^{coh}(m_i)$ and $e^{loc}(m_i)$, search coverage flags, threshold τ.
Output: A list of candidates $\mathbf{e}^*(m_i)$ for m_i (potentially empty)

1 create an initially empty query q
2 $q \leftarrow q_{\text{title}}(title(e^{loc}(m_i)), 5)$ `// add weighted terms for` $title(e^{loc})$
3 **if** *prioritize on collective search candidate* **then**
4 $\quad q \leftarrow q \wedge q_{\text{title}}(title(e^{coh}(m_i)), 5)$ `// add weighted terms for` $title(e^{coh})$
5 $q \leftarrow q \wedge q_{\text{alias}}(name(m_i))$ `// add alias terms for` $name(m_i)$
6 **if** *use type information* **then**
7 $\quad q \leftarrow q \wedge q_{\text{type}}(type(m_i))$ `// add type terms`
8 **if** *use context information* **then**
9 $\quad q \leftarrow q \wedge q_{\text{text}}(text(m_i))$ `// add context terms`
10 $p^*_{in} \leftarrow \min\left(p_{in}(e^{loc}(m_i)), p_{in}(e^{coh}(m_i)), 5\right)$ `// adjust popularity prior`
11 search $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using q with mandatory term on inlink prior p^*_{in}, i.e. $q_{p_{in} > p^*_{in}}$
12 keep the 3 retrieved entities with highest score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q, e)$ as $\mathbf{e}^*(m_i)$
13 **if** $\max_{e^*_j \in \mathbf{e}^*(m_i)} s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q, e^*_j) \leq \tau$ **then**
14 $\quad$ search $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using q without mandatory term on inlink prior p^*_{in}
15 $\quad$ keep the 3 retrieved entities with highest score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q, e)$ as $\mathbf{e}^*(m_i)$
16 **return** $\mathbf{e}^*(m_i)$

cient. Initially, we also require each entity $e^*_i(m_i) \in \mathbf{e}^*_i(m_i)$ to have at least 5 inlinks (line 10). This popularity prior aims at filtering out rarely referenced entities. If this prior exceeds the number of inlinks of either $e^{loc}(m_i)$ or $e^{coh}(m_i)$, we automatically adapt it accordingly. This is necessary since, given that the prior term on inlinks is a mandatory term, the prioritized entities could otherwise not be retrieved from $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$. We fully neglect the popularity prior if the maximum observed score using this search is less than a threshold of τ (line 13). Doing so, we account for entities that are either very rarely referenced or have very short articles and thus will usually obtain very small scores $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}$. We have evaluated different thresholds and, observing that Lucene returns a score smaller than 1 only for very unrelated documents, set $\tau = 1$.

Now, since this candidate retrieval can only return entities contained in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$, there are two possible outcomes: we either arrive at an empty candidate set or at a ranked set of candidates $\mathbf{e}^*_i(m_i)$. The first case may arise when the mention refers to an uncovered entity (NIL) or to an erroneously missing entity (NIL*). As uncovered entities are naturally not contained in the index, we can postulate that the method worked correctly in that case. The same holds for missing entities.

However, we may also fail to retrieve a candidate. This is the case when we use only name information but the mention's name appears in none of the queried

fields of the index $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$, for instance when the name is an entirely new one such as an unknown translation. When the returned candidate set for a mention m is empty, the mention is automatically linked to NIL as there is no available alternative candidate.

In the other case, the ranked candidate set is consolidated in order to fine tune the ranking and to detect uncovered entity mentions. For this, we use a supervised Ranking SVM as described in Section 3.5.3. In our experimental section, we will evaluate entity linking both in an unsupervised variant that uses the top-ranked entities retrieved from the index as predictions (Section 4.8.2) as well as in a supervised variant, where candidates are consolidated through the Ranking SVM (Section 4.8.4). The details and design of candidate consolidation are described next.

4.7 Candidate Consolidation

As already stated, the search in $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ returns for each mention $m_i \in \mathbf{M}$ either an empty set or a set of ranked candidates $\mathbf{e}_i^*(m_i)$. In the first case we have no choice but to resolve the respective mention to NIL. The latter case is the desirable case and far more interesting. Indeed, it splits up into three subcases that need to be handled. First, the correct entity may be the top ranked candidate in $\mathbf{e}_i^*(m_i)$. Here, we would be fine using only the index prediction as final output. Secondly, the correct entity may have a lower rank (i.e. a lower score) or thirdly may not at all be contained in the retrieved candidates, as is the case for uncovered entities.

Therefore, we validate the retrieved candidate set $\mathbf{e}_i^*(m_i)$ using a supervised method, i.e. a linear Ranking SVM, that has access to additional information. Depending on its confidence, this classifier may re-rank the search result and assign the final prediction based on features both from candidate retrieval as well as features that are contained only implicitly or latently in Wikipedia, i.e. features such as coherence, EMP and similarity of topic distributions.

To apply candidate consolidation, we first collect the retrieved candidate sets $\mathbf{e}_1^*(m_1), \ldots, \mathbf{e}_k^*(m_k)$ and represent each candidate $e_i^* \in \mathbf{e}_i^*(m_i)$ by a vector of indicative features. These features are summarized in Tab. 4.2 and described in detail next.

The first group of features is computed from the index score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q, e_i^*)$ (Eq. 4.6) where q is the query formed according to Alg. 2. More specifically, we use the score both in the original form as given by Eq. 4.6, as well as in a log-variant, i.e.

$$s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}},\log}(q, e_i^*) = \log\left(s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q, e_i^*) + 1\right) \in \mathbb{R}^+, \tag{4.17}$$

with the usual addition of 1 to ensure a positive value. Additionally, we explicitly relate the score of a candidate $e_i^*(m_i) \in \mathbf{e}_i^*(m_i)$ to the scores of all candidates $\mathbf{e}_i^*(m_i)$

Table 4.2: Features for supervised candidate consolidation

feature	description
$s_{\mathcal{I}_\mathcal{W}}(q, e_i^*(m_i))$	the index score for the candidate $e_i^*(m_i)$ (Eq. 4.6) given the query q formed in prioritized candidate retrieval (Alg. 2)
$s_{\mathcal{I}_\mathcal{W},\log}(q, e_i^*(m_i))$	log value of the original index score (Eq. 4.17)
$s_{\mathcal{I}_\mathcal{W},norm}(q, e_i^*(m_i))$	the index score of the candidate, normalized with respect to all candidates (Eq. 4.18)
$s_{\mathcal{I}_\mathcal{W},rank}(q, e_i^*(m_i))$	the index score of the candidate, ranked with respect to all candidates (Eq. 4.19)
$coh_\times(e_i^*(m_i), \{\mathbf{e}_l^*(m_l)\}_{\substack{l=1 \\ l\neq i}}^{\lvert\mathbf{M}\rvert})$	the candidate's cross coherence weight given a specific weighting rule (Eqs. 4.11 to 4.13 and 4.15)
$mix(e_i^*(m_i))$	the product of cross coherence weight and index score (Eq. 4.20)
$H(\mathcal{T}_{e_{l_a}(m_i)}, \mathcal{T}_{m_i})$	the Hellinger distance of topic distributions over mention and candidate context (Eq. 4.21)
$p(e_i^*(m_i)\lvert m_i)$	the entity-mention probability for the candidate (Eq. 2.7)
NIL-feature	a binary feature that is active only for NIL candidates (Eq. 3.19)

using the normalized form

$$s_{\mathcal{I}_\mathcal{W},norm}(q, e_i^*(m_i)) = \frac{s_{\mathcal{I}_\mathcal{W}}(q, e_i^*(m_i))}{\sum_{e_i^{*\prime}(m_i)\in\mathbf{e}_i^*(m_i)} s_{\mathcal{I}_\mathcal{W}}(q, e_i^{*\prime}(m_i))} \in [0,1] \tag{4.18}$$

as well as the ranked form

$$s_{\mathcal{I}_\mathcal{W},rank}(q, e_i^*(m_i)) = \frac{s_{\mathcal{I}_\mathcal{W}}(q, e_i^*(m_i))}{\arg\max_{e_i^{*\prime}(m_i)\in\mathbf{e}_i^*(m_i)} s_{\mathcal{I}_\mathcal{W}}(q, e_i^{*\prime}(m_i))} \in [0,1]. \tag{4.19}$$

The normalized form in Eq. 4.18 is the index score of a candidate divided by the sum of scores for all candidates $\mathbf{e}_i^*(m)$. The ranked score in Eq. 4.19 is the index score of a candidate divided by the maximum score returned for any of the candidates in $\mathbf{e}_i^*(m_i)$. In contrast to the original score or the variant in logarithmic form, the latter two representations have a closed range of $[0,1]$ whereas we may observe a high variance in the absolute scores of candidates. For instance, a candidate representing the ground truth entity may have a score $s_{\mathcal{I}_\mathcal{W}}(q, e_i^*(m_i))$ of 0.04, 10, 20 or even higher, depending on the query q and the number of matches for $e_i^*(m_i)$. Thus, the normalized variant gives also a smoother representation compared to the absolute score. Furthermore, these features explicitly relate to the scores of other candidates and encode the position of a candidate in the ranked list of all candidates.

Another feature is the cross coherence weight of the candidate $e_i^*(m_i)$ computed as in Eq. 4.10 but now in relation to the improved set of candidate entities $\mathbf{e}^*$. Furthermore, to account for the mixture of index score and cross coherence, i.e. the local and document-level information, we also use the product of these values as a feature:

$$mix(e_i^*(m_i)) = s_{\boldsymbol{\mathcal{I}_W}}(q, e_i^*(m_i)) \cdot coh_{\times}(e_i^*(m_i), \{\mathbf{e}_l^*(m_l)\}_{\substack{l=1 \\ l \neq i}}^{|\mathbf{M}|}). \tag{4.20}$$

We also evaluate thematic information not only for relatedness weighting but also as a distinct ranking feature. For this, we use the same topic model over link anchor texts and infer the topic distribution $\mathcal{T}_{e_i^*}$ on the outlink anchor texts of the candidate e_i^*. Analogously, to infer the topic distribution $\mathcal{T}_m$ of the mention context, we use the surface forms of all entity mentions in the document. As in the previous chapter, we use local boosting and give the local context words of a mention m_i a five-times higher weight to arrive at localized topic distribution $\mathcal{T}_{m_i}$ for each mention m_i. Again, this is motivated by the assumption that local context words are especially high important for disambiguation. Then, analogous to the thematic coherence weight $\Delta_{\mathcal{T}}$ in Eq. 4.14, we compute the Hellinger distance over the two topic distributions $\mathcal{T}_m$ and $\mathcal{T}_{e_{l_a}}$

$$H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m) = \sum_{k=1}^{K} \sqrt{p_{\mathbf{L}_{out}(e)}(\phi_k) p_m(\phi_k)}, \tag{4.21}$$

with K the number of topics in the LDA model, $p_{\mathbf{L}_{out}(e)}$ the topic probability distribution vector for the concatenation of link anchor texts $\{l_a\} \in \mathbf{L}_{out}(e)$ and p_m the topic probability distribution vector for the concatenation of all surface forms of entity mentions in the document referencing m. This value is then used as a dedicated feature for the Ranking SVM. Technically, this is the only feature that can not be computed directly from any of our indices. Since it is computationally also the most expensive feature as it requires a trained topic model, we will evaluate its influence separately in our experiments.

Given the effectiveness demonstrated by other approaches, we also make use of the entity-mention probability EMP $p(e_i^*(m_i)|m_i)$ (cf. Eq. 2.7) and employ it as a prior feature. Note that this feature is only available for *known* mentions contained as link anchor text l_m in the link index $\boldsymbol{\mathcal{I}_L}$. If the mention was never used to reference e_i^*, this feature naturally has value zero.

Each of the described features is then scaled from training data. We record for each feature the highest and lowest instantiation and then clamp feature instantiations on test data to the respective range. This is especially important for the index score feature that is technically not bounded to a specific range and may take on very high values.

Finally, the threshold for the detection of uncovered entity mentions is learned from a dedicated NIL-feature that was proposed in Bunescu and Pasca [2006] and

described by us in Section 3.5.3. Following Bunescu and Pasca [2006], we add for each non-empty candidate set $\mathbf{e}_i^*(m_i)$ a NIL-candidate for which the representing vector has only the NIL-feature (cf. Eq. 3.19). The threshold is then learned automatically from the weight of this feature.

Then, the prediction $\hat{e}(m_i)$ for a mention m_i is given by

$$\hat{e}(m_i) = \begin{cases} \arg\max_{e \in \{\mathbf{e}_i^*(m_i) \cup \{\text{NIL}\}\}} \text{rank}(x_e), \\ \text{NIL, if } \mathbf{e}_i^*(m_i) = \emptyset \end{cases} \tag{4.22}$$

where x_e is the vector of the described features representing an entity e and $rank(x_e)$ the Ranking SVM prediction value (cf. Eq. 3.17). In the first case, the prediction is the vector of the entity ranked highest by the Ranking SVM. This vector can either represent NIL or an entity $e \in \boldsymbol{\mathcal{W}}$. The second case accounts for those cases where we could not retrieve a candidate from our index and then automatically resolve a mention to NIL.

Now, applying the Ranking SVM for potential re-ranking on each candidate set $\mathbf{e}_i^*(m_i)$, yields the final output of our entity linking model. This is the (disambiguated) list of input mentions, where each mention m_i is linked either to an unique entity in Wikipedia or to NIL, i.e. $\{\hat{e}(m_1), \ldots, \hat{e}(m_k)\}$ with $\hat{e}(m_i) \in \boldsymbol{\mathcal{W}} \cup \{\text{NIL}\}$.

The training of this model will be described along with the experimental evaluation of the proposed candidate retrieval and consolidation in the following section.

4.8 Evaluation

In the following we will evaluate all components of the proposed linking model. We will compare our method to a representative selection of five recent works, namely Kulkarni et al. [2009], Han et al. [2011], Ratinov et al. [2011], Hoffart et al. [2011b] and Milne and Witten [2008b], using the corpora and performance measures introduced in Section 4.2. Doing so, we present the first thorough comparison of these recent entity linking systems and provide a unified view on the variety of proposed performance measures. We give the results as published in Pilz and Paaß [2012] and provide additional experimental evaluation of candidate retrieval and candidate consolidation.

The remainder of this section is structured as follows. First, we briefly review the benchmark corpora and detail the applied preprocessing steps (Section 4.8.1). Then, we assess the quality of candidate retrieval and evaluate this part in isolation (Section 4.8.2). To do so, we omit candidate consolidation and demonstrate the effect of different search coverages as well as the weighting factors proposed for cross coherence. To employ candidate consolidation, we first depict the training procedure of the underlying model in Section 4.8.3. Lastly, we show the effect of candidate consolidation, again for different search coverages as well as cross coherence weighting

Table 4.3: Benchmark corpora by ground truth annotations $e^+(m)$. The table shows the total number of documents and mentions in these documents. Mentions are broken up by being linked to Wikipedia ($e^+(m) \in \mathcal{W}$) or to NIL ($e^+(m) = \text{NIL}$). The number in brackets is the number of missing entities NIL*.

corpus	**D**	**M**	$e^+(m) \in \mathcal{W}$ ($e^+(m) = \text{NIL}^*$)	$e^+(m) = \text{NIL}$
MSNBC	20	755	658 (18)	97
ACE	36	306	257 (3)	49
AQUAINT	50	727	727 (25)	0
CoNLLb	228	4363	4363 (46)	0
IITB	104	11185	11185 (1746)	0

factors. This will include the usage of different performance measures for a better comparability with related work (Section 4.8.4).

4.8.1 Benchmark Corpora

We evaluate our method on the benchmark corpora introduced in Section 4.2, i.e. **MSNBC**, **ACE**, **AQUAINT**, **CoNLLb** and **IITB**. All of these corpora are annotated with mentions and their respective ground truth entities as shown in Tab. 4.3. For **MSNBC** we use the updated version published by Ratinov et al. [2011]. The corpus contains the same documents as the version used in Cucerzan [2007] but mentions are linked to a more recent version of Wikipedia which means that about 30 previously uncovered entity mentions are now covered.

As already stated, these benchmark corpora vary in annotation scheme as well as the types of mentions (cf. Tab. 4.4). While the first four corpora contain mostly named entity mentions, Kulkarni et al. [2009] aimed for aggressive linkage and annotated all interesting mentions in the web documents constituting **IITB**. Thus, as Tab. 4.3 shows, we also observe the highest number of mentions per document for this corpus and, as depicted in Tab. 4.4, with about 83% also the highest number of mentions referring to conceptual entity types.

Additional to differing mention types, there are also differences in the annotation scheme that render comparison difficult. For instance, in **CoNLLb** the mention Taiwan is always linked to Republic of China, even though a distinct article on Taiwan exists in Wikipedia. Interestingly, the latter was always chosen as ground truth target for mentions of Taiwan by the annotators of **ACE**.

Moreover, **CoNLLb** contains many news articles about sport events. These documents consist not only of natural language text but contain many tables. These variations make it challenging to apply the same system to different corpora. Furthermore, Hoffart et al. decided to ignore uncovered entities during evaluation and consequently roughly 20% of the mentions. To allow for a fair comparison with

Table 4.4: Benchmark corpora by mention type $type(m)$. We give the number of persons, locations, organizations an miscellaneous entities, 'not available' indicates that we could not retrieve a type for the mention using Apache OpenNLP NER.

	$type(m)$				
corpus	person	location	organization	miscellaneous	not available
MSNBC	213	186	144	1	211
ACE	43	116	71	-	76
AQUAINT	61	134	96	4	432
CoNLLb	977	1388	1458	540	-
IITB	402	557	596	332	9298

Hoffart et al.'s system AIDA, we need to follow this restriction when applying our method on **CoNLLb** and therefore ignore mentions resolving to NIL as well.

The statistics on entity types given in Tab. 4.4 are derived from the application of the Apache OpenNLP NER tool on these corpora. We use the same model to obtain named entity type annotations and the Apache OpenNLP PoS tagger to obtain PoS tags. As described in Section 4.5, PoS tags are used to extract local mention contexts consisting of nouns and named entity types are used for name expansion as well as type sensitive search. We apply the NER tool on all corpora except for **CoNLLb** as for this corpus the mention types are already given. Thus, on **CoNLLb**, we only need to run the PoS tagger. Additionally, as for the Wikipedia article texts, we use the Lucene standard analyzer for English for tokenization and stemming[1].

For all corpora we proceed as follows: given a mention m, we first check if the assigned ground truth entity $e^+(m)$ is contained in $\boldsymbol{\mathcal{I}}_{\boldsymbol{\mathcal{W}}}$. If this is not the case, but the annotators assigned the mention some $e^+(m) \neq \text{NIL}$, we adjust the ground truth to NIL^* by setting $e^+(m) = \text{NIL}^*$. The procedure is the same for entities that do no longer exist in Wikipedia. Doing so, we account for missing entities that are always considered during evaluation. We observe with about 10% most missing entities on **IITB**, on the other corpora this amounts to no more than 3% of the mentions.

4.8.2 Evaluation of Candidate Retrieval

First, we evaluate the quality of the proposed candidate retrieval. In these experiments, we omit candidate consolidation through the Ranking SVM and set the prediction to the top ranked candidate returned by the index search, i.e.

$$\hat{e}(m) = \arg\max_{e \in \mathbf{e}^*(m)} s_{\boldsymbol{\mathcal{W}}}(q, e), \tag{4.23}$$

[1]This analyzer is also available for other languages, e.g. German and French.

where the query q is formed according to Alg. 2. This corresponds to an unsupervised entity linking model for which we here show performance in Ratinov et al.'s BoT measure, i.e. F_{BoT}, to be consistent with the results published in Pilz and Paaß [2012]. For this model, we evaluate the influence of search coverage as well as the effect of prioritization on e^{coh} candidates retrieved from collective search for the different weighting factors of cross coherence.

For evaluation, we distinguish among the following degrees of search coverage that treat the different attributes of a mention as described in Section 4.5:

1. **name coverage** ($\mathbf{SC}_{n^*}$, $\mathbf{SC}_n$): We use only the surface form of the mention $name(m)$ to place queries against alias fields (line 5 in Alg. 2). We evaluate the name of a mention both in its original form ($\mathbf{SC}_{n^*}$) as well as in the expanded form ($\mathbf{SC}_n$). Here, we use no other information such as type or context in the query terms.
2. **name and type coverage** ($\mathbf{SC}_{nt}$): We extend the query with terms treating the type $type(m)$ as assigned to a mention through the NER model. This information may be missing for some entities but if available activates line 7 in Alg. 2.
3. **name, type and context coverage** ($\mathbf{SC}_{ntc}$): In the full search coverage, we additionally query context fields using the mention context $text(m)$. This additionally activates line 9 in Alg. 2.
4. **prioritization**: To evaluate the quality of the candidates retrieved from collective search, we prioritize on the candidate $e^{coh}(m)$ using the baseline cross coherence weight coh_{SRL^*} (Eq. 4.11). This activates line 4 in Alg. 2.

Search coverage is evaluated cumulatively, i.e. experiments using type information use expanded names, experiments using contextual evidence use expanded names and type information. We use the baseline cross coherence weight coh_{SRL^*} for the collective search candidate and evaluate the effect of different weight factors separately.

Tab. 4.5 shows the results obtained for different search coverages on the benchmark corpora. In the table, the figure left from the arrow is obtained for varying degrees of search coverage, the figure right from the arrow shows how the performance is influenced when we additionally use collective search and a prioritization on e^{coh} candidates.

We observe that the proposed name expansion in $\mathbf{SC}_n$ has a positive effect and generally increases performance or at least yields similar results to the usage of the original name. The increase in performance is the highest on **MSNBC** which can be explained by the annotation scheme: for this corpus, all entity mentions are to be linked and not only the first ones that typically use the full name of the underlying entity. Since later mentions of an entity in a document are often abbreviated, name expansion is especially useful.

Table 4.5: Influence of search coverage and prioritization on e^{coh} candidate on F_{BoT} performance (all values in %). We omit candidate consolidation and use the candidate with highest score $s_{\mathcal{I}_{\mathcal{W}}}$ as prediction (cf. Eq. 4.23). The figure left from the arrow is obtained without prioritization on e^{coh}, the figure right from the arrow is obtained with prioritization. Apart from **IITB**, candidate prioritization consistently improves performance on all corpora.

		increased search coverage →		
corpus	$\mathbf{SC}_{n^*}$	$\mathbf{SC}_{n}$	$\mathbf{SC}_{nt}$	$\mathbf{SC}_{ntc}$
MSNBC	75.20 ↗ 76.75	77.86 ↗ 78.98	77.97 ↗ 78.92	77.86 ↗ **79.43**
ACE	76.17 ↗ 78.03	76.44 ↗ 78.19	76.43 ↗ **78.19**	76.98 ↗ 78.03
AQUAINT	81.27 ↗ **81.80**	81.16 ↗ 81.69	80.61 ↗ 81.69	81.58 ↘ 80.97
CoNLLb	64.27 ↗ 68.34	65.17 ↗ 69.02	66.32 ↗ 74.43	70.18 ↗ **77.10**
IITB	75.90 ↘ 75.48	76.13 ↘ 75.60	**76.14** ↘ 75.63	73.67 ↘ 72.49

Interestingly, Cucerzan [2007] reported that his title&redirect baseline using exact matches in combination with the EMP prior achieved an accuracy of 51.7% on **MSNBC**. This is notably lower compared to the accuracy value of 63.7% obtained with our name baseline that does not even yet use EMP.

The usage of type information ($\mathbf{SC}_{nt}$) has only marginal influence. This is not surprising considering our model design: we did not focus on this attribute in order to avoid type dependency and error propagation from NER models. Only on **CoNLLb** we observe a slight increase of about 1 pp in performance. Note that this corpus was designed for the evaluation of NER models and contains high quality manual annotations of named entity types. Given that for the other corpora the influence of this attribute is negligible and does also not dramatically decrease performance, we argue that its usage is in general acceptable.

We find that the usage of contextual information ($\mathbf{SC}_{ntc}$) is also helpful in general. For **CoNLLb** we observe the highest influence of contextual information, boosting performance by about 5 pp compared to the purely name based search, and by about 4 pp compared to the search using type information. We assume that this is because this corpus is from editorial news documents where authors use canonical names and give special attention to clarify the ambiguity of mention by providing disambiguation terms close to the mention. On web documents such as **IITB** this may be missing. Here, somewhat counterintuitively the usage of contextual information leads to a notable decrease in performance.

Concerning candidate prioritization, we find a general improvement in performance on most of the corpora and nearly all configurations of search coverage. The highest increase is again observed on **CoNLLb** with up to 7 pp but also on the other corpora with an average increase of about 1 pp. Again, **IITB** is the exception. This may stem from the comparably high percentage of mentions denoting missing enti-

Table 4.6: F_{BoT} performance on the benchmark corpora for different cross coherence weights using full search coverage $\mathbf{SC}_{ntc}$ (all values in %). We omit candidate consolidation and use the candidate with highest score $s_{\mathcal{I}_{\mathcal{W}}}$ as prediction (cf. Eq. 4.23). We observe no notable difference among the weighting factors, for **MSNBC** there is no difference at all.

	weighting factors in cross coherence $coh_{\times}$			
corpus	coh_{SRL^*}	$coh_{\tau\text{SRL}^*}$	$coh_{\cos\text{SRL}^*}$	$coh_{\cos}$
MSNBC	**79.43**	**79.43**	**79.43**	**79.43**
ACE	**78.03**	**78.03**	77.54	77.54
AQUAINT	80.09	80.83	80.78	**80.86**
CoNLLb	77.10	**77.38**	77.35	77.37
IITB	72.49	72.39	72.52	**72.57**

ties in this corpus. Note that we did not manually check the existence of the given ground truth entities and thus missing entities may indeed denote also truly uncovered entities. In that case, collective search can not retrieve as many relevant source entities since this fraction is anti-proportional to the number of matches on these entities link text fields. Another explanation is that for this corpus, entities are just not as semantically related as for the other corpora. Interestingly, this goes along with the results published in Kulkarni et al. [2009] showing that their collective approach performs only one point in percentage better than a local name baseline using popularity priors.

To summarize the findings so far, we observe that the more information we use, the better the performance of the linking model in general. Thus we evaluate now the different weighting factors for cross coherence (Eqs. 4.11 to 4.13 and 4.15) using the full search coverage $\mathbf{SC}_{ntc}$. As Tab. 4.6 shows, the influence of the proposed weighting factors is not striking. Comparing to the baseline coh_{SRL^*} using no additional weight on semantic relatedness, we find no difference for **MSNBC** and only minor improvements for the other corpora when varying the weighting scheme. However, this result is not very surprising, since the influence of cross-coherence weights on purely search based prediction is not very strong. It affects only the identity of the collective search candidate and as we see from the obtained results, this does not happen often. Nevertheless, we will still evaluate the different weighting schemes in the experiments on candidate consolidation. As stated in Section 4.7, these weights are used in two dedicated features and thus may have higher influence in the context of candidate consolidation.

For a better interpretability of cross coherence influence, we also analysed the average cross coherence of the ground truth entities in the benchmark corpora. The results are given in Tab. 4.7. Independently of the used weight, the average cross coherence is the highest on **CoNLLb**. This may be due to the underlying nature of

Table 4.7: For each of the proposed cross coherence weights, the table shows the average cross coherence of ground truth entities in the benchmark corpora. The resulting values are strongly correlated ($p < 0.02$) even though $coh_{\cos}$ does not use relational information from SRL*.

	weighting factors in cross coherence $coh_{\times}$			
corpus	coh_{SRL^*}	$coh_{\tau\text{SRL}^*}$	$coh_{\cos\text{SRL}^*}$	$coh_{\cos}$
MSNBC	0.381	0.179	0.104	0.215
ACE	0.354	0.139	0.096	0.211
AQUAINT	0.310	0.119	0.070	0.161
CoNLLb	0.402	0.208	0.134	0.262
IITB	0.224	0.087	0.041	0.112

the documents at hand, but we should also keep in mind that this corpus was chosen as reference to demonstrate the effect of relational or collective information (Hoffart et al. [2011b]). Interestingly, the average coherence is the lowest on **IITB**, a dataset that was also published by a collective approach (Kulkarni et al. [2009]). We assume that here the average coherence is low because of the comparably high number of mentions per document. Fig. 4.3 depicts the average cross coherence over ground truth entities in relation to the increase in F_{BoT} performance when prioritizing on e^{coh} candidates. The figure implies that the increase in performance is related to the average cross coherence and that a high value results in a high increase in F_{BoT} performance.

To summarize, we observe that unsupervised entity linking using only the name of a mention already achieves fairly good results. The F_{BoT} ranges between 64% and 81% with an average of about 75% across the different corpora. This is due to our carefully chosen alias resources as well as the title&redirect baseline candidate $e^{loc}(m)$. We can increase performance through the usage of collective search candidates but this increase varies by the corpus at hand. This goes along with the results obtained by Varma et al. [2009] who showed that, while depending on the corpus at hand, an elaborate candidate selection method has major impact on performance. This may either reduce the required complexity of the consecutive candidate consolidation or may even render it obsolete.

However, since we did not incorporate any kind of threshold on the index score $s_{\mathcal{I}_{\mathcal{W}}}$, this unsupervised linking model can not handle uncovered entity mentions. Instead of empirically determining this threshold, we prefer learning an appropriate candidate consolidation model. The training procedure as well as the results obtained with this candidate consolidation model will be described next.

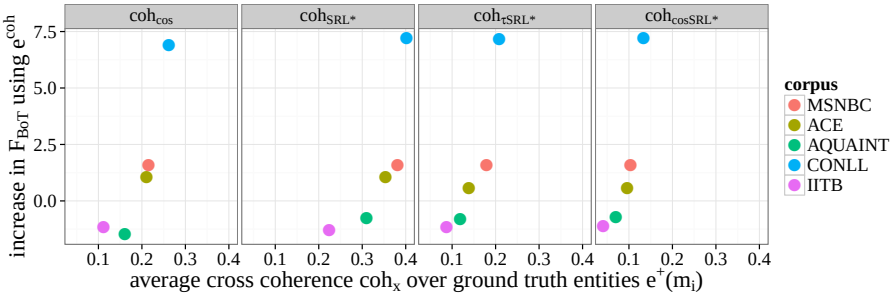


Figure 4.3: The figure shows for each cross coherence weight the averaged cross coherence of ground truth entities in the benchmark corpora in relation to the increase in F_{BoT} performance when prioritizing on e^{coh} candidates. The figure implies that the increase in performance is related to the average cross coherence and that a high value may result in a high increase in F_{BoT} performance.

4.8.3 Training the Candidate Consolidation Model

Neither of the described benchmark corpora includes a training set and each corpus may only be used for testing. Instead of using Wikipedia references for the training of our candidate consolidation model, we follow Hoffart et al. [2011b] and use the **CoNLL train** corpus, as this corpus reflects the nature of the benchmark corpora more than Wikipedia references. **CoNLL train** is a collection of 946 Reuters news articles from the CoNLL 2003 shared task. The named entity mentions in these documents were annotated by Hoffart et al. [2011b] with links to Wikipedia entities as well as a placeholder to indicate mentions of uncovered entities.

Unfortunately, there are some issues with this corpus that we need to solve in order to use it. These issues are inconsistencies in the annotations in **CoNLL train** that are presumably due to inter-annotator disagreement (20%) or candidate selection. First, the authors annotated all mentions that could not be directly mapped to YAGO2 with NIL. This affects mostly abbreviations and acronyms. For instance, while the mention European Union is linked to the appropriate Wikipedia entity, it's acronym EU is linked to NIL. Similarly, surnames such as Fischler are linked to NIL while the long form Franz Fischler appearing in the same document is linked correctly. This is also the case for abbreviations such as M. Moxon. This mention refers to the cricket player Martyn Moxon who also has a corresponding article in Wikipedia.

Since Hoffart et al. ignored uncovered entity mentions both in training and evaluation, their model is not affected severely by these inconsistencies. In contrast, this may lead to considerable errors for our approach when training our system on this corpus. Thus, we use here an additional pre-processing step and verify the links of all mentions marked as uncovered by Hoffart et al.. For each mention annotated as uncovered, we perform a look up both in $\mathcal{I}_{\mathcal{W}}$ as well as a simple web look up

in online Wikipedia. If this look up yields a positive result and hence a potentially covered entity, we ignore this mention and do not use it for training in order to prevent inconsistencies[1]. Thus from the total number of 23.499 named entity mentions in this corpus, we arrive at 18923 mentions that we may use for training.

For all of the mentions in **CoNLL train**, we generate labelled vectors x_e representing positive and negative training examples in the same way as we generate disambiguation candidates. For instance, a positive example is created using the correct candidate $e^+(m_i) \in \boldsymbol{\mathcal{W}}$ for a covered mention and negative examples are created using all other candidates $e_i^*(m_i) \in \{\mathbf{e}_i^*(m_i) \cup \{\text{NIL}\}\} \setminus \{e^+(m_i)\}$. Consequently, we provide a NIL candidate for each mention in order to learn the threshold for the prediction of uncovered entities. We proceed analogously for mentions with $e^+(m) = \text{NIL}$ where the NIL candidate is used to create a positive example and all other candidates $\mathbf{e_i}^*(m) \in \boldsymbol{\mathcal{W}}$ are used to create negative examples.

4.8.4 Evaluation of Candidate Consolidation

To demonstrate the effect of candidate consolidation, we evaluate our model in all of the configurations we have described in the previous section. This includes the different search coverages and weighting factors in cross-coherence but also the usage of thematic similarity as dedicated feature. To summarize, in this last section of experimental evaluation we want to answer the following questions:

Question 1 Is there a configuration that outperforms all other competitor methods on all corpora?

Question 2 Has the prioritization on the collective search candidate e^{coh} a positive effect on performance in general?

Question 3 Is there a cross coherence weight that performs best on all corpora?

Question 4 What is the average error reduction compared to baselines using only name information?

To answer these questions, we first evaluate how search coverage affects candidate consolidation. To do so, we evaluate candidate consolidation independently from collective search and omit the prioritization on collective search candidates e^{coh}. Doing so, we omit relational information and neither use collectively retrieved candidates nor the features derived from the cross coherence weight of these candidates. We also omit the thematic similarity feature (Eq. 4.21), since being trained on link anchor texts in Wikipedia, the underlying LDA model also latently covers relational information. Instead, we use purely index based features derived from candidate retrieval, i.e. the variants reflecting the index score $s_{\mathcal{I}_{\mathcal{W}}}$ (Eqs. 4.17 to 4.19) of the retrieved candidate, as well as the entity-mention probability EMP (Eq. 2.7). To

[1]The test dataset **CoNLLb** is not affected by this in any way.

Table 4.8: F_{BoT} performance on the benchmark corpora for different search coverages (all values in %). Candidates are consolidated by the Ranking SVM but the prioritization on e^{coh} candidates is omitted. The figure left from the arrow is obtained without candidate consolidation (cf. Tab. 4.5), the figure right from the arrow is obtained with candidate consolidation.

	increased search coverage →			
corpus	$\mathbf{SC}_{n^*}$	$\mathbf{SC}_{n}$	$\mathbf{SC}_{nt}$	$\mathbf{SC}_{ntc}$
MSNBC	75.20 ↗ 84.76	77.86 ↗ **87.69**	77.97 ↗ 86.1	77.86 ↗ 86.43
ACE	76.17 ↗ 84.15	76.44 ↗ 84.46	76.43 ↗ 83.3	76.98 ↗ **86.49**
AQUAINT	81.27 ↗ **84.87**	81.16 ↗ 84.77	80.61 ↗ 84.41	81.58 ↗ 84.81
CoNLLb	64.27 ↗ 68.36	65.17 ↗ 69.05	66.32 ↗ 68.54	70.18 ↗ **70.42**
IITB	75.90 ↗ 77.99	76.13 ↗ 78.19	76.14 ↗ **78.74**	73.67 ↗ 78.01

learn the threshold for the decision on uncovered entity mentions, we use the dedicated NIL feature (Eq. 3.19) that is active only for the vector representing the NIL candidate.

In line with the evaluation of candidate retrieval, Tab. 4.8 shows the obtained results in F_{BoT} performance. The figure left from the arrow is obtained using the unsupervised variant without candidate consolidation (cf. Tab. 4.5), the figure right from the arrow is obtained with candidate consolidation. The first observation is that candidate consolidation consistently improves entity linking performance. With an increase of about 10 points in percentage (pp) this is most notably on **MSNBC** and **ACE**. The effect is also observable on the other corpora, albeit with a lower average increase in performance of about 3 pp.

Apart from **IITB**, the increase in F_{BoT} is proportional to the number of uncovered entity mentions that can be resolved correctly using candidate consolidation (cf. Tab. 4.3). For **IITB** we have with about 15% a comparably high number of missing entities that need to be resolved to NIL*. However, the effect of candidate consolidation is with an increase of about 3 pp not as strong as expected. In contrast, for **MSNBC** and **ACE** we have about 15% of uncovered entity mentions and a strong increase in performance of about 10 pp. For **AQUAINT** we have a lower number of about 3% of uncovered entity mentions and also a lower increase in performance of about 3 pp.

Similarly, we have for **CoNLLb** no ground truth NIL and with about 1% very few missing entities. Notably, the increase in performance is the lowest for **CoNLLb**. Since Hoffart et al. [2011b] ignored uncovered entity mentions in their evaluation on **CoNLLb**, we do the same here for the sake of comparability. This deems the task of NIL detection through candidate consolidation somewhat useless and consequently the effect of candidate consolidation is not that strong. What's even more important is that the performance is here about 7 pp lower compared to the unsu-

pervised approach that uses relational information from collective search candidates. Admittedly, given the intentions of the authors, this corpus is especially useful to demonstrate the effect of relational or collective information.

Again, we find that name expansion ($\mathbf{S}_n$) is beneficial for all corpora apart from **AQUAINT** and that increasing search coverage also increases performance in general. Given that for **AQUAINT** we are able to correctly link most of the mentions using only named based attributes and that no additional information increases performance, we may argue that at least for this corpus our alias resource design is of highest influence and importance. On the other hand, note that contextual information is important for other corpora, especially **CoNLLb**.

So far we have empirically shown that the usage of a supervised classifier for candidate re-ranking and the detection of uncovered entities is able to increase performance in general across the different corpora. But the most important finding is that the performance of an entity linking model strongly depends on the corpus at hand and the given mentions to be linked. While specific attributes may increase performance notably on one corpus, their influence can be marginal or even misleading on another corpus. Now, given that we can not determine the best model across the different corpora so far, we have evaluated our model with relational information, i.e. the prioritization on collective search candidates, in all of the above configurations. Having determined name expansion ($\mathbf{S}_n$) to be helpful in general, we use it in all of the following experiments.

For the sake of clarity, we will here focus on the best configurations considering the search coverage of our system and give the detailed results in Tables B.1 to B.5 in Appendix B. Figs. 4.4 to 4.8 therefore show the best coverage configuration for our system in combination with prioritization on collective search candidates using different cross coherence weights. In addition, we also provide results that are obtained with the baseline $\mathbf{NAME}_c$ in these figures. This baseline corresponds to the first column in Tab. 4.8 and uses only the mention name in its original form for candidate retrieval. For candidate consolidation, the baseline $\mathbf{NAME}_c$ uses only the variants of the index score (Eqs. 4.17 to 4.19), the dedicated NIL feature (Eq. 3.19) and EMP (Eq. 2.7) as features for the Ranking SVM.

Figs. 4.4 to 4.8 also show results obtained by competitor methods. First of and in line with the findings of Hachey et al. [2013], we emphasise that re-implementations of entity linking systems towards Wikipedia are generally difficult to evaluate. This is because of published results being unfortunately not always reproducible. For instance, Hachey et al. report an accuracy of 88.3% on **MSNBC** for their implementation of Cucerzan's system, whereas Cucerzan originally reported an accuracy of 91.1% for their method on this corpus. Even though this difference is not striking, it is noteworthy as the implied error reduction differs notably.

Now, considering that most approaches use different versions of Wikipedia, such differences may be partially due to changes in Wikipedia. On the other hand, they may also be due to variations in pre- and post-processing that lead for instance

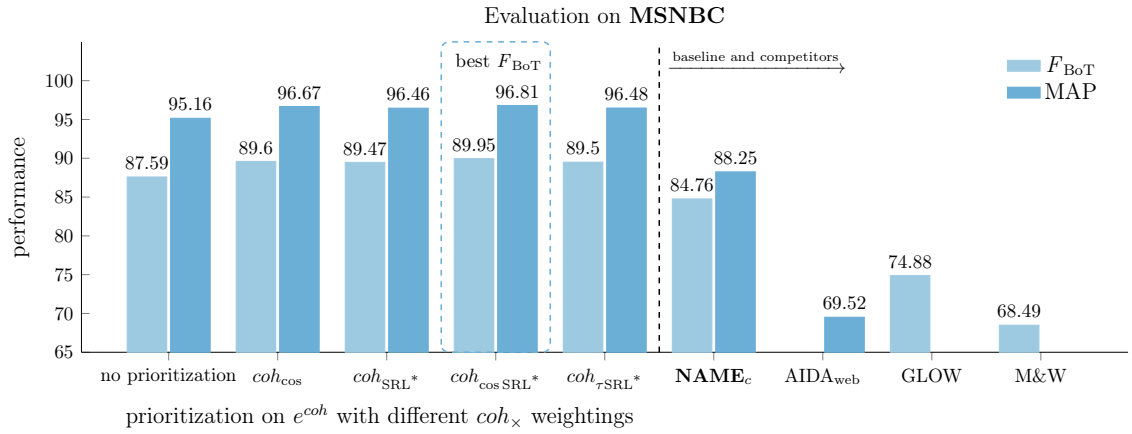


Figure 4.4: Comparison of our system with competitor methods AIDA_{web}, GLOW and M&W on **MSNBC** in F_{BoT} and MAP performance (all values in %). The best configuration using full search coverage $\mathbf{S}_{ntc}$ and $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ for candidate consolidation achieves an F_{BoT} of 89.95% with corresponding MAP of 96.81% and F_{BoT^*} of 91.26%.

to different feature sets. Hence, we argue that it is practically challenging if not infeasible to re-implement every competitor system in exact the same variant as published by the authors. Thus, for a fair comparison, we refer here either to the published results of the different works or use implementations kindly provided by the authors.

Unfortunately, even though the GLOW implementation is publicly available, we decided against using it. We could not reproduce the results published in Ratinov et al. [2011], even though we discussed the arising issues in detail with the authors[1]. Hence, we use the figures as reported by Ratinov et al. [2011] both for GLOW as well as for the approach of Milne and Witten [2008b] (denoted by M&W in the following). Also for comparison with Kulkarni et al. [2009] and Han et al. [2011], we use the figures as published in the respective paper. For comparison with AIDA (Hoffart et al. [2011b]), we use the online interface AIDA_{web} which was kindly provided to us by the authors[2]. As this implementation gives results very close to the published ones and since AIDA_{web} also handles uncovered entity mentions, we assume that we can fairly compare with AIDA_{web} on all corpora.

Since the interpretation of model performance is difficult across different performance measures, we give the performance for the best configuration of our model in F_{BoT}, F_{BoT^*} and MAP, the measures used by the related approaches (we described these measures in detail in Section 4.2). As the MAP measure assumes a confidence score to order predictions, we use the $rank(x_e)$ (Eq. 3.17) predicted by the Ranking

[1]Many thanks to Lev-Arie Ratinov for his helpful assistance.
[2]We use the version of July 30th, 2012.

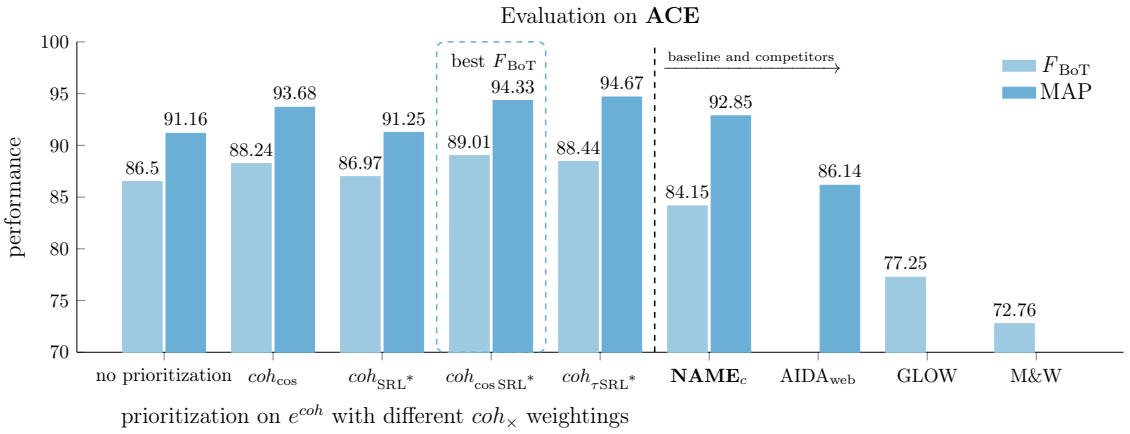


Figure 4.5: Comparison of our system with competitor methods AIDA_{web}, GLOW and M&W on **ACE** in F_{BoT} and MAP performance (all values in %). The best configuration using full search coverage $\mathbf{S}_{ntc}$ and $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ for candidate consolidation achieves an F_{BoT} of 89.01% with corresponding MAP of 94.33% and F_{BoT^*} of 85.55%.

SVM to evaluate our method in MAP.

Now, for **MSNBC** (Fig. 4.4), the best performing configuration uses full search coverage $\mathbf{S}_{ntc}$, prioritization on collective search candidate e^{coh} with $coh_{\text{cos SRL}^*}$ as cross coherence weighting (Eq. 4.12) and the topic distribution derived Hellinger distance $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ (Eq. 4.21) as additional feature in candidate consolidation. With this configuration, we obtain an F_{BoT} of 89.95% with associated values of 96.81% in MAP and 91.26% in F_{BoT^*}. The F_{BoT} performance of our system is 15 pp higher than that of GLOW (74.88% in F_{BoT}) and 20 pp higher than that of M&W (68.49% in F_{BoT}). Also, the respective MAP value of our system is with 96.81% more than 25 pp higher than that of AIDA_{web} that achieves a MAP of only 69.52%. This means that our approach achieves an error reduction of about 60% compared to GLOW, 68% compared to M&W and 89% compared to AIDA_{web}.

While the difference among cross coherence weight factors is not noteworthy, the prioritization on collective search candidate e^{coh} gives in general better results. We find that this prioritization increases performance about 2 pp compared to the variant using no prioritization. The baseline $\mathbf{NAME}_c$ also gives satisfactory results and beats all competitors with an F_{BoT} of 84.76%, even if this performance is about 5 pp lower than that of the best configuration of our system.

We found that the same configuration as on **MSNBC** also yields the best result on **ACE** (Fig. 4.5). On this corpus, our system achieves an F_{BoT} of 89.01%, which outperforms GLOW (77.25% in F_{BoT}) and M&W (72.67% in F_{BoT}) by more than 12 pp. Also, the MAP of our system is with 94.33% about 9 pp higher than the MAP of 86.14% obtained by AIDA_{web}. Again, our approach achieves a high error

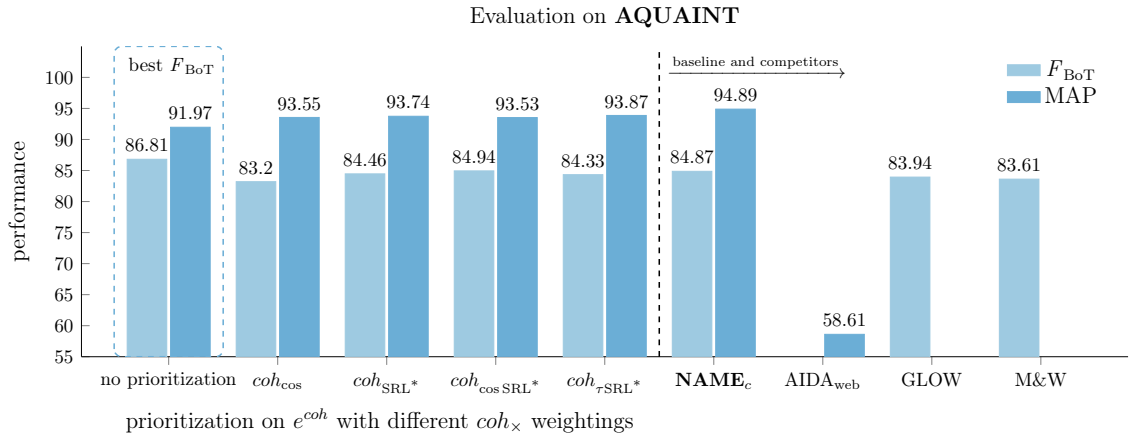


Figure 4.6: Comparison of our system with competitor methods AIDA_{web}, GLOW and M&W on **AQUAINT** in F_{BoT} and MAP performance (all values in %). The best configuration using full search coverage $\mathbf{S}_{ntc}$ and $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ for candidate consolidation achieves an F_{BoT} of 86.81% with corresponding MAP of 91.97% and F_{BoT^*} of 82.56%.

reduction of about 51% compared to GLOW, 60% compared to M&W and 58% compared to AIDA_{web}.

Similar to **MSNBC**, the difference among cross coherence weight factors is not striking, but the prioritization on collective search candidate e^{coh} gives about 2 pp higher results compared to the baseline that uses no prioritization and obtains an F_{BoT} of 86.5%. Again, the baseline $\mathbf{NAME}_c$ performs with an F_{BoT} of 84.15% about 5 pp worse than the best configuration but still beats all competitors.

For **AQUAINT** (Fig. 4.6), the best configuration of our system has full search coverage $\mathbf{S}_{ntc}$ and uses the topic feature (Eq. 4.21) for candidate consolidation. Without the usage of collective information, our system achieves an F_{BoT} of 86.81% which outperforms GLOW (83.94% in F_{BoT}) and M&W (83.61% in F_{BoT}) by 3 pp. Also, the MAP of our system is with 91.97% about 30 pp higher than the MAP of 58.61% achieved by AIDA_{web}. Note that the figure for M&W is here taken from the results reported by Ratinov et al. [2011], whereas Milne and Witten [2008b] reported an accuracy of 76.4% on **AQUAINT**. As not otherwise stated, we assume that Ratinov et al. used the API[1] instead of a re-implementation of Milne and Witten's method. Then, this difference may be due to the way performance measures are calculated or to differences in the API model implementation.

Even though the difference in performance is not striking, note that our method reduces the error by 18% compared to GLOW and 19% compared to M&W. Comparing to AIDA_{web}, the difference in performance is more obvious and we achieve an error reduction of 80%.

[1] `http://wikipedia-miner.cms.waikato.ac.nz/services/?wikify`

For **AQUAINT**, the prioritization on e^{coh} candidates did not increase performance in either cross coherence weight. We argue that this is due to the rather low average cross coherence over the ground truth entities. However, even if we use prioritization on e^{coh} candidates, the obtained results are higher than that obtained by the competitor methods. The exception is the $coh_{\cos}$ weighting (Eq. 4.13), but then the obtained performance is only less than 1 point in percentage lower than that of GLOW, the best performing competitor. Especially for **AQUAINT** we observe that the baseline $\mathbf{NAME}_c$ is hard to beat: with an F_{BoT} of 84.87% it performs not only better than the competitor methods but also close to the best configuration.

To summarize, the differences among the compared methods are not striking on **AQUAINT** and only the low performance of AIDA_{web} stands out. We can think of two reasons for that. First, the low performance of AIDA_{web} can be related to the low average cross coherence over the ground truth entities. Also, the EMP baseline used in AIDA_{web} may be misleading. Recall the example document on space crafts (Example 15): we found the EMP of Space Shuttle Columbia for the mention Columbia to be rather low, i.e. only 5%. When this is the case for a substantial amount of mentions, the EMP baseline is prone to perform very poorly.

Furthermore, Ratinov et al. [2011] reported that the SVM used for candidate consolidation did not improve accuracy consistently on all datasets. The gains were found to be marginal and for **AQUAINT** the accuracy was even decreased. Ratinov et al. assume that this is because the model is trained on Wikipedia references, but tested on non-Wikipedia text which has different characteristics. This may be a valid point and given that our model is also trained on the **CoNLL train** news articles, we have to admit that the results might be even more convincing if we had also trained our model on Wikipedia references.

However, we strongly rely on the training data to learn the threshold for NIL prediction and assume that the **CoNLL train** corpus may be more suitable than the strategy we pursued in Chapter 3, where we needed to simulate uncovered entity mentions in Wikipedia references. In contrast, Ratinov et al. [2011] did not thoroughly model NIL candidates in their approach. There is no threshold or dedicated feature from which a threshold could be learned. The only feature in that direction is a Good-Turing estimate of how likely a mention is to be a NIL entity, based on the counts in the entity-mention probability model. Since this is computed over Wikipedia data, this may not be a very reliable feature.

For **CoNLLb** we compare not only to the results obtained with AIDA_{web} but also to the results published for other configurations of AIDA. These are $\text{AIDA}_{\text{r+}}$ and $\text{AIDA}_{\text{r-}}$. $\text{AIDA}_{\text{r+}}$ is the variant using robustness tests that was reported to achieve highest precision. $\text{AIDA}_{\text{r-}}$ is the variant that uses no robustness test but achieves the highest MAP. Interestingly, these results are not symmetric as the MAP of $\text{AIDA}_{\text{r+}}$ is reported lower than that of $\text{AIDA}_{\text{r-}}$ (about 2 pp), while the precision of $\text{AIDA}_{\text{r+}}$ is higher than that of $\text{AIDA}_{\text{r-}}$ (about 1 pp).

As depicted in Fig. 4.7, the best configuration of our system for **CoNLLb** uses full

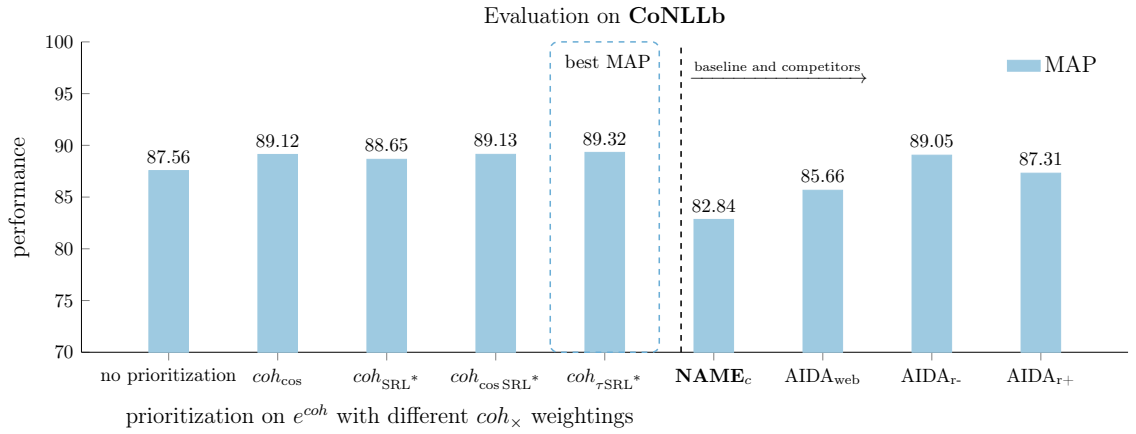


Figure 4.7: Comparison of our system with competitor methods AIDA_{web}, $\text{AIDA}_{\text{r+}}$ and $\text{AIDA}_{\text{r-}}$ on **CoNLLb** in MAP performance (all values in %). The value for $\text{AIDA}_{\text{r+}}$ indicates the performance of AIDA with robustness test, the value for $\text{AIDA}_{\text{r-}}$ was the best reported MAP without robustness test. The best configuration using full search coverage $\mathbf{S}_{ntc}$ and $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ for candidate consolidation achieves a MAP of 89.32% with corresponding F_{BoT} of 82.16% and F_{BoT^*} of 78.86%.

search coverage $\mathbf{S}_{ntc}$, the topic feature in candidate consolidation, and prioritization on e^{coh} candidates with $coh_{\tau\text{SRL}^*}$ weighting (Eq. 4.15). This configuration achieves a MAP of 89.32% with corresponding values of 82.16% in F_{BoT} and 78.86% in F_{BoT^*}. This value is only slightly better than the figures published for $\text{AIDA}_{\text{r-}}$ (89.05% in MAP) but with an increase of 2 pp already more notably better than the value of 87.31% in MAP published for $\text{AIDA}_{\text{r+}}$. It is even about 4 pp higher than the MAP of 85.66% we obtained with AIDA_{web}. Even though the absolute performance of the compared approaches is very close, we emphasise that the error reduction of our approach is 25% compared to AIDA_{web}, 15% compared to $\text{AIDA}_{\text{r+}}$ and still 2.5% compared to $\text{AIDA}_{\text{r-}}$.

Also, we find that all other configurations of our system using different cross coherence weights perform better than AIDA_{web} and $\text{AIDA}_{\text{r+}}$. Comparing to $\text{AIDA}_{\text{r-}}$, we find better performance only when using prioritization on collective search candidates e^{coh}. An exception is the baseline weight coh_{SRL^*}, however the difference is negligible as performance is less than a half point in percentage lower.

For **CoNLLb**, the baseline $\mathbf{NAME}_c$ performs with a MAP of 82.84% about 6 pp worse than the best configuration. This goes in line with the experiments on search coverage where we found that contextual information is very important on this corpus (cf. Tab. 4.8)).

In an error analysis, we found that the performance of our system is negatively affected by differences in the annotation schemes, especially for **CoNLLb**. While our system links mentions like British to entities such as ENGLISH LANGUAGE or

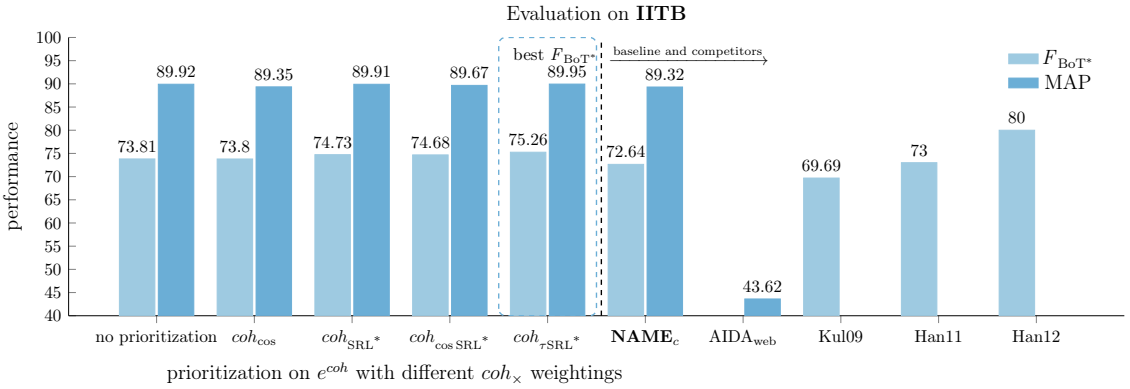


Figure 4.8: Comparison of our system with competitor methods AIDA$_{\text{web}}$, Kul09 (Kulkarni et al. [2009]), Han11 (Han et al. [2011]) and Han12 (Han and Sun [2012]) on **IITB** in F_{BoT^*} and MAP performance (all values in %). The best configuration using only name coverage $\mathbf{S}_n$ achieves an F_{BoT^*} of 75.26% with corresponding MAP of 89.95% and F_{BoT} of 80.41%.

British people depending on the context, the annotators of **CoNLLb** always assigned such mentions to United Kingdom. Even though the assignment to this ground truth entity may be correct in many cases, one can argue whether it is *always* correct. We postulate that this is not the case but unfortunately are dependent on the gusto of the annotators.

Hoffart et al. [2011b] also reported results for re-implementations of Cucerzan [2007] and Kulkarni et al. [2009]. While the implementation of Kulkarni et al. [2009] achieved with a MAP of 86.50% a result close to the values obtained for the variants of AIDA, the implementation of Cucerzan [2007] performed poorly and achieved a MAP of only 40.06%. This result is surprising, since Cucerzan's approach was found to be effective on various other corpora, specifically including the TAC challenges (Hachey et al. [2013]). Again, we note that results obtained with re-implementations of complex entity linking models should be judged carefully.

For **IITB** (Fig. 4.8), the best result is obtained using only name coverage ($\mathbf{S}_n$) and prioritization on e^{coh} candidates with $coh_{\tau\text{SRL}^*}$ weighting (Eq. 4.15). This configuration achieves an F_{BoT^*} of 75.26% with associated 89.95% in MAP and 80.41% in F_{BoT}, which is 5 pp higher than the F_{BoT^*} of 69.69% reported by Kulkarni et al. [2009]. Also note that the performance of AIDA$_{\text{web}}$ on **IITB** is with a MAP of 43.62% very low, whereas the corresponding MAP of our system is 89.95%. Consequently, our approach yields a noteworthy error reduction of 18% compared to Kulkarni et al.'s method and 82% compared to AIDA$_{\text{web}}$.

Although we found for **IITB** the lowest average cross coherence over ground truth entities among all benchmark corpora, the prioritization on collective search candidates can reduce the error by about 5.5%. However, this is only the case for the

SRL* based cross coherence weights, i.e. coh_{SRL^*} (Eq. 4.11), $coh_{\cos \text{SRL}^*}$ (Eq. 4.12) and $coh_{\tau\text{SRL}^*}$ (Eq. 4.15). Using the purely context based weight $coh_{\cos}$ (Eq. 4.13), we obtain performance close to the variant using no prioritization. We assume that this result is due to the very high number of mentions per document but also to the nature of the documents. One the one hand, the high number of mentions has a diminishing effect on the average cross coherence. On the other hand, given that the documents are web pages and not editorial news stories, this may also imply thematically diverse contexts where our representation of mention contexts and the inferred contextual similarity towards candidate entities may not be appropriate.

Also, as detailed in Tab. B.5, adding additional information or the topic feature decreases performance on this corpus by up to 6 pp. This corresponds to the results in Tab. 4.8, where we found that additional context information also slightly decreased results. Unfortunately, there is no direct explanation for this behaviour. Seemingly, the surface form information of mentions is the most important feature for this corpus. This is also reflected by the comparably high value of 72.64% in F_{BoT^*} obtained with the baseline $\mathbf{NAME}_c$ which is only less than 3 pp lower compared to the variant using prioritization on collective search candidates.

Comparing to the collective approaches of Han et al. [2011] and Han and Sun [2012], we find that our method performs better than Han et al. [2011] who reported an F_{BoT^*} of 73%. In contrast, the F_{BoT^*} of 80% reported in Han and Sun [2012] is about 5 pp higher than our best configuration. However, we should point out that both approaches ignore NIL entities in their model design. Also, the two methods are evaluated only on a small variety of datasets, namely **IITB** and the TAC 2009 dataset that we discussed in Section 3.7. Comparing the two earlier approaches, i.e. Han and Sun [2011] and Han et al. [2011], Han and Sun [2012] reported comparable performance for all methods with accuracy values of 85.4% (Han and Sun [2012]), 86% (Han and Sun [2011]) and 83.8% (Han et al. [2011]).

Concerning efficiency, we should note that Han et al. [2011] proposed a graph based method that needs to update the node-edges or even construct the full reference graph for each input document and each mention to link. Here, both of our proposed indices need to be created only once and do not require additional computational updates depending on input documents or mentions to link.

In an error analysis for **IITB** we found that our approach is negatively affected by Kulkarni et al.'s tendency of grounding mentions to disambiguation pages. This affects 129 mentions and makes up for about 10% of missing ground truth targets NIL* since disambiguation pages are not contained in our index $\boldsymbol{\mathcal{I}}_{\boldsymbol{\mathcal{W}}}$. For example, we observed a document with a sports subject that mentions the word fitness. This mention was linked to the disambiguation page Fitness by the **IITB** annotators. Our system predicted the suitable entity Physical Fitness, but unfortunately we were bound to treat this as an erroneous prediction since we had to re-target the disambiguation page Fitness to NIL* being that disambiguation pages are intentionally excluded from our index.

We may now answer the questions we asked at the beginning of this section. First, unfortunately the answer to **Question 1** is *No*, we did not find a configuration that consistently outperforms all of the eight competitor methods. While the full search coverage $\mathbf{SC}_{ntc}$ in combination with the topic feature (Eq. 4.21) gives the best result on **MSNBC**, **ACE**, **AQUAINT**, and **CoNLLb**, this is not the case for **IITB**. For **IITB**, the name coverage $\mathbf{SC}_{n}$ is most effective. Further, while our approach yields better results than the competitors GLOW, M&W and AIDA_{web} on **MSNBC**, **ACE**, **CoNLLb**, and **IITB** independently of the configuration, this is not the case for **AQUAINT**. On **AQUAINT**, the name baseline $\mathbf{NAME}_{c}$ was found to be very effective and this corpus was the only one where we found prioritization on collective search candidate e^{coh} to reduce performance.

This gives us also the answer to **Question 2**: apart from **AQUAINT**, the prioritization on collective search candidates e^{coh} has a positive effect on all corpora. For **MSNBC**, **ACE**, **CoNLLb**, and **IITB** performance is consistently increased by about 2 pp and achieves an average error reduction of 11.7% compared to the configuration using no prioritization. Also, again apart from **AQUAINT**, the cross coherence weights combining contextual and semantic information, i.e. $coh_{\cos \text{SRL}^*}$ (Eq. 4.12) and $coh_{\tau\text{SRL}^*}$ (Eq. 4.15), yield slightly superior results on all corpora compared to the context free version coh_{SRL^*} (Eq. 4.11) or the version $coh_{\cos}$ (Eq. 4.13) that neglects semantic similarity. This answers **Question 3** and shows that, at least for the corpora at hand, there is no single cross coherence weight that gives the best result across all corpora.

This leads us to the question whether we can effectively determine a threshold over candidate entities, for instance comparable to the coherence test used in AIDA (Hoffart et al. [2011b]). Such a test allows us to automatically enable or disable the prioritization on collective search candidates. Hoffart et al. [2011b] learned the threshold for their test from development data and certainly this could be an interesting avenue for further research. However, given the diversity of results we observed in this experimental evaluation, we can not assume that a learned threshold will generalize to all potential use cases and apply for all application corpora.

Further, Hoffart et al. [2011b] point out that coherence needs to be treated with care. First, collective disambiguation requires a certain number of entities to originate from a thematically related context. Second, as collective disambiguation aims at maximizing the coherence over candidates, it may also erroneously enforce that predicted entities fit into a single coherent set. Hoffart et al. give the example of a document about a football game between Manchester and Barcelona taking place in Madrid. Then, collective disambiguation may erroneously link all three of these mentions to football clubs, i.e. Manchester United, FC Barcelona, and Real Madrid. Here, we tried to account for this issue through the mixture of prioritization on collective search candidates e^{coh}, title&redirect baseline as well as the associated features in our Ranking SVM. Given that our collective approach was slightly inferior to the non-collective variant, i.e. the configuration without prioriti-

zation on e^{coh} candidates, only for one corpus (**AQUAINT**), we assume that this model is appropriate.

Still, we find that the prior-like entity-mention probability EMP (Eq. 2.7) is a very strong feature. While Hoffart et al. used a heuristic to activate this feature, we tried to learn an appropriate weight through the Ranking SVM. Given its importance, EMP is determined very influential by the Ranking SVM which is realized through a high feature weight. Still, this may be misleading. This was for instance observed on **CoNLLb**, a corpus containing many sport statistics that mention countries participating in a match. As an example, we observed that even though the e^{coh} candidate Japan National Football Team is correctly retrieved due to high cross coherence, the Ranking SVM erroneously re-ranked candidates by giving the highest score to the more popular entity Japan. This is because Japan has with 97% a much higher EMP for the mention Japan compared to the very low EMP of 0.63% of Japan National Football Team.

Given that name baselines appear to be very competitive, we should lastly answer **Question 4**. Comparing to the unsupervised baseline using only name information (Tab. 4.5), we achieve an average error reduction of 49.6% across all configurations and corpora. The average error reduction is with 10.5% the lowest on **IITB**[1] and with 68.5% the highest on **CoNLLb**. For **AQUAINT**, this is with 16.3% also expectedly lower, for **MSNBC** it is 53.2% and for **ACE** 49.8%.

Comparing to the supervised name baseline $\mathbf{NAME}_c$, the average error reduction is with about 24% expectedly lower but also very remarkable. Accordingly, the average error reduction is with 7.2% the lowest on **IITB** and with 36.22% the highest on **CoNLLb**. For **MSNBC** and **ACE** this is with 32% resp. 25% also notable. As collective search did not increase performance on **AQUAINT** in the supervised setting, we unfortunately also find no error reduction compared to $\mathbf{NAME}_c$.

To summarize, the diversity of the corpora renders the formulation of a linking model with consistent performance challenging. Depending on the corpus nature and the entities to be linked, different configurations can be more suitable. This goes along with the observations pronounced in Ratinov et al. [2011]. Ratinov et al. found that their variant of global, link based features are not always helpful and that especially in the candidate consolidation may be negatively influenced by domain changes.

Also, some corpora such as **CoNLLb**, may be more suitable for the evaluation of graph based methods that rely strongly on relational information. Other corpora like **AQUAINT** may be more suitable for the evaluation of less complex feature based methods. Note that even though our system was not tuned on either dataset, we achieve a high performance on all of the five different benchmark corpora. We argue that this makes our system the most stable compared to other approaches both in terms of generalizability and applicability.

[1]The F_{BoT^*} corresponding to the F_{BoT} performance of 76.13% in Tab. 4.5 is 71.63%.

We are aware that the empirical evaluation described in this section is not supported by significance tests. This is an inconvenience but unfortunately an inevitable one due to the nature of the benchmark corpora. Each corpus consists only of a single test set so that we can not perform a cross-validation to measure the variance in performance. Also, given that we have only the figures as reported in Ratinov et al. [2011], Hoffart et al. [2011b], Kulkarni et al. [2009], Han et al. [2011] and Han and Sun [2012], and not the predictions per instance, we can not compare the error rates of the methods, for instance using a McNemar-test (Dietterich [1998]). This would have been presumably most beneficial for comparison with GLOW (Ratinov et al. [2011]) on the **AQUAINT** corpus, given that on this corpus we observed the lowest difference in performance for our method and GLOW. However, we argue that we still have demonstrated the stability of our method through a detailed empirical evaluation on several benchmark corpora where we always obtained comparable, and in the majority of cases also superior results.

Comparison with Thematic Context Distance

We also evaluated the method described in Pilz and Paaß [2011] (Chapter 3) on these datasets. Unfortunately, the obtained results were not satisfactory. We assume two reasons for that. When applying the thematic distance method from the previous chapter, we consequently also used the respective candidate retrieval method. Now, since this candidate retrieval method relies on (partial) matches of mentions against Wikipedia titles, it is more restricted than the method we proposed in this chapter and uses far less resources, i.e. it uses no information from link anchor texts or redirects. Thus, a considerable portion of candidates could not be retrieved, for instance all candidates for mentions that are referenced using an acronym. The second reason we assume is that the corpora tackled in this chapter contain many different entities apart from persons. For entities such as locations or organizations, the thematic overlap (as measured by the topic distribution) between mention and entity context was found to be low. Thus, again for a significant number of mentions, the assumption of close thematic overlap between mention and candidate context, as made by the method based on thematic distances, was not fulfilled.

4.9 Summary

In this chapter we have proposed a novel entity linking model that reliably detects if an entity mention is covered in Wikipedia and then very accurately assigns this mention to its unique representative in Wikipedia. The proposed model treats covered as well as uncovered entity mentions of various types and makes no restrictive assumptions on the nature of referencing contexts.

In contrast to other approaches, our method is not tuned for one specific corpus.

Therefore we evaluated various configurations of our approach on five benchmark corpora and compared the results to five competitor systems. This was challenging not only due to diverse natures of the corpora but also to remarkable differences in the annotation schemes. We also analysed different evaluation criteria proposed by related work and discussed their relative strengths and weaknesses. Evaluating our method in all of these performance measures, we argue that we presented an evaluation that is more comparable than previous results.

While evaluated only on English documents, we postulate that the proposed method also applies to other languages given that the required resources are available in the respective version of Wikipedia. We have also shown that the careful design of alias resources and candidate retrieval results in satisfactory linking performance even when no supervised candidate consolidation model is used. We have shown that supervised candidate consolidation can further increase linking performance and we argue that it is a necessity for the reliable detection of uncovered entities.

For some benchmark corpora our system performed dramatically better compared to other approaches, while for other corpora the differences are not so pronounced. Except for one case, our system always has better performance figures than the competitor systems. It turned out that the effect of collective search on linking performance is more prominent when the average coherence among candidate entities is higher. Also, in one case, we found that using only the surface forms of mentions for linking was most effective. This result leads us to the insight that entity linking has a subjective nature and that the performance of a model may strongly depend on the corpus at hand. Nevertheless, we have empirically shown that our model is ready for practical application given its stable performance across different domains as demonstrated for various benchmark corpora.

Chapter 5

Conclusion

This thesis proposed different methods to link entity mentions in natural language text to unique entity representatives in Wikipedia. Since not all-real world entities are represented in Wikipedia, we distinguished among mentions of covered, linkable entities and uncovered, not linkable entities absent from Wikipedia. Depending on the task and corpus at hand, specific methods are most successful. We have introduced a thematic distance measure computed over the contexts of entity mentions and candidate entities in Wikipedia, a method that is naturally able to handle the usage of synonyms and found to be especially suitable to link references of named entities such as persons. Due to the thematic coherence observable between person references and person descriptions in Wikipedia, latent topic distributions are strong indicators of the true underlying entity of these references. While in the first part of this thesis we exploited the unstructured textual information in Wikipedia, we relied on the structured information encoded in its hyperlink graph in the second part. We introduced collective search over Wikipedia's hyperlink graph in order to collectively link mentions of more general entities, including not only named entities but also proper nouns. When contextual clues are sparse and latent topics do not emerge directly, finding the maximum joint occurrence of entity mentions in this graph gives reliable clues towards underlying entities.

We have proposed methods to relate unstructured text documents with the semi-structured knowledge base Wikipedia which opens up a multitude of applications, a step towards facilitating many information retrieval and information extraction tasks. Still, most information is stored in unstructured text documents such as newspaper articles and both the human as well as the automated extraction of knowledge from these texts is non trivial. From the human point of view, enriching text documents with encyclopedic knowledge allows for a better text understanding. This comprises explanatory links resolving technical terms or cross-referencing documents in educational contexts with encyclopaedic knowledge, probably the most obvious use cases. Furthermore entity linking enables entity-based retrieval, which is superior to retrieval based on naive string matching that can not resolve polysemy and synonymy. Entity-based retrieval in semantic search spares a human the hard time to sift through sources retrieved due to namesakes and helps to re-

trieve material focused on or relevant to a particular entity of interest. This is a step towards making the vast amount of unstructured knowledge stored in information distribution media more manageable. On the other hand, mutually dependent tasks such as named entity recognition and relation extraction are likely to gain from the knowledge of underlying entities. Entity linking may provide the type of an entity mention from Wikipedia categories but also list potential relations from links. If a relation extraction model consumes the prediction of a linking model that states that a mention refers to an organization instead of a person, specific relation types such as *bornIn* can be excluded. This holds for example for Axel Springer, a mention that may refer to a person or an organizational entity (a publisher). Simultaneously, a named entity recognition model may also be enriched with such a prediction. Qualified relations can be extracted from infoboxes or, at least for English, more conveniently from YAGO or DBpedia, if these Wikipedia derivatives contain entries of the respective entity. Entity linking is therefore an important step towards knowledge extraction on a global level, relating singular sources with others in the Linked-Open data cloud with Wikipedia as a hub.

Throughout this thesis, we gave equal attention to popular entities and thus easily linkable mentions, to less popular entities, where few information is available, but also to entities that are not linkable. Popular entities usually have high quality descriptions in Wikipedia that provide many details. Linking mentions of less popular entities is more challenging as their descriptions are usually short and contain only few keyterms that may not appear in referencing contexts, or may have been replaced by synonyms.

In some cases, entity linking boils down to assigning a mention to its most popular candidate entity, especially in editorial texts of nationwide news papers where entities are mentioned with often canonical names that directly match the title or name of the corresponding entity in Wikipedia. In contrast, for local news articles, this strategy will presumably result in false positive assignments when the underlying entity is in fact not covered in the knowledge base, either Wikipedia or Googles Knowledge Graph. Furthermore, the recall of such methods will be low when confronted with text documents from a domain where nicknames, abbreviations or spelling mistakes are common. High popularity entities are also the major focus of Googles disambiguation that was launched in May 2012 and is also based on an inverted index. Recently Google started to show entity profiles from Wikipedia. For popular entities these are aligned with the search results: clicking on one specific entity alters the set of retrieved pages and enables entity-based retrieval. However, this applies only to popular entities: searching for Michael Jordan produces no entity disambiguation. If the entity of interest is the Berkeley professor and not the basketball player one needs to alter the search string to Michael I. Jordan. Other proper nouns are also not yet thoroughly handled, since Google does not distinguish among fruits and companies (apple) or animals and cars (jaguar).

5.1 Lessons Learned

One of the main challenges encountered during this thesis was the often subjective interpretation of the task by other researchers. Subjectiveness is often encountered in natural language processing (NLP) tasks where the interpretation of natural language depends on the reader. The most prominent example is keyphrase extraction, where competing systems rarely achieve scores higher than 40% and inter-annotator agreement is only slightly higher (Hasan and Ng [2014]). While some tasks in NLP have come to a consistent treatment due to a long tradition this is not really the case for entity linking.

As we have shown in the last chapter of this thesis, there are various differences regarding both the targeting of hyperlinks as well as the performance evaluation of the models predicting these links. We find model design and evaluation techniques to depend on the selection of mentions to be considered, but also on the nature of predictions. Does an approach exclude specific entities and therefore render the task easier than it is? Or does an approach aim at an aggressive linkage towards Wikipedia, producing links for things that are not interesting to the reader? Can a disambiguation page that does not provide identity but hints at possible candidates be considered as a solution? Should a link denote identity or just be helpful for the reader by providing related information? Should a province be linked to the state it belongs to or the article describing this province?

Certainly, the answers to these questions depend on use cases and the intention of the authors. Different authors gave different answers to these questions and therefore we miss a consistent interpretability across current entity linking systems. The absence of a well-defined goal therefore makes entity linking a subjective task. But, based on the results of this thesis, we argue that we proposed methods that do not only satisfy the majority of possible interpretations but at the same time also outperform most other proposed models, them being either restricted or aggressive in their linking goal. Surely, the TAC series (McNamee and Dang [2009] and successors) aims at laying the ground for the consistent comparison of various systems, regarding both annotation guidelines as well as performance. We therefore gave attention to the obtained results in the presentation of related work. However, also due to personal concerns regarding the implied applications of this challenge, we could not and did not participate in these challenges.

To summarize, providing *the overall best* entity linking is difficult due to the just stated reasons. Restrictions to entity types are tractable, but differences in the targeting render systems less comparable. Providing one entity linking model that achieves superior performance across different tasks and corpora thus remains a major challenge. This thesis made two major steps towards this. First, we introduced the first contextual linking model with multilingual applicability that achieves excellent results in various languages without language specific model adaptions. Second, we presented a collective linking method exploiting all entity mentions in a docu-

ment that, due to its generality, showed superior performance not only across various corpora but also in different evaluation schemes. This is an important asset that notably increases the comparability of our method over that of other approaches. However, there is still room for improvement and we will describe below some possible avenues to pursue.

5.2 Outlook

There are more avenues to explore to further increase candidate recall. One promising resource emerges from citations. Citations are links from a Wikipedia article to references in external resources such as news paper articles or individual web pages. These sources could be used as example references for training and evaluation. Since such a distantly labelled dataset is not guaranteed to truly mention an entity of interest, it could at least be used to generate new contextual or relational features. However, this may require a certain amount of human interactions and adaptions to crawl and extract the textual content of these sources.

Some approaches also used Wikipedia's infoboxes and tables for their entity linking models. Even though these are certainly valuable resources, we neglected them because the existence and quality of infoboxes is not the same for all articles and all languages. Since their correct extraction can be cumbersome due to markup language and template variations, which also applies to disambiguation pages, we relied on YAGO, a research system built with major focus on this task. However, we did not so much rely on it as AIDA (Hoffart et al. [2011b]). A promising avenue would thus be the combination of our systems with AIDA, for example by defining local agreement over thematic instead of cosine similarity.

Recent research in deep learning for NLP proposed new continuous word representations of via *word vectors* (Mikolov et al. [2013]) or the more context sensitive variant of *paragraph vectors* (Le and Mikolov [2014]). These methods provide semantic relations computed over the co-occurrence of terms in large data sets, either on document or paragraph level. Exemplary pairwise relations emerging from these methods are *king-queen – man-woman* or *Paris-France – Berlin-Germany*. Deep learning networks *learn* features on different levels of abstraction more or less merely from the amount of provided data. Learning one such vector per entity is comparable to learning entity profiles but without the need to manually specify descriptive features in these profiles. Augmenting each mention, or even each term in a mention's context with its word vector is a new avenue for joint disambiguation that may not need to solve the NP-hard global optimization problem. For future work, we also note that the continuous word representations may be an alternative to LDA worth investigating.

Generally speaking, truly joint or collective disambiguation is probably the most promising avenue to pursue, since such methods are close to a human's understand-

ing of text. A human can use both factual and relational world knowledge to understand (or at least guess) the intention of a writer. The reasoning through connection of facts enables a human to understand and interpret natural language text, including the decision on appropriate senses and underlying entities.

As Mendes et al. [2011] pointed out, interactions with users are promising. If a user states that only persons in a document should be linked, then our thematic context distance will often be successful and is able to retrieve correct links in several languages. However, if the context is sparse, as can be the case for short news feeds, there is no guarantee that this context provides sufficient information to infer a reliable topic distribution. This can be tackled by predefined confidence thresholds. Each of our proposed methods can be extended with a confidence knob. If only high confidence predictions should be presented to a user or a consuming method, then we may return only those predictions that exceed a specified confidence threshold computed for example either from hyperplane offsets for the SVM based methods or the difference in the top two ranked candidates for the Ranking SVM based methods.

5.3 Applications

There are various applications for entity linking, ranging from aggregated information retrieval for specific entities over various sources to automated reasoning over the extracted information to produce new knowledge and facts and facilitate knowledge base curation.

Entity Linking in Digital Archives

For illustration, Fig. 5.1 shows a document contained in a digital archive of German newspaper articles that was created in the context of the Contentus project[1]. In this project, the entity linking method published in Pilz and Paaß [2011] was used to enable the entity-based search in the document archive. While named entity recognition is used to mark the occurrences of named entities in the document, entity linking aligns these mentions with Wikipedia. This is shown in Fig. 5.1 for the mention Merkel that is linked to the article describing Angela Merkel in the German version of Wikipedia. Using the links provided by an entity linking model, existing articles, for example that on Angela Merkel, may be endowed with new facts that are automatically qualified by citation sources. Using the quote extraction approach presented in Paaß et al. [2012], we may also add quotes that carry reliable information about the opinions of a person.

In Paaß et al. [2009] we have proposed a named entity recognition model for audio transcripts, i.e. statistical translations of spoken language into text documents. Using such a model it would also be possible to create links for spoken named entity

[1] http://www.contentus-projekt.de

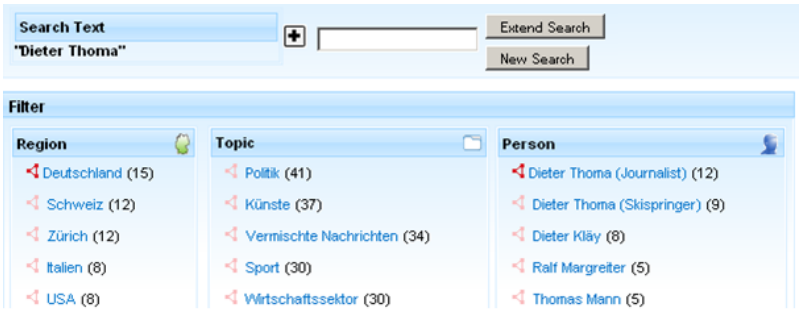
Figure 5.1: This screen shot taken from the Contentus system shows a news article from the German Jahreschronik that is enriched with named entity tags. The figure exemplifies the link for the mention Merkel which grounds the mention to the Wikipedia article of the German politician Angela Merkel.

mentions in videos, podcasts or broadcast news that are converted into electronic text documents. Then, a link could not only be placed in a textual reference but also anchored in a video stream so that a listener or a viewer can be presented with additional, perhaps even visual information.

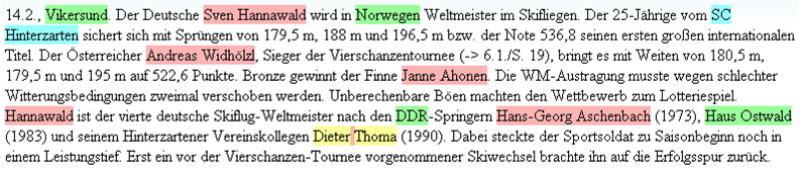
Entity Linking for Entity-Based Retrieval

Entity linking allows to aggregate the retrievable information about a specific entity into a more actionable set. It enables focused, entity-based retrieval, it is a key component of semantic search. As an example, the semantic search interface of Contentus (Fig. 5.2a) provides the facility to distinguish documents referring to Dieter Thoma (Skispringer) from documents referring to Dieter Thoma (Journalist) (Fig. 5.2b). Further, using the German Wikipedia for the disambiguation of person names, we can extract their Personennamendatei (PND)[1] identifiers that are provided in most Wikipedia articles describing persons. The PND is an entity catalogue provided by the Deutsche Nationalbibliothek and contains about 3.6 million persons with 1.8 million discriminated entries. In contrast to the encyclopedia Wikipedia, the PND provides only few discriminating pieces of information such as pseudonyms, affiliations and origin of birth together with an identifier, but no further description comparable to article texts in Wikipedia. However, having linked a mention to its appropriate article in Wikipedia, this PND identifier can be extracted and used to link the disambiguated mention to the database of the Deutsche National Bibliothek. This again provides further information, for example details about the book written by Dieter Thoma (Skispringer) as shown in Fig. 5.2c.

[1]Since April 2012, the PND (engl. *Name Authority File*) is integrated in the Gemeinsame Normdatei (engl. *Integrated Authority File*) (`http://www.dnb.de/gnd`)



(a) Entity-based search retrieves documents for disambiguated entity mentions, here Dieter Thoma.



14.2., Vikersund. Der Deutsche Sven Hannawald wird in Norwegen Weltmeister im Skifliegen. Der 25-Jährige vom SC Hinterzarten sichert sich mit Sprüngen von 179,5 m, 188 m und 196,5 m bzw. der Note 536,8 seinen ersten großen internationalen Titel. Der Österreicher Andreas Widhölzl, Sieger der Vierschanzentournee (-> 6.1./S. 19), bringt es mit Weiten von 180,5 m, 179,5 m und 195 m auf 522,6 Punkte. Bronze gewinnt der Finne Janne Ahonen. Die WM-Austragung musste wegen schlechter Witterungsbedingungen zweimal verschoben werden. Unberechenbare Böen machten den Wettbewerb zum Lotteriespiel. Hannawald ist der vierte deutsche Skiflug-Weltmeister nach den DDR-Springern Hans-Georg Aschenbach (1973), Haus Ostwald (1983) und seinem Hinterzartener Vereinskollegen Dieter Thoma (1990). Dabei steckte der Sportsoldat zu Saisonbeginn noch in einem Leistungstief. Erst ein vor der Vierschanzen-Tournee vorgenommener Skiwechsel brachte ihn auf die Erfolgsspur zurück.

(b) Document retrieved for Dieter Thoma (Skispringer) with named entity mentions coloured by type (persons in red, locations in green, organizations in blue). The mention of Dieter Thoma (Skispringer) is marked in yellow.

(c) Entry for Dieter Thoma (Skispringer) in the Deutsche National Bibliothek acquired from the PND identifier provided in Wikipedia.

Figure 5.2: These screen shots taken from the Contentus system illustrate that entity linking in unstructured text allows entity-based retrieval. The semantic search in Contentus groups results on the entity level (5.2a), retrieves documents for specific entities (5.2b) and provides additional information for them from other linked sources, such as the Deutsche National Bibliothek (5.2c).

Hence, entity linking to a Wikipedia enables not only the distinction of name mentions in unstructured text. It further allows to enrich the mention and its context with new information that may not be provided in the input document but can be extracted from Wikipedia. Thus, entity linking may provide structured or semi-structured knowledge about any unstructured document. This does not only apply to named entities but also to general concepts such TREE or GRAPH. For example, a computer science student might want to retrieve information about the graph concept of trees and not about the trees in a forest. He might also want to learn more about the algorithm named after the physicist Metropolis and not Superman's home town. There are many more examples of ambiguity of natural language text, since many entities share the same name and one entity may be referenced by various different names. Now, if the textual content of the retrieved pages in the search result is linked against Wikipedia, he or she may very easily acquire further information not only on the subject of interest but also on concepts related to it. Clearly, such an entity-based retrieval will allow for an accelerated information retrieval and also an enhanced text understanding.

Entity Linking for Opinion Mining

A more industry related application for entity linking is Opinion Mining. Most companies are interested in customer opinions on specific, often newly launched products. Customers, on the other hand, provide valuable information in form of online product reviews, posts in fora, blog entries or various other online platforms. In an aggregated form these statements can give a company valuable opinions on their latest product releases. One solution would be to manually determine platforms that are thematically related to the companies product range, so that the thematic constraint would reduce the number of false positive retrieved examples. On the other hand, one could use *all* possible web search results for the string representing the product name and in a second, automatically performed disambiguation step, determine which of the retrieved results refer to the product of interest. Fig. 5.3 shows contexts for three products named Prestige, a tennis racket, a beer brand and a movie. Having evaluated an entity linking on such contexts, we may present the company producing the rackets only those entity-based retrieved results that refer to its product, leaving out the reviews on the beer and the movie.

In a more political context of opinion mining, entity linking can also be combined with the quote extraction presented in Paaß et al. [2012], for instance to extract summaries of statements given by politicians in public news papers.

Other Applications

Recently, mainly in the context of the TAC challenges, the community investigated the *slot filling* and *entity creation* tasks. Slot filling aims at enriching existing entities

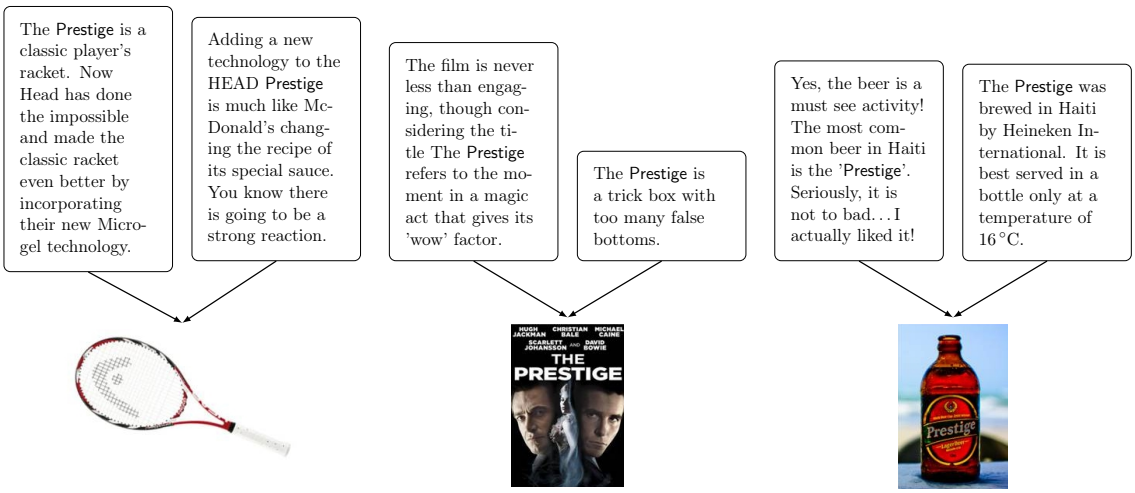


Figure 5.3: Entity Linking for Opinion Mining enables the product-based retrieval of customer reviews.

with new information that is retrieved through the application of entity linking models on new documents. Entity creation aims at automatically generating articles for previously uncovered entities. This requires the clustering and the distinction of uncovered entities to collect the necessary information but also further human investigation to create high quality content from automatically generated summaries.

Similarly, entity linking can also be used to help a contributor during the creation of an article. For instance, we can use entity linking to correct links or detect inconsistencies in redirects. One possible line of application would be to execute our linking model on a new article before it is added to Wikipedia so that links and redirects can be checked for consistency using the predictions of our system. As a side product, such a procedure would also enable an active learning environment for entity linking that can be exploited for an online training method.

Other use cases can be found in the educational context, for example in the enrichment of teaching material with explanatory links to an online encyclopedia. For instance, we may use the Encyclopedia of Machine Learning as reference knowledge resource and link scientific publications against it. Often, publications assume basic knowledge of techniques. Linking a computer science publication to such a dedicated knowledge resource may help a student to understand its contents and contributions. It may also hint him or her at sources that may be more concrete or technical than those provided by Wikipedia articles that maybe only superficially describe the subject.

In this context, we also participated in a Kaggle challenge on author disambiguation, where the purpose was to de-duplicate records in scientific publications[1]. For

[1] `https://www.kaggle.com/c/kdd-cup-2013-author-disambiguation`

this challenge, the usage of external resources was not allowed. Using an adaption of the unsupervised techniques for candidate retrieval as in Chapter 4 resulted in a placement in the first quarter of all participants. In line with the research on author disambiguation, we argue that the incorporation of knowledge resources such as the Encyclopedia of Machine Learning or the DBLP as a database of scientific publications, is more than likely to increase predictive performance.

To summarize, entity linking opens up a multitude of both scientific and industrial applications that will hopefully be investigated in the near future. Regarding the increase in the number of publications investigating this topic during the last years, this is more than likely.

Appendix A

Algorithm: Pseudo Code for Candidate Retrieval (Stage 1)

Algorithm 3: Candidate retrieval (Stage 1)

Input: List of mentions $\mathbf{M} = \{m_1, \ldots, m_k\}$.
Output: Collective search candidate e^{coh} for each $m_i \in \mathbf{M}$, if available.

`// create ensemble query` $q_{\mathbf{M}}$
$q_{\mathbf{M}} \leftarrow q_{l_a}(name(m_1)) \wedge \ldots \wedge q_{l_a}(name(m_k))$
search $\boldsymbol{\mathcal{I}_{\mathcal{W}}}$ using $q_{\mathbf{M}}$
keep the 30 retrieved entities with highest score $s_{\boldsymbol{\mathcal{I}_{\mathcal{W}}}}(q_{\mathbf{M}}, e)$ as source entities $\mathbf{S}_{q_{\mathbf{M}}}$
$\mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \leftarrow \bigcup_{e_{q_{\mathbf{M}}} \in \mathbf{S}_{q_{\mathbf{M}}}} \mathbf{L}_{out}(e_{q_{\mathbf{M}}})$ `// collect outlinks from` $\mathbf{S}_{q_{\mathbf{M}}}$
for $e \in \mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}})$ **do**
  compute $w_r(e)$ (cf. Eq. 4.9)
`// reduce` $\mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}})$ `to the 100 links with maximum weight` $w_r(e)$
while $|\mathbf{L}^*_{out}(\mathbf{S}_{q_{\mathbf{M}}})| \leq 100$ **and** $\mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \neq \emptyset$ **do**
  $e \leftarrow \arg\max_{e \in \mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}})} w_r(e)$
  $\mathbf{L}^*_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \leftarrow \mathbf{L}^*_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \cup \{e\}$
  $\mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \leftarrow \mathbf{L}_{out}(\mathbf{S}_{q_{\mathbf{M}}}) \setminus \{e\}$
$\{\mathbf{e}^c_i(m_i)\}^k_{i=1} \leftarrow \{\emptyset\}^k_{i=1}$ `// initialize candidate sets`
`// relate link targets to mentions`
for $m_i \in \mathbf{M}$ **do**
  for $e \in \mathbf{L}^*_{out}(\mathbf{S}_{q_{\mathbf{M}}})$ **do**
    if $m_i \subseteq name(e)$ *or* $m_i \in \boldsymbol{r}(e)$ **then**
      $\mathbf{e}^c_i(m_i) \leftarrow \mathbf{e}^c_i(m_i) \cup \{e\}$
for $\mathbf{e}^c_i(m_i) \in \{\mathbf{e}^c_i(m_i)\}^k_{i=1}$ **do**
  for $e_{ij} \in \mathbf{e}^c_i(m_i)$ **do**
    compute $coh_{\times}(e_{ij}, \{\mathbf{e}^c_l(m_l)\}^k_{l=1,l\neq i})$ (cf. Eq. 4.10)
for $m_i \in \mathbf{M}$ **do**
  if $\mathbf{e}^c_i(m_i) \neq \emptyset$ **then**
    set e^{coh} for m_i (cf. Eq. 4.16)
return $\{e^{coh}(m_1), \ldots, e^{coh}(m_k)\}$

Appendix B

Supplementary tables from experimental evaluation

This appendix gives the detailed results for the experiments on supervised candidate consolidation as described in Section 4.8.4. Tables B.1 to B.5 show the results on the benchmark corpora **MSNBC**, **ACE**, **AQUAINT**, **CoNLLb** and **IITB** respectively. We report the effect of different search coverages in combination with the prioritization on collective search candidates e^{coh}. For this, we always use the expanded mention names ($\mathbf{S}_n$) as described in Section 4.5 and Section 4.8.4. We also detail the effect of the different weight factors (Eqs. 4.11 to 4.13 and 4.15) used for cross-coherence computation as described in Section 4.6.1. In all tables, the column called "no prioritization" holds the results that are obtained without prioritization on collective search candidates. Further, in all tables, the last line shows the effect of topic similarity as additional feature for candidate consolidation. As described in Section 4.7, this feature is computed from the Hellinger distance $H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$ (Eq. 4.21) over the topic distributions of mention and candidate entity contexts, i.e. $\mathcal{T}_m$ and $\mathcal{T}_{e_{l_a}}$.

To emphasize that the interpretation of model performance is difficult across different performance measures, we give the performance for the best configuration of our model in F_{BoT}, F_{BoT^*} and MAP, the measures used by the related approaches of Ratinov et al. [2011], Hoffart et al. [2011b], Kulkarni et al. [2009], Han et al. [2011], Han and Sun [2012] and described in detail in Section 4.2. The discrepancy among performance measures is especially obvious for **AQUAINT** (Tab. B.3), where the MAP measure would indicate a different configuration than the measure F_{BoT}.

Table B.1: F_{BoT} of our system on **MSNBC** (all values in %). The best value is marked in bold and has associated values of 96.81% in MAP and 91.26% in F_{BoT^*}.

		weighting factors in cross coherence			
search coverage	no prioritization	coh_{SRL^*}	$coh_{\tau\mathrm{SRL}^*}$	$coh_{\cos\mathrm{SRL}^*}$	$coh_{\cos}$
$\mathbf{S}_n$	87.69	86.83	88.12	88.96	86.73
$\mathbf{S}_{nt}$	86.10	88.79	88.22	89.53	88.46
$\mathbf{S}_{ntc}$	86.43	89.50	89.30	89.95	89.20
$+H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$	87.59	89.47	89.50	**89.95**	89.60

Table B.2: F_{BoT} of our system on **ACE** (all values in %). The best value is marked in bold and has associated values of 94.33% in MAP and 85.55% in F_{BoT^*}.

		weighting factors in cross coherence			
search coverage	no prioritization	coh_{SRL^*}	$coh_{\tau\mathrm{SRL}^*}$	$coh_{\cos\mathrm{SRL}^*}$	$coh_{\cos}$
$\mathbf{S}_n$	84.46	86.18	87.91	87.02	86.70
$\mathbf{S}_{nt}$	83.30	87.23	87.75	87.18	87.23
$\mathbf{S}_{ntc}$	86.49	86.76	88.40	88.85	87.75
$+H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$	86.50	86.97	88.44	**89.01**	88.24

Table B.3: F_{BoT} of our system on **AQUAINT** (all values in %). The best value is marked in bold and has associated values of 91.97% in MAP and 82.56% in F_{BoT^*}.

		weighting factors in cross coherence			
search coverage	no prioritization	coh_{SRL^*}	$coh_{\tau\mathrm{SRL}^*}$	$coh_{\cos\mathrm{SRL}^*}$	$coh_{\cos}$
$\mathbf{S}_n$	84.77	84.71	85.07	85.45	84.53
$\mathbf{S}_{nt}$	84.41	84.93	85.61	85.43	84.41
$\mathbf{S}_{ntc}$	84.81	84.50	84.19	84.59	82.95
$+H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$	**86.81**	84.46	84.33	84.94	83.20

Table B.4: MAP of our system on **CoNLLb** (all values in %). The best result is marked in bold and has associated values of 82.16% in F_{BoT} and 78.86% in F_{BoT^*}.

		weighting factors in cross coherence			
search coverage	no prioritization	coh_{SRL^*}	$coh_{\tau\mathrm{SRL}^*}$	$coh_{\cos\mathrm{SRL}^*}$	$coh_{\cos}$
$\mathbf{S}_n$	84.83	85.03	85.71	85.75	85.12
$\mathbf{S}_{nt}$	85.36	86.72	88.13	87.26	87.44
$\mathbf{S}_{ntc}$	86.04	88.23	89.25	88.70	88.80
$+H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$	87.56	88.65	**89.32**	89.13	89.12

Table B.5: F_{BoT^*} of our system on **IITB** (all values in %). The best result is marked in bold and has associated values of 89.95% in MAP and 80.41% in F_{BoT}.

		weighting factors in cross coherence			
search coverage	no prioritization	coh_{SRL^*}	$coh_{\tau\mathrm{SRL}^*}$	$coh_{\cos\mathrm{SRL}^*}$	$coh_{\cos}$
$\mathbf{S}_n$	73.81	74.74	**75.26**	74.68	73.89
$\mathbf{S}_{nt}$	73.96	74.90	75.10	74.85	74.08
$\mathbf{S}_{ntc}$	72.57	69.07	69.81	68.54	69.29
$+H(\mathcal{T}_{e_{l_a}}, \mathcal{T}_m)$	71.10	68.74	69.41	68.35	69.13

References

Charu C. Aggarwal and ChengXiang Zhai, editors. *Mining Text Data*. Springer, 2012.

David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating Domain Knowledge into Topic Modeling via Dirichlet Forest Priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 25–32. ACM, 2009.

Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., 1999.

Amit Bagga and Breck Baldwin. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 79–85. ACL, 1998.

Somnath Banerjee, Krishnan Ramanathan, and Ajay Gupta. Clustering short texts using Wikipedia. In *Proceedings of the 30th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 787–788. ACM, 2007.

Ron Bekkerman and Andrew McCallum. Disambiguating Web appearances of people in a social network. In *Proceedings of the 14th International Conference on World Wide Web*, pages 463–470. ACM, 2005.

Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - A crystallization point for the Web of Data. *Web Semantics*, 7(3):154–165, 2009.

David M. Blei and John Lafferty. Topic Models. In A. Srivastava and M. Saham, editors, *Text Mining: Theory and Applications*. Taylor and Francis, 2009.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Razvan C. Bunescu and Marius Pasca. Using Encyclopedic Knowledge for Named Entity Disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 9–16. ACL, 2006.

Ying Chen and James Martin. Towards robust unsupervised personal name disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 190–198. ACL, 2007.

Rudi L. Cilibrasi and Paul M. B. Vitanyi. The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, 2007.

Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

Silviu Cucerzan. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 708–716, 2007.

Gerard de Melo and Gerhard Weikum. MENTA: Inducing Multilingual Taxonomies from Wikipedia. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1099–1108. ACM, 2010.

Thomas G. Dietterich. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, 10(7):1895–1923, 1998.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. Entity Disambiguation for Knowledge Base Population. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 277–285. ACL, 2010.

Anthony Fader, Stephen Soderland, and Oren Etzioni. Scaling Wikipedia-based Named Entity Disambiguation to Arbitrary Web Text. In *WikiAi (IJCAI workshop)*, 2009.

Angela Fahrni and Michael Strube. Jointly Disambiguating and Clustering Concepts and Entities with Markov Logic. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 815–832, 2012.

Evgeniy Gabrilovich and Shaul Markovitch. Overcoming the Brittleness Bottleneck using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge. In Anthony Cohn, editor, *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 1301–1306. AAAI Press, 2006.

Andrew Gelman, John Carlin, Hal Stern, David Dunson, Aki Vehtari, and Donald Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 3rd edition edition, 2013.

Chung H. Gooi and James Allan. Cross-document coreference on a large scale corpus. In *Proceedings of the 2004 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 9–16. ACL, 2004.

Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1): 5228–5235, 2004.

Daniel Gruhl, Meena Nagarajan, Jan Pieper, Christine Robson, and Amit Sheth. Context and Domain Knowledge Enhanced Entity Spotting in Informal Text. In *Proceedings of the 8th International Semantic Web Conference*, pages 260–276. Springer, 2009.

Ben Hachey, Will Radford, Joel Nothman, Matthew Honnibald, and James R. Curran. Evaluating Entity Linking with Wikipedia. *Artificial Intelligence*, 194:130–150, 2013.

Xianpei Han and Le Sun. A Generative Entity-Mention Model for Linking Entities with Knowledge Base. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, pages 945–954. ACL, 2011.

Xianpei Han and Le Sun. An entity-topic model for entity linking. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 105–115. ACL, 2012.

Xianpei Han and Jun Zhao. Named entity disambiguation by leveraging Wikipedia semantic knowledge. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pages 215–224. ACM, 2009.

Xianpei Han, Le Sun, and Jun Zhao. Collective entity linking in web text: a graph-based method. In *Proceedings of the 34th international ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 765–774. ACM, 2011.

Kazi Saidul Hasan and Vincent Ng. Automatic Keyphrase Extraction: A Survey. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1262–1272, 2014.

Erik Hatcher, Otis Gospodnetic, and Mike McCandless. *Lucene in Action*. Manning, 2nd revised edition, 2010.

Taher H. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the 11th International Conference on World Wide Web*, pages 517–526. ACM, 2002.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, pages 115–132, 2000.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, Edwin Lewis Kelham, Gerard de Melo, and Gerhard Weikum. YAGO2: Exploring and Querying World Knowledge in Time, Space, Context, and Many Languages. In *Proceedings of the 20th International World Wide Web Conference*, 2011a. Demo paper.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust Disambiguation of Named Entities in Text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, 2011b.

Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence*, 1999.

Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. Overview of the TAC 2010 Knowledge Base Population Track. In *Proceedings of the Text Analysis Conference*. National Institute of Standards and Technology, 2010.

Heng Ji, Ralph Grishman, and Hoa Trang Dang. Overview of the TAC 2011 Knowledge Base Population Track. In *Proceedings of the Text Analysis Conference*. National Institute of Standards and Technology, 2011.

Thorsten Joachims. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of the 14th International Conference on Machine Learning*, pages 143–151. Morgan Kaufmann Publishers Inc., 1997.

Thorsten Joachims. Optimizing Search Engines Using Clickthrough Data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142. ACM, 2002.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice-Hall, Inc., 2nd edition, 2009.

Maurice Kendall. *Rank Correlation Methods*. Hafner, 1955.

Saul Kripke. *Naming and Necessity*. Basil Blackwell, Oxford, 1980.

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. Collective annotation of Wikipedia entities in Web text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 457–466. ACM, 2009.

Solomon Kullback and Richard A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):49–86, 1951.

Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1188–1196, 2014.

John Lehmann, Sean Monahan, Luke Nezda, Arnold Jung, and Ying Shi. LCC Approaches to Knowledge Base Population at TAC 2010. In *Proceedings of Text Analysis Conference*, 2010.

Jianhua Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

Bing Liu and Lei Zhang. A survey of opinion mining and sentiment analysis. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 415–463. Springer, 2012.

Donna Maglott, Jim Ostell, Kim D. Pruitt, and Tatiana Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic Acids Research*, 39(Database issue): D52–D57, 2011.

John C. Mallery. Thinking about foreign policy: Finding an appropriate role for artificially intelligent computers. Cambridge: Master's thesis, M.I.T. Political Science Department, 1988.

Gideon S. Mann and David Yarowsky. Unsupervised Personal Name Disambiguation. In *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL*, volume 4, pages 33–40. ACL, 2003.

Andrew McCallum. MALLET: A Machine Learning for Language Toolkit, 2002. URL `http://www.cs.umass.edu/~mccallum/mallet`.

Paul McNamee and Hoa Dang. Overview of the TAC 2009 Knowledge Base Population Track. In *Proceedings of TAC 2009 Workshop*, 2009.

Olena Medelyan, Ian H. Witten, and David N. Milne. Topic indexing with Wikipedia. In *Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, pages 19–24. AAAI Press, 2008.

Pablo N. Mendes, Max Jakob Max, Andrés García-Silva, and Christian Bizer. DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. ACM, 2011.

Rada Mihalcea and Andras Csomai. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pages 233–242. ACM, 2007.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

George A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. ISSN 0001-0782.

George A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

David N. Milne and Ian H. Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, pages 25–30. AAAI Press, 2008a.

David N. Milne and Ian H. Witten. Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pages 509–518. ACM, 2008b.

David N. Milne, Ian H. Witten, and David M. Nichols. A knowledge-based search engine powered by Wikipedia. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pages 445–454. ACM, 2007.

David M. Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. Polylingual Topic Models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 880–889. ACL, 2009.

David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, 2007.

Roberto Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):10:1–10:69, 2009.

David Newman, Edwin V. Bonilla, and Wray L. Buntine. Improving Topic Coherence with Regularized Topic Models. In *Proceedings of the 25th Annual Conference on Neural Information Processing Systems*, pages 1–9, 2011.

Joel Nothman, Tara Murphy, and James R. Curran. Analysing Wikipedia and gold-standard corpora for NER training. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 612–620. ACL, 2009.

Gerhard Paaß, Anja Pilz, and Jochen Schwenninger. Named Entity Recognition of Spoken Documents Using Subword Units. In *Proceedings of the third IEEE International Conference on Semantic Computing*, pages 529–534. IEEE, 2009.

Gerhard Paaß, Andre Bergholz, and Anja Pilz. A Knowledge-Extraction Approach to Identify and Present Verbatim Quotes in Free Text. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, pages 31:1–31:4. ACM, 2012.

Ted Pedersen and Anagha Kulkarni. Name discrimination and e-mail clustering using unsupervised clustering of similar concepts. *Journal of Intelligent Systems (Special Issue: Recent Advances in Knowledge-Based Systems and Their Applications)*, 17(1-3):37–50, 2008.

Ted Pedersen, Amruta Purandare, and Anagha Kulkarni. Name Discrimination by Clustering Similar Contexts. In *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 226–237. Springer, 2005.

Anja Pilz. Entity Disambiguation using Link based Relations extracted from Wikipedia. In *First Workshop on Automated Knowledge Base Construction*, 2010.

Anja Pilz and Gerhard Paaß. Named Entity Resolution using Automatically Extracted Semantic Information. In *Workshop on Knowledge Discovery, Data Mining, and Machine Learning*, pages 84–91, 2009.

Anja Pilz and Gerhard Paaß. From Names to Entities using Thematic Context Distance. In *Proceedings of 20th ACM Conference on Information and Knowledge Management*, pages 857–866. ACM, 2011.

Anja Pilz and Gerhard Paaß. Collective Search for Concept Disambiguation. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 2243–2258. The COLING 2012 Organizing Committee, 2012.

Anja Pilz, Lukas Molzberger, and Gerhard Paaß. Entity Resolution by Kernel Methods. In *Proceedings of the SABRE Conference on Text Mining Services*, pages 71–80, 2009.

Danuta Ploch. Exploring Entity Relations for Named Entity Disambiguation. In *Proceedings of the ACL 2011 Student Session*, pages 18–23. ACL, 2011.

Simone Paolo Ponzetto and Michael Strube. Taxonomy induction based on a collaboratively built knowledge repository. *Artificial Intelligence*, 175(9-10):1737–1756, 2011. ISSN 0004-3702.

Martin F. Porter. Snowball: A language for stemming algorithms. Published online, 2001. URL `http://snowball.tartarus.org/texts/introduction.html`.

Bruno Pouliquen, Marco Kimler, Ralf Steinberger, Camelia Ignat, Tamara Oellinger, Ken Blackler, Flavio Fuart, Wajdi Zaghouani, Anna Widiger, Ann-Charlotte Forslund, and Clive Best. Geocoding multilingual texts: Recognition, disambiguation and visualisation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

Lev Ratinov and Dan Roth. GLOW TAC-KBP 2011 Entity Linking System. In *Proceedings of the Text Analysis Conference*, 2011.

Lev-Arie Ratinov, Dan Roth, Doug Downey, and Mike Anderson. Local and Global Algorithms for Disambiguation to Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 1375–1384. ACL, 2011.

Timothy N. Rubin, America Chambers, Padhraic Smyth, and Mark Steyvers. Statistical Topic Models for Multi-label Document Classification. *Machine Learning*, 88(1-2):157–208, 2012.

Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2nd edition, 2003. ISBN 0137903952.

Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In *Proceedings of the 7nth Conference on Natural Language Learning at HLT-NAACL*, pages 142–147. ACL, 2003.

Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006. ISSN 1541-1672.

Wei Shen, Jianyong Wang, Ping Luo, and Min Wang. LINDEN: Linking Named Entities with Knowledge Base via Semantic Knowledge. In *Proceedings of the 21st International Conference on World Wide Web*, pages 449–458. ACM, 2012.

Dezhao Song and Jeff Heflin. Automatically generating data linkages using a domain-independent candidate selection approach. In *Proceedings of the 10th International Semantic Web Conference*, pages 649–664. Springer, 2011.

Harish Srinivasan, John Chen, and Rohini Srihari. Cross Document Person Name Disambiguation using Entity Profiles. In *Proceedings of the Text Analysis Conference Workshop*, 2009.

Mark Steyvers, Padhraic Smyth, and Chaitanya Chemuduganta. Combining background knowledge and learned topics. *Topics in Cognitive Science*, 3(1):18–47, 2011.

Michael Strube and Simone Paolo Ponzetto. WikiRelate! Computing Semantic Relatedness Using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence*, pages 1419–1424. AAAI Press, 2006.

Fabian Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO - a large ontology from wikipedia and wordnet. *Elsevier Journal of Web Semantics*, 6(3):203–217, 2008.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2000.

Vasudeva Varma, Praveen Bysani, Vijay Bharat Kranthi Reddy, Karuna Kumar Santosh GSK, Sudheer Kovelamudi, N Kiran Kumar, and Nitin Maganti. IIIT Hyderabad at TAC 2009. In *Proceedings of Text Analysis Conference*, 2009.

Raphael Volz, Joachim Kleb, and Wolfgang Mueller. Towards Ontology-based Disambiguation of Geographical Identifiers. In *I3'07*, 2007.

Mirwaes Wahabzada, Kristian Kersting, Anja Pilz, and Christian Bauckhage. More Influence Means Less Work: Fast Latent Dirichlet Allocation by Influence Scheduling. In *Proceedings of 20th ACM Conference on Information and Knowledge Management*. ACM, 2011. (Poster Paper).

Hanna Wallach, David Mimno, and Andrew McCallum. Rethinking LDA: Why Priors Matter. In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems*, 2009.

Kristian Woodsend and Mirella Lapata. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 409–420. ACL, 2011.

Fei Wu and Daniel S. Weld. Autonomously semantifying Wikipedia. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pages 41–50. ACM, 2007.

Yulan Yan, Naoaki Okazaki, Yutaka Matsuo, Zhenglu Yang, and Mitsuru Ishizuka. Unsupervised Relation Extraction by Mining Wikipedia Texts Using Information from the Web. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1021–1029. ACL, 2009.

Yiming Yang. An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval*, 1(1-2):69–90, 1999.

Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 937–946. ACM, 2009.

Zhicheng Zheng, Fangtao Li, Minlie Huang, and Xiaoyan Zhu. Learning to link entities with knowledge base. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 483–491. ACL, 2010.

Zhicheng Zheng, Xiance Si, Fangtao Li, Edward Y. Chang, and Xiaoyan Zhu. Entity Disambiguation with Freebase. In *Proceedings of the 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology - Volume 01*, pages 82–89. IEEE Computer Society, 2012.