Visual analytics methods for supporting semantic abstraction of multivariate time series data

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

vor gelegt von
Gota Shirato
aus
Shizuoka

Bonn, 2024

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

Gutachter / Betreuer: Prof. Dr. rer. nat. Stefan Wrobel
Gutachterin: Prof. Dr. rer. nat. Tatiana von Landesberger

Tag der Promotion: 6. September 2024
Erscheinungsjahr: 2024

Abstract

This dissertation focuses on the challenges and methods in analyzing multivariate time series (MVTS) data, crucial in various fields like finance, healthcare, and sports. The raw MVTS data, often complex due to its volume and multiple attributes, limits insightful pattern recognition. Temporal abstraction is proposed as a solution, integrating elementary data (i.e., sequences of time-stamped attribute values) into meaningful entities for easier comprehension and analysis. At the basic level, the abstraction transforms data referring to time points into interval-based representations. At higher levels, temporal relations between earlier derived representations are used to integrate them into more complex concepts.

Previous research in MVTS analysis has two main directions: computational methods and visual analytics. The former focuses on extracting patterns from data but is not concerned with enabling human analysts to consider the contexts in which the patterns occur and the relationships between the patterns. Visual analytics research encompasses several application-specific studies but lacks a systematic approach to incorporating temporal abstraction in analytical workflows. This dissertation aims to bridge these gaps by developing a framework for integrating computational methods with techniques for interactive visual analysis designed to support human cognition and knowledge generation.

The thesis focuses on the extraction, interpretation, and analysis of patterns from MVTS through progressive abstraction. It starts with methods for detecting temporal patterns in individual variables, progresses to deriving higher-level patterns by combining univariate patterns, and works out approaches to exploring the distribution of these patterns across the dataset. In all these steps, it addresses the problem of supporting human understanding and analytical reasoning by means of effective visualizations.

The dissertation includes the following major parts: First, it focuses on detecting patterns such as up-trends and peaks in discretized MVTS, employing geometric rules and visualization for pattern recognition. Second, it explores identifying patterns in continuous MVTS, using computational algorithms to understand patterns and their temporal relations. Lastly, it introduces topic modeling to examine concurrency between multiple univariate patterns in discretized MVTS.

In conclusion, this dissertation provides a conceptual framework and methods for progressive temporal abstraction in MVTS data. It advances the field by combining computational and visual techniques to aid domain experts in data interpretation. Future research involves adapting and refining this framework across different domains, enhancing temporal pattern analysis, and incorporating expert feedback for continual improvement and broader applicability.

Acknowledgment

First of all, I would like to thank Prof. Dr. Stefan Wrobel for giving me the opportunity to start my doctoral studies, making all the research in this thesis possible.

I would like to show my sincerest gratitude to my daily advisors, Prof. Dr. Gennady Andrienko and Prof. Dr. Natalia Andrienko, who have patiently supported me throughout my doctoral journey. I am grateful for the different approaches they have taken to give feedback. Their guidance has encouraged me to become a better scientist, and I would be honoured if this dissertation could stand even as a small step towards their achievement.

Also, I am thankful to everyone from the University of Bonn or Fraunhofer IAIS, who has encouraged me to tackle technical issues in my research or administrative processes, especially Georg Fuchs, Myriam Jourdan, and Martina Doelp. I am grateful that Prof. Dr. Christian Bauckhage has consented to be the chairperson of my doctoral committee. Prof. Dr. Tatiana von Landesberger and Prof. Dr. Jan-Henrik Haunert have kindly accepted to act as the reviewers. Sven Giesselbach has provided advice on proceeding with the doctoral process and for proofreading the dissertation. This dissertation represents all the supports I have received during my studies at the university and the institute.

To my friends in Japan, who have mentally supported my challenge in Germany: I couldn't have dared to travel to Europe without your support.

Starting a PhD program in a country you don't know very well could be troublesome, especially during the pandemic. My Mitbewohner have been a big help to me in settling down in Germany, they have been good friends since then. I also feel lucky to have all the friends I have made in these years.

My special thanks go to everyone involved in football (i.e., soccer), rugby, and any other sports, irrespective of their tiers or countries (e.g., J.League, English Premier League, German football from 1. Bundesliga, Regionalliga West, Mittelrheinliga, to Kreisliga in Ippendorf, and any small-sided football matches taken place between friends). They always made it easy for me to culturally integrate myself in a new environment and have been the motivation for me to work on my projects during the doctoral period.

Lastly, my parents, sister, and two cats (Don and Ed) have supported my entire journey from distance. It would have been impossible to event stand at the starting point of this journey without them.

Contents

1 Introduction **1**
- 1.1 Background and motivation . . . 2
- 1.2 Problem statement . . . 4
- 1.3 Research questions . . . 5
- 1.4 Outline & Contributions . . . 6
- 1.5 Publications . . . 7
  - 1.5.1 Publications used in the dissertation . . . 7
  - 1.5.2 Additional publications during PhD . . . 8
  - 1.5.3 My contribution in the publications . . . 8

2 Background **12**
- 2.1 Pattern theory . . . 13
- 2.2 Temporal abstraction . . . 13
- 2.3 Extracting or detecting univariate patterns . . . 15
- 2.4 Defining and identifying composite patterns . . . 16
  - 2.4.1 Computational methods . . . 17
  - 2.4.2 Interactive query interfaces . . . 17
  - 2.4.3 Visualization techniques . . . 17
- 2.5 Exploring the distribution of the patterns . . . 18
- 2.6 Introduction to the general framework of MVTS abstraction . . . 18

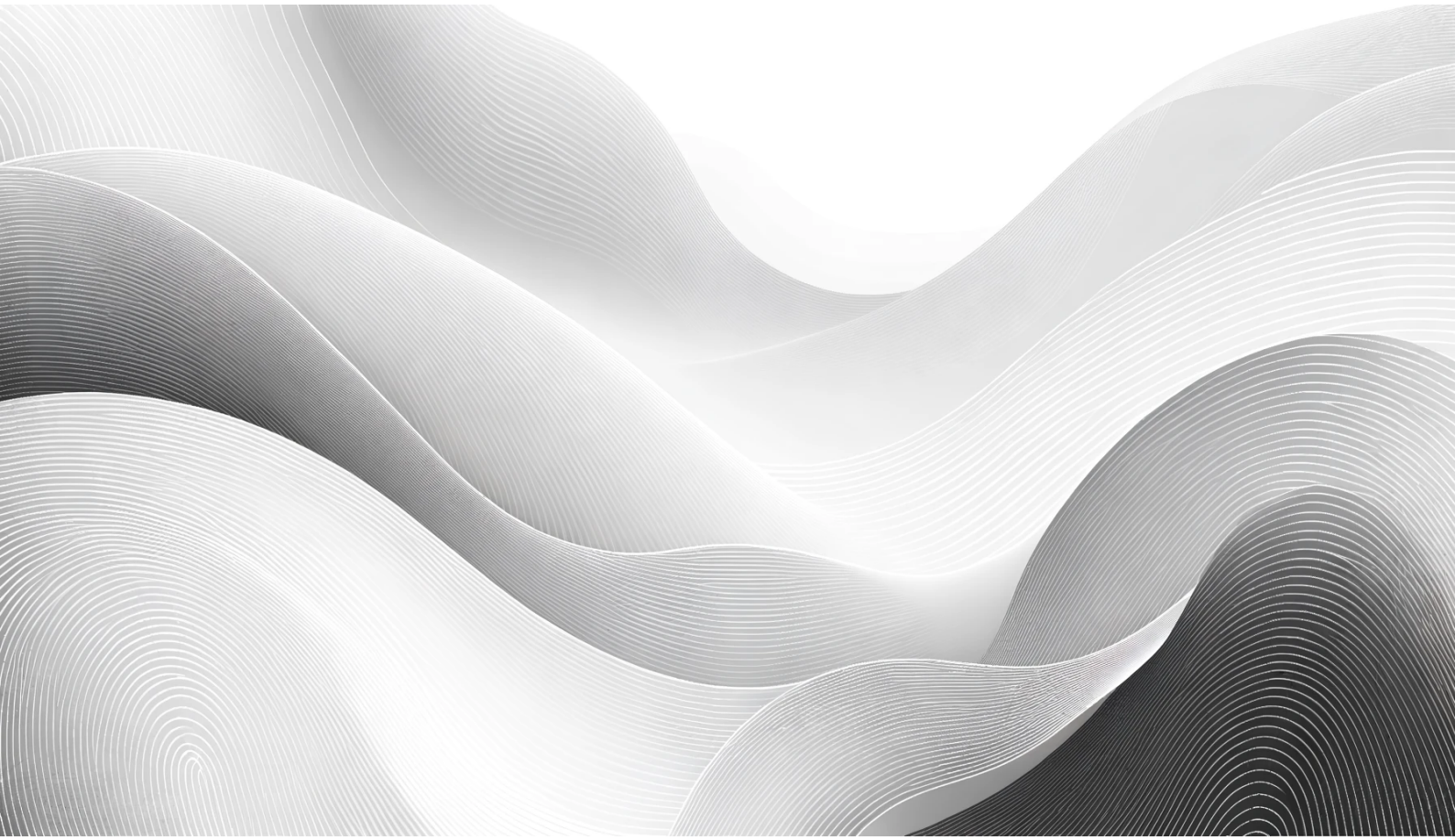
3 Identifying, exploring, and interpreting time series shapes in multivariate time intervals **20**

4 Exploring and visualizing temporal relations in multivariate time series **57**

5 Episodes and topics in multivariate temporal data **94**

6 Discussion **117**
- 6.1 Summary of contributions and discussion . . . 118
- 6.2 Conclusions and future work . . . 126

Chapter 1
Introduction



1.1 Background and motivation

In the current era of technological advancements, the analysis of time series data with multiple attributes, known as *multivariate time series* (MVTS), has become increasingly vital across various domains such as finance, healthcare, environmental monitoring, transportation, and sports. Despite the potential insights that can be gained from these data, domain experts face challenges in extracting meaningful knowledge due to the volume of raw data and the complexity arising from multiple attributes. Raw data points, in their original form, often fail to convey insightful information, making the discovery of patterns an arduous task. The complexity of such data comes from the complex nature of the phenomenon which we try to understand. One of the analytical goals is to enable reasoning about data, which are observations of the phenomena. While visualization techniques can facilitate pattern recognition in the raw data, the insights they provide are products of a result of human perception and cognitive processes. These insights, which appear in the human mind, are difficult to externalize (i.e., represent in an explicit form), share, store, and use in further analysis.

The concept of *temporal abstraction* serves as a key to solving these challenges. Temporal abstraction transforms the mere sequence of data points into a structured form that is more accessible and comprehensible to human analysts and allows for the effective utilization of data by experts. This abstraction process can be applied progressively. Basic temporal abstraction [30] creates interval-based representations from time-stamped data [32], while complex (or composite) temporal abstraction [30] treats these representations as input data for constructing structures with an increased level of abstractions. In this way, a time interval is not just a sequence of data points but can be perceived as a pattern, a composite entity formed by temporally ordered attribute values. Such patterns provide more concise insights into data aspects compared to elementary values.

Within each pattern, *basic pattern types* such as trends or states can be identified. Consider, for example, the statement incorporating a pattern instance: "the temperature *increased* by 10 degrees from Friday to Sunday". This sentence conveys a more intuitive comprehension of weather changes than a mere presentation of the individual temperatures on Friday and Sunday as 15 and 25 degrees respectively.

Following the extraction of basic patterns, these can be interconnected to form *composite pattern types* due to relationships between them, especially temporal ones. Examples include situations like "the temperature first increased then dropped" (i.e., successive pattern type) or "while the temperature was increasing, the air humidity was also increasing, and the wind direction was changing from north to west" (i.e., multivariate pattern type). Hence, through composite temporal abstraction, these basic patterns are used as input data to construct more complex structures, demonstrating how higher levels of abstraction can depict relationships between various basic patterns.

What has been missing in the previous research

In MVTS analysis, there are two primary research directions: computational and algorithmic analysis methods, and visual analytics approaches, as detailed in Table 1.1. The former focuses on deriving artifacts from data, such as patterns, relationships, and structures. The majority of methods in this area implicitly involve temporal abstraction, as they derive summary characteristics or aggregate artifacts from multiple time points. Additionally, there exist frameworks that explicitly define and utilize the concept of temporal abstraction. Knowledge extraction from MVTS data typically involves identifying

patterns across attributes, transforming these patterns into appropriate data structures such as networks, and compressing MVTS into lower-dimensional representations. This extracted knowledge then facilitates various tasks, including classification, clustering, and regression analysis. However, these approaches, while adept at handling large datasets and uncovering patterns, often fall short in terms of intuitiveness and interpretability, particularly for domain experts who may not be rooted in time series analysis. Moreover, there is absence of a conceptual framework that unites these methods and incorporate domain knowledge into the analysis process.

On the other hand, visual analytics approaches focus on supporting analytical workflows by integrating human cognition processes with the computational derivation of analytical artifacts. Although numerous application-oriented studies exist in this domain, employing specific computational methods for MVTS, they often lack a systematic approach or a universal paradigm for incorporating temporal abstraction into visual analytics workflows. Traditional visualization tools, designed for temporal data analysis and focusing on predefined patterns and automated insights, might not fully leverage the specialized knowledge and analytical capabilities of experts.

We should note that while the phenomenon is the primary subject of analysis, to understand the available data and the computational model is also essential [5]. Once we understand characteristics of the data, we can proceed to interpret the phenomenon reflected in the data and build computational models. Then, the insights gained through this process can be further utilized to better understand the characteristics of the data, phenomenon, and model. Human analysis should always be in these iterations to understand the phenomena.

Our work focuses on developing an environment that enhances the utility of existing computational components in data analysis, thereby addressing the gaps in both computational and algorithmic analysis methods and visual analytics approaches. We have constructed a framework that enriches traditional methodologies by integrating human expertise. Our focus has been on creating an environment by integrating existing computational tools. This environment is designed to merge the capabilities of computers with the strengths of human recognition and reasoning and allow for iterative analytical process using interactive techniques. Using visual analytics, we aim to bridge the gap by providing an interactive platform where analysts can visually navigate through the data, gain insights, and iteratively refine their analysis strategies.

Aim and methodology of the dissertation

The aim of this thesis is to establish a conceptual and practical framework for extracting knowledge from multivariate time intervals (MVTS) using a progressive temporal abstraction approach. This methodology uses computational algorithms coordinated with visualization tools to assist in assigning meaning and interpreting the processed data.

First, we present the methods to identify temporal patterns of univariate time series within fixed-length time intervals, i.e., episodes (Chapter 3). In addition to the temporal distributions of these individual patterns, we also investigate joint behavior of multiple patterns, i.e., co-occurrences.

Then, we extend the idea of extracting patterns from time series data by introducing the concept of temporal abstraction (Chapter 4). We propose a conceptual framework for progressive temporal abstraction, from basic patterns of univariate variables within time intervals of various lengths, via complex patterns consisting of a set of the basic patterns,

	Computational	Visual Analytics
Focus	**Deriving artifacts from data**, such as patterns, relationships, structures, and the aggregation of characteristics.	**Supporting analytical workflows** that integrate human cognitive processes with computational artifact derivation.
Existing	**Time abstraction and various methods** to derive temporal abstractions from time series, such as pattern extraction, data transformation, and data compression.	**Application-oriented studies** employing specific computational methods for MVTS, focusing on detection and visualization of predefined patterns.
Missing	**Lack of support for humans** to perceive , interpret, and use the results of computational abstraction. **Absence of a unified conceptual framework** that incorporates domain knowledge.	**Lack of a systematic approach or a universal paradigm** for effectively incorporating temporal abstraction into workflows.

Table 1.1: Contributions in MVTS: focus, existing solutions, and gaps in computational and visual analytics methods

finally to the distribution of these behavior patterns. This framework is supported by the Visualization Mantra (Overview first, zoom and filtering, then details-on-demand) and interaction techniques, such as filtering and coordinated multiple views.

Finally, we focused on the interpretation of the distribution of multivariate patterns within episode-based data (Chapter 5). We investigate distributions of co-occurrence patterns using topic modeling methods.

1.2 Problem statement

To facilitate knowledge building from MVTS, we seek to address the problem of characterizing a complex phenomenon or behavior using abstractions, i.e., patterns that have been detected or derived.

Our approach to understanding these behaviours is structured into three tasks. These tasks are organized progressively: starting from a) detecting temporal patterns of individual variables, advancing to b) deriving multivariate patterns from univariate patterns, and leading to c) exploring the distribution of identified behavior patterns over the dataset.

Detecting and investigating temporal patterns of individual variables

The first task involves detecting and investigating temporal behavior patterns of individual variables. These pattern types provide human analysts with abstract information about the behavior of one variable in each interval of interest. This involves defining a set of basic pattern types that can be easily understood by analysts and transforming time intervals containing elementary value sequences into univariate patterns.

Deriving higher-level pattern types

The second task involves using the univariate pattern types from different attributes as building blocks to form composite pattern types within MVTS. The composite pattern type includes consecutive sequences of univariate patterns from a single attribute (i.e., *successive pattern type*) and joint behavior of multiple variables (i.e., multivariate pattern type). This integration process relies on the exploration of temporal concurrency relations, providing a composite view of multivariate behavior patterns. It combines the detected univariate patterns and their relationships to form multivariate patterns.

Identifying patterns in the distribution of behavior patterns

The third task involves exploring the distribution of identified behavior patterns over the dataset (i.e., with respect to different dimensions of the data). Investigating these distributions includes scrutinizing the frequency distribution, along with the temporal distribution across the time axis and within temporal cycles. Furthermore, this task involves a visual search aimed at unveiling patterns of temporal relations across multiple variables such as those introduced by Allen [2].

1.3 Research questions

Enabling human analysts in building knowledge by providing increasing degrees of abstraction about MVTS presents a challenge that involves the aforementioned tasks. To provide a roadmap for analysis and enable human analysts to fulfill these tasks, we develop a conceptual and methodological framework for supporting semantic abstraction. This framework poses the following research questions:

- **RQ1: Identifying relevant intervals**: How to find relevant intervals in univariate time series, such that the content of each interval can be considered holistically as an interpretable pattern?
- **RQ2: Extracting univariate patterns of individual variables**: How to transform sequences of elementary values of individual variables into constructs that can be interpreted by humans as recognisable behavior patterns?
- **RQ3: Deriving higher-level pattern types**: How to help analysts to 1) define higher-level concepts as combinations of univariate pattern types linked by particular relationships, and 2) identify instances of these concepts (i.e., composite pattern types) in the data?
- **RQ4: Exploring the distribution of identified behavior patterns over the dataset**: How to enable finding patterns in the distribution of earlier extracted patterns over the dataset dimensions?
- **RQ5: Visualization**: How to represent computationally derived constructs to humans to enable pattern recognition and interpretation?

1.4 Outline & Contributions

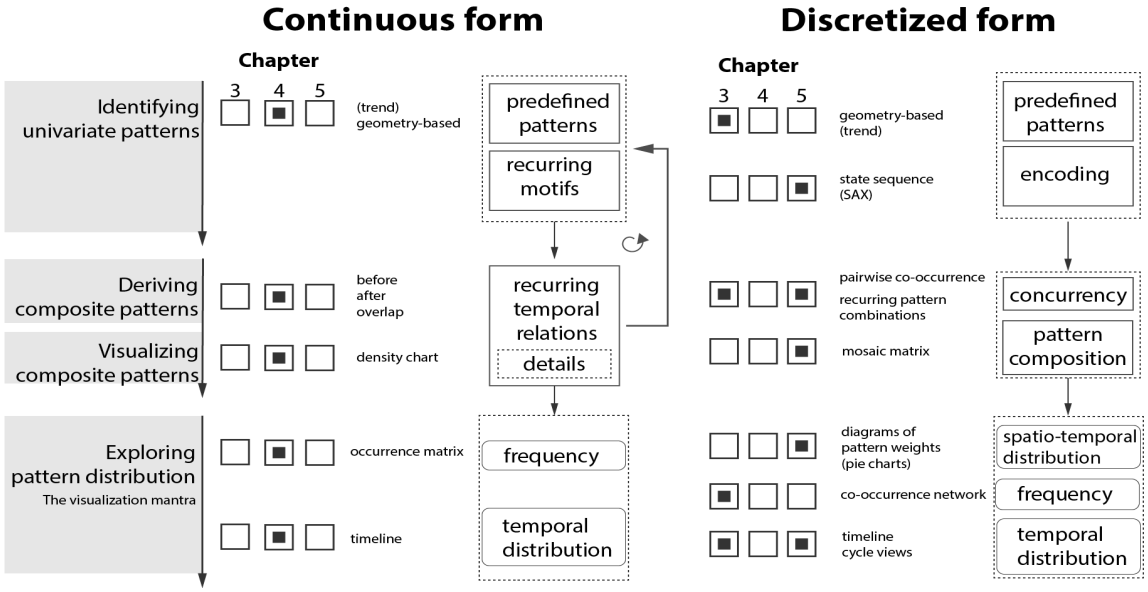


Figure 1.1: An overview of the research scope: A methodological framework for analyzing and visualizing patterns in time series data, differentiating between continuous and discretized approaches.

Figure 1.1 provides an overview of the contributions made in this dissertation. The figure explains our approach to temporal data abstraction, distinguishing between "continuous" and "discretized" forms. For each form, specific abstraction tasks are employed, and the relevant chapters that discuss these methods are indicated, directly addressing the research questions (RQ1 - RQ5) .

Chapter 2 presents the theoretical and methodological background, and offers a succinct overview of the state of the art in the relevant research domains.

Chapter 3 to 5 form the core of this dissertation, with each chapter presenting distinct contributions. Chapter 3 introduces new techniques for detecting univariate patterns in discretized time series, addressing *RQ1 (Identifying relevant intervals)* and *RQ2 (Extracting univariate patterns)* . It proposes techniques for the visual exploration of pattern distribution as well as certain types of relationships between patterns, thus contributing to *RQ3 (Deriving higher-level pattern types)*, *RQ4 (Exploring the distribution of identified behavior patterns over the dataset)*, and *RQ5 (Visualization)* . Chapter 4 builds on this groundwork to create multivariate patterns from univariate patterns in discretized MVTS, addressing *RQ2 and RQ3* . Additionally, it introduces techniques that support the interpretation and exploration of these patterns in terms of time, space, and relevant contexts, contributing to *RQ4 and RQ5* . Chapter 4 focuses on identifying univariate patterns in continuous MVTS without prior discretization, tackling *RQ1 and RQ2* . It explores the temporal relationships between the detected patterns, engaging with *RQ3* and *RQ5* . This process involves a visual search, aimed at revealing patterns of temporal relations across multiple variables, addressing *RQ4* .

Chapter 3 introduces methodologies for characterizing univariate discretized time series, i.e., time series segmented into time intervals, using two approaches for extracting

predefined univariate patterns: state-based and shape-based. The state-based approach encodes temporal data into sequences of states such as high, medium, and low, helping us to understand relative value states within their corresponding intervals. On the other hand, the shape-based approach transforms temporal data into sequences of shapes such as increasing, constancy, or decreasing, which emphasizes the dynamics to clarify general trends and changes over time. Additionally, the chapter presents relationships between univariate patterns (e.g., transitions) and methods for visual exploration of pattern distribution.

Chapter 4 presents an approach to characterizing continuous MVTS, i.e., time series without prior discretization, by extracting univariate patterns and exploring temporal relationships between them. The chapter expands upon the shape-based approach for extracting patterns, introduced in Chapter 3, to extract univariate pattern from continuous time series. Then, it employs Allen's interval algebra to examine temporal relations across multiple attributes. The chapter also presents a visual search technique, enabling the detection of patterns in temporal relations across multiple variables. In contrast to the preceding two chapters that delve into discretized time series, this chapter focuses on continuous MVTS, highlighting the techniques necessary to discern patterns within data that remains undivided.

Chapter 5 continues the exploration of pattern distribution, discussed in Chapter 3, by deriving multivariate patterns from the univariate patterns identified within discretized MVTS. This process relies on the exploration of temporal concurrency relations. For grouping patterns that co-occur frequently in MVTS, we utilize topic modeling. Here, an episode with patterns from multiple attributes is treated as a document consisting of words. To help understand distributions of multivariate patterns, we propose visualization techniques for co-occurrences. Homogeneous groups, identified by topic modeling, are mapped into a 2D space.

Finally, Chapter 6 concludes the dissertation by summarizing the key findings, highlighting the contributions, discussing the implications, and proposing directions for future research. This chapter synthesizes the answers to all the research questions (RQ1 - RQ5).

1.5 Publications

The chapters in this dissertation are based on the following research publications:

1.5.1 Publications used in the dissertation

Chapter 3

- G. Shirato, N. Andrienko, and G. Andrienko, "Identifying, exploring, and interpreting time series shapes in multivariate time intervals", Visual Informatics, vol. 7, no. 1, pp. 77–91, Mar. 2023, doi: 10.1016/j.visinf.2023.01.001. [36]

Chapter 4

- G. Shirato, N. Andrienko, and G. Andrienko, "Exploring and visualizing temporal relations in multivariate time series", Visual Informatics, Sep. 2023, doi: 10.1016/j.visinf.2023.09.001. [34]

Chapter 5

- N. Andrienko, G. Andrienko, and G. Shirato, "Episodes and topics in multivariate temporal data", Computer Graphics Forum, vol. 42, no. 6, Sep. 2023, doi: 10.1111/cgf.14926. [7]

1.5.2 Additional publications during PhD

- G. Shirato, N. Andrienko, and G. Andrienko, "What are the topics in football? Extracting time-series topics from game episodes", presented at the IEEE VIS 2021, 2021. [34]

- D. Brughardt, A. Dunkel, E. Hautahl, G. Shirato, N. Andrienko, G. Andrienko, M. Hartmann, and R. Purves, "Extraction and visually driven analysis of VGI for understanding people's behavior in relation to multi-faceted context", Springer, 2023, pp. 213–234. [9]

1.5.3 My contribution in the publications

Parts of the contributions in this thesis have been published in international peer-reviewed journals. My contributions are clarified based on CRediT (Contributor Roles Taxonomy) [31].

Chapter 3

This chapter contains the paper titled "Identifying, exploring, and interpreting time series shapes in multivariate time intervals" published in Visual Informatics [36]. We introduce a concept of episode referring to a time interval of a dynamic phenomenon that is characterized by a multivariate time series.

My contributions to this research are comprehensive and include various aspects such as formulating research goals, scrubbing and maintaining research data, applying computational techniques, developing methodology, designing computer programs, verifying the overall replication of results, creating visual representations, and writing the initial draft and editing the article.

First, we identified the phenomenon we want to understand: temporal variations of multiple features, i.e., multivariate time series. We aimed to present methods that could analyze multivariate time series, beginning with the identification of temporal development patterns of individual attributes and progressing toward the analysis of patterns of joint development in a set of episodes. After performing necessary preprocessing steps such as feature extraction and normalization, I defined the temporal patterns we utilized in the publication, including down-trend, up-trend, constancy, trough, and peak. I used a rule-based determination process to identify these patterns (Fig. 4). Additionally, I applied computational methods to downsample the time series data, especially when dealing with a large number of data points within a single episode (Fig. 9).

Following the preprocessing and pattern definition, I proceeded to identify the temporal patterns within real-world datasets, specifically focusing on mobility data post-COVID-19 outbreak and tracking data from football matches. To facilitate an analysis, I made the data visualization interactive, which allowed us to iteratively investigate the data with reduced effort and increased efficiency. While common visual representations

such as time series views and co-occurrence network charts were utilized for both datasets (Figs. 8, 15, and 16), I adjusted certain charts to align with domain-specific knowledge. For instance, I used circular displays of temporal patterns to better represent periodic data (Fig. 6). When it came to analyzing beyond univariate patterns, such as coordinating the selection of a pattern in one chart with another or exploring multivariate patterns, we leveraged interaction techniques like filtering and highlighting.

The iterative development of these visual analytics tools was important in reviewing and refining the results, thereby enhancing our understanding of the phenomenon. For instance, after identifying sets of basic patterns that describe tactical movements in football (e.g., slow build-ups as shown in Fig. 14), we could define these sets as higher-level patterns, which can be reused in subsequent analyses. I found that incorporating event data or publicly available external sources relevant to the field significantly improved the reasoning process. For instance, in football datasets, integrating information such as shots and goals, and in mobility datasets, including mortality rates, provided valuable context. This additional data facilitated a more coherent connection between the outcomes of computational methods and our domain knowledge. Finally, to understand the phenomena occurring within the research subjects, I implemented domain-specific data representations using the identified patterns from our analysis. These temporal patterns enabled us to filter temporal data based on the specific patterns we were interested in identifying.

The processes were implemented using D3.js, a JavaScript-based library for data visualization. This library provided extensive customization options for layouts (i.e., positions), interactivity, and animations.

During the writing phase of the research, the iterations we performed with the computational algorithms and visualization techniques provided substantial material directly applicable to the paper. For instance, some charts presented in this chapter, such as Figs. 1, 6, 7, and 8, were produced through iterative analysis illustrating both basic patterns (Figs. 1 and 6) and higher-order patterns (Figs. 7 and 8). This iterative process ensured that the results were robust and well-documented.

Chapter 4

This chapter contains the paper titled "Exploring and visualizing temporal relations in multivariate time series" published in Visual Informatics [35]. We present an approach to analyzing multivariate time series data through progressive temporal abstraction of the data into patterns that characterize the behavior of the phenomenon.

My contributions to this research are consistent with those in the first publication [36], including formulating research goals, cleaning and maintaining research data, applying computational techniques, developing methodologies, designing computer programs, verifying replication of results, creating visual representations, and writing and editing the article.

First, we identified the temporal relationships in multivariate time series that we aimed to understand. Our goal was to establish a method to analyze these series through progressive temporal abstraction, starting with basic patterns in univariate time series and advancing to higher-level patterns formed by temporal relationships between these basic patterns. I applied the pattern detection algorithm from the first publication [36] to segment continuous time series into intervals (episodes) (Figs. 2 and 5). I detailed how these algorithms function with comprehensive captions (Figs. 2, 3, 4, 5, and 6).

Then, I calculated temporal relationships [2] (higher-level patterns) between adjacent time intervals, such as before, after, and overlaps. Finally, I analyzed the distributions of these patterns to provide an overview of the complex behaviors in multivariate time series.

After the temporal abstraction, I applied this method to real-world datasets, focusing on post-COVID-19 mobility data and tracking data from football matches. As in the first publication, I made the data visualization interactive to maximize the efficiency of each chart.

I created relation occurrence matrices to comprehensively describe relationships between all patterns for each feature (Fig. 7). Using the concept of Small Multiples [42], I juxtaposed these matrices for different categories, such as countries for mobility trends (Figs. 8 and 9). Additionally, I applied the t-SNE projection to the similarity values between the matrices to provide a spatial representation of the similarities (Fig. 10).

To improve understanding of the algorithms and the phenomena underlying the data, it was important to visualize both the processes and thresholds involved. In Fig. 4, the iterative division process of a time series into intervals was compared across different thresholds, representing varying tolerance levels for data variation. With larger thresholds, the time series is segmented into broader segments. Similarly, in Fig. 6, the threshold representing the overlap tolerance (ω) is visualized. If the overlap between two intervals is smaller than the threshold, the relation "before" or "after" is assigned instead of "overlap". As described in Chapter 3, the implementation was done using the d3.js library, which allowed easy access to primitive visual components such as points or lines, while offering customization for visualizing the processes and thresholds involved in the algorithms.

One of the challenges was visualizing the temporal distribution of temporal relations between intervals. In addition to the thresholds used for calculating temporal relations, several aspects had to be considered. First, one interval was fixed as a reference, and the temporal distribution of its neighboring intervals was calculated. Next, I considered how to plot the occurrences of neighboring intervals along the x-axis (i.e., temporal axis). In this research, I used the relative time between the middle points of intervals. Although the resulting charts were not always intuitive, integrating interactive elements to explore details of specific elements helped mitigate this issue while maintaining the overall view of the distribution.

The entire abstraction process from raw data through basic and higher-order patterns to their distributions, is fully automated. Any changes of settings, such as thresholds or algorithm selection, will automatically generate new visualization results.

Chapter 5

This chapter contains the paper titled "Episodes and topics in multivariate temporal data", published in Computer Graphics Forum [7]. In this publication, we focused on analyzing episode-based data to understand the distribution of multi-attribute dynamic characteristics across a series of episodes, which can represent various phases or states within the data. In addition to the event-based approach described in Chapter 4, we also employed a sliding window approach, which allowed for overlap between windows. This method was designed to minimize the loss of important behavior patterns that might be missed in non-overlapping windows, ensuring that the capture of temporal dynamics was continuous and more comprehensive.

My contributions to this research included scrubbing and maintaining research data, applying computational techniques, designing computer programs, and editing the article. I developed preliminary algorithms to segment time series into equal intervals using symbolic encoding with three value states: low, middle, and high. We later extended this by adopting the Symbolic Aggregate Approximation (SAX) method [22], which used five value states. For visualizing the resulting patterns, a diverging color scheme, such as from red through yellow to blue, proved to be interpretable.

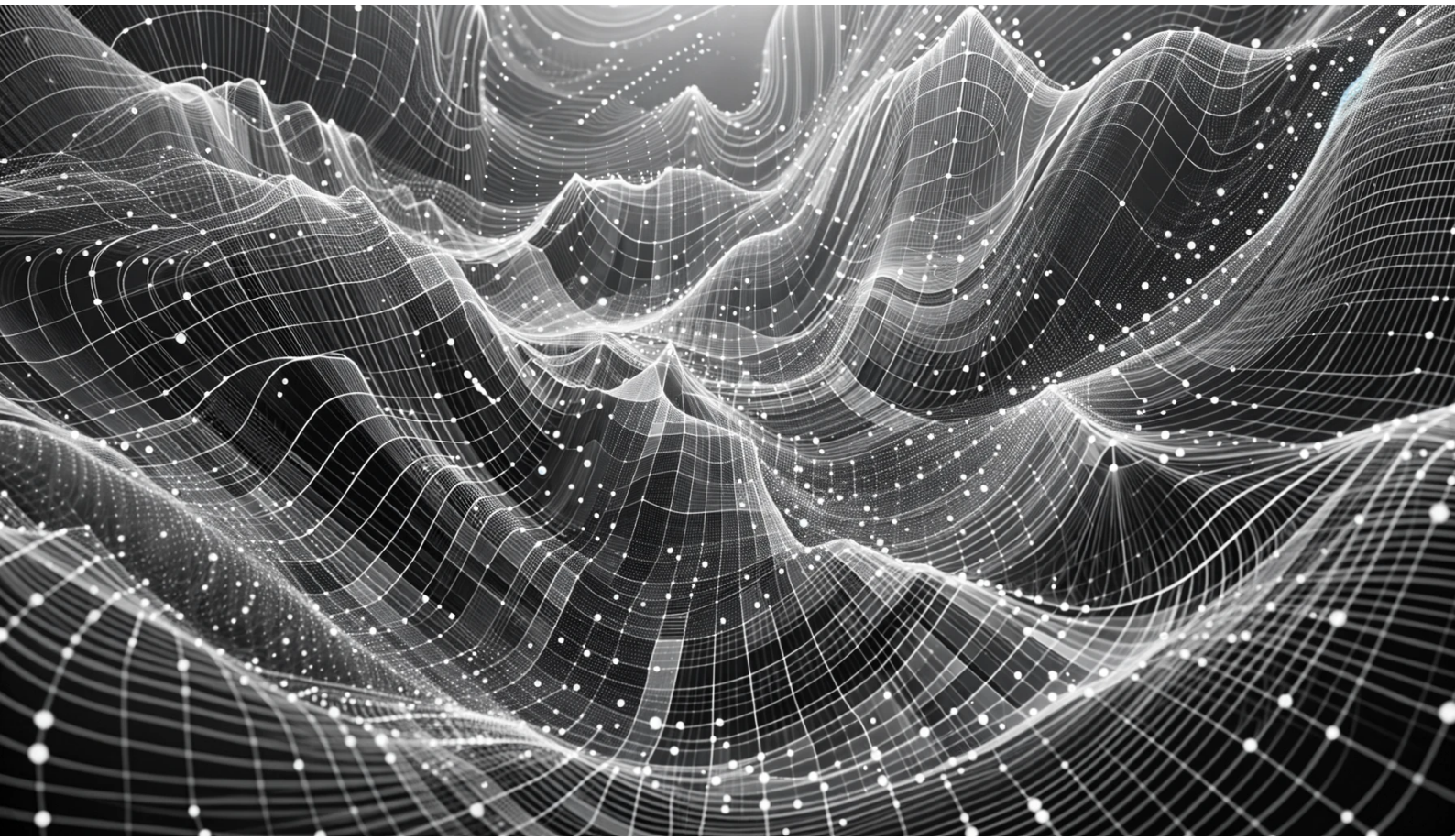
Additionally, I explored the application of topic modeling to non-textual data, a technique typically used in text analysis. In this context, we adopted topic modeling to identify recurring basic patterns across episodes, rather than focusing on the sequential order of those patterns. By doing so, we could generate interpretable results that revealed underlying structures such as co-occurrence of these patterns. This was particularly effective in yielding when applied to football data, where early results were first presented in a poster [34]. This chapter extends and builds on those early concepts and methodologies. Although initial work relied solely on Latent Dirichlet Allocation (LDA) as the topic modeling algorithm, we further compared it with Non-negative Matrix Factorization (NMF) in this chapter. Through our case study of football data, we found that NMF produced more coherent groupings and delivered results that were easier to interpret.

The visualization of topic modeling results was achieved through several approaches, ensuring the versatility of our framework. In addition to matrix displays that featured pie charts or small multiple maps to show topic weights (Figs. 11, 12, and 16), I applied a method that projected data points onto a two-dimensional plane through the use of t-SNE projection. This serves as an initial step for clusering data points based on shared characteristics. However, despite its usefulness in spatially organizing similar data, this method proved to be less intuitive compared to the matrix displays and small multiple visualizations. For instance, the matrix display used Principal Component Analysis (PCA) [18] to align the y-axis, positioning countries with geographically close capitals nearer in the linear order. Furthermore, in the small multiple views, a football pitch served as a background, which allows domain experts to fully utilize their specialized knowledge more effectively during the analysis.

Although the methods were implemented using various tools and systems, such as Python and V-Analytics [3], the entire analysis followed the principles of the progressive abstraction framework. This framework guided our approach by helping us focus on both low-level details, such as elementary points, and higher-level patterns, including univariate and higher-level temporal patterns.

Throughout the iterative process of developing the interactive visual interfaces, we implemented a series of computational algorithms designed to handle multiple levels of abstraction. These algorithms facilitated transitions from elementary data points to higher-level abstractions, including univariate temporal patterns and composite patterns. Our visual interfaces were designed to support the exploration of these abstractions, allowing users to seamlessly navigate between different levels of data granularity. The interfaces provided an environment for uncovering patterns through visualizing both individual data points and their aggregated behavior patterns.

Chapter 2
Background



2.1 Pattern theory

A conceptual framework for the pattern discovery is introduced by Andrienko et al. [6]. Within this framework, a behavior pattern is defined by the relationships found in a distribution that includes at least two data components. To illustrate, in time series data, we typically aim to discover relationships between two data components: 1) time and 2) the corresponding values. Time establishes the set of positions of a temporal distribution (called *base*) while values are associated with the positions. This set of positioned values is called the *overlay* of the distribution.

The authors describe a *pattern* as a combination of relationships between the elements of data components in a distribution. A pattern can be viewed as a unit comprising multiple data elements, interlinked through certain relationships. Temporal patterns are formed by the association of values with their respective temporal positions, the relationships between temporal positions themselves, and the relationships between the associated values. An example of this would be temporally ordered values where each subsequent value is greater than the one before, creating a pattern of increase. This pattern is formed by the temporal order relations between the time stamps and, consequently, the associated values, as well as the "greater than" relationships between consecutive values. Hence, the concept of a pattern emerges as a confluence of these interconnected relationships.

The process of considering a collection of items linked by relationships as one single object is known as *abstraction*. This concept of abstraction can be applied recursively. Initially, it serves to group elementary values into basic patterns.

In this study, a *basic pattern* denotes a sequence of values of a single attribute, arranged over time, that can be represented by a single entity understandable to humans.

In a specific application, patterns termed as "basic" are treated as individual units in comparison to more complex patterns. Although not necessarily simple in nature, these basic patterns are selected as the fundamental elements for the construction of higher-level patterns. Subsequent abstraction processes that are applied to these basic patterns result in the formation of more complex, higher-level patterns. This iterative method of generating increasingly complex patterns from previously created units is referred to as *progressive abstraction*. Basic patterns in time series, such as an increasing trend, are derived via abstraction: treating time as the base and mapping the varying values onto it as the overlay. The degree of abstraction can therefore be progressively adjusted, with elementary values having lower degrees and composite patterns having higher degrees. The gradual transition in degree of abstraction, or progressive abstraction, can enhance the understanding of multivariate time series as it allows analysts to perceive various relationships within the dataset expressed by differing degrees of abstraction.

2.2 Temporal abstraction

Temporal abstraction typically transforms time point-based representation (where each data item refers to an individual time step) into *interval-based* representation (where multiple original data items referring to consecutive time steps are represented by a single entity), defined by their start and end times. These identified patterns can then be transformed into patterns of multiple attributes or used in a visual search to uncover patterns of *temporal relations* between them.

Framework for temporal abstraction

Shahar introduced a conceptual framework for temporal abstraction using domain knowledge, termed as knowledge-based pattern abstraction [32]. This framework decomposes the temporal abstraction task into five subtasks, each relating to specific knowledge types, such as structural, functional, logical, and probabilistic. Such a framework requires a formalized representation of domain knowledge, such as a formal ontology or set of rules. This framework not only structures operations within temporal abstraction but also enhances understanding of temporal changes by addressing the acquisition, maintenance, reuse, and sharing of temporal knowledge.

Defining relevant intervals

Fu [13] presents two types of time series segmentation: fixed-length and dynamic. While the fixed-length approach, including the static sliding window approach [11], is relatively straightforward, the dynamic method offers a more flexible segmentation.

Dynamic time series segmentation has three primary techniques: extensible sliding window, top-down, and bottom-up, as detailed in a survey [19]. Each technique processes the time series several times, stopping once specific criteria are met. What differentiates them is their method of handling time series data: whether to expand, partition, or merge segments.

In the extensible sliding window technique, a segment continues to *expand* until the difference between a prototype pattern and the segment's values exceeds a threshold. Upon reaching the end of one segment, the subsequent one commences immediately.

The top-down technique *partitions* continuous time series into intervals containing some elements that analysts aim to find. Such elements can either be events (i.e., event-based segmentation) or univariate patterns. Event-based segmentation defines segments or episodes based on specific events detected within the time series data. These events, which may be manually annotated or detected by prior computational processes, are then used to identify significant segments that may correspond to particular occurrences or transitions. A practical example of event-based segmentation might be selecting intervals between changes in ball possession during a football match. Time query languages [14], and later, interactive tools such as the TimeMask technique [4] can be used to select time intervals by specifying temporal queries. Another top-down technique is a geometry-based approach, targeting the identification of the largest triangle within the time series, initially designed to downsample time series [39]. A time interval including a large triangle indicates the presence of a sequence of increasing and decreasing patterns (i.e., a peak or trough). These intervals can be further subdivided into intervals with simpler temporal shapes, such as increasing, constancy, and decreasing, depending on analysts' needs.

Lastly, the bottom-up approach *merges* intervals with the shortest patterns possible. This approach can be considered as the complement to the top-down algorithm [19], given that forming the shortest intervals involves partitioning the time series. Adjacent intervals are merged when they include similar patterns, e.g., trends [20].

Temporal relations

Within a multivariate time series, each detected pattern possesses a specific time interval representing its existence. These intervals can have different temporal relations with one

another. The set of possible temporal relations between time intervals, such as preceding, following, and overlapping, is defined by Allen [2].

In a practical application, Lee and Shen explore multivariate time series data to discern relationships between specific trend patterns [21]. They focused on analyzing whether these patterns occur simultaneously and their associations with various factors. This allowed them to identify elements influenced by specific trends, enhancing their comprehension of complex temporal relations within the data.

Progressive temporal abstraction in MVTS analysis

While the concept of temporal abstraction is a recognized concept in the literature, this dissertation advances the field by introducing an approach that applies various levels of abstraction in a progressive manner. The approach provides an environment where human analysts can employ reusable workflows. These workflows, designed to incorporate progressive temporal abstraction, enable analysts to not only apply existing methods but also explore, interpret, and exploit the outcomes of their analyses. Our work, therefore, enhances the understanding of multivariate time series data by enabling analysis of relationships between patterns, thereby distinguishing our research from the previous work.

2.3 Extracting or detecting univariate patterns

Detecting univariate patterns involves the acquisition of abstract representations capable of capturing the temporal behavior of a single attribute. *Supervised pattern detection* involves locating specific predefined patterns within a dataset. The time intervals in the dataset are compared with predefined patterns by calculating a distance measure between them. On the other side, *unsupervised pattern detection* is about deriving patterns from data, usually by identifying recurring structures or sequences without prior knowledge of these patterns. In both cases, pattern detection typically follows a segmentation process, as introduced in Section 2.2.

Types of patterns typically involve state and trend primitives [26]. Takabayashi et al. extend the trend primitives by speed, i.e., fast increasing and decreasing, and peaks, comninations of trends [41].

Supervised pattern detection

Supervised pattern detection primarily involves calculating distances from predefined target patterns for each segment in a time series.

One intuitive approach for pattern detection when target patterns are known is allowing experts to provide explicit representations of desired patterns. This can harness domain expertise or specific requirements. For instance, users can directly sketch target patterns, offering a visual and interactive way to specify which patterns to detect [33]. In situations with an available annotated dataset, supervised learning techniques can be employed. Here, machine learning models are trained on these datasets, enabling them to recognize and classify patterns in new, unseen data.

Patterns in time series, such as trends are often deduced using algorithms that implicitly contain definitions. Besides identifying trend types such as increase, decrease,

and constancy, other trend-related information, such as gaps [40] and speed [41], can also be derived. Moreover, it is worth noting that statistical methods often decompose time series into layers, encompassing seasonality, trend, and random fluctuations [16].

Various distance measures exist for computing similarity between time series segments. Dynamic Time Warping (DTW) measures the distance between two intervals, even when they vary in length [27]. Ye and Keogh suggest comparing subsections of temporal shapes instead of the entire shapes [49]. A related method, SUBDTW, computes the distance between a subsequence and a target trend [21].

Unsupervised pattern detection

When target patterns are undefined, the pattern detection framework by Das et al. [11] is relevant. First, distances between time intervals are determined using a specific distance measure. Then, a clustering algorithm uses the set of distances, typically in the form of a matrix, to group the intervals. The representative fragment of time series in each cluster then serves as the target pattern.

Approximation methods aim to represent data in a simple manner. The simplest technique is to fit linear segments to a series of data points, called Piecewise Linear Approximation [45]. Approximation can be done with segments with more complex structure such as polygonal curves [38].

The Symbolic Aggregate approXimation (SAX) technique [22] discretizes time series data based on standard deviation and mean values. The derived time intervals are symbolized, leading to distinct states that are interpretable with simple concepts like "high", "medium", and "low". This characterization allows for human understanding of state sequences as specific types of behavior. Ruan et al. proposed a distance measure between symbolic series [29].

Decomposition methods represent a univariate time series as a combination of simpler patterns. The original time series can be regarded as the sum of patterns found in these patterns. These layers are often depicted as waveforms, such as square-shaped waves [10] or sine waves [1]. However, since this representation does not align with a single concept, it is not suited for progressive abstraction, especially when deriving multivariate patterns. This aspect is thus beyond the scope of my dissertation.

2.4 Defining and identifying composite patterns

Temporal abstraction involves the formation of composite patterns by integrating univariate patterns from one or more variables. Patterns, in general, emerge due to relationships between entities. While basic patterns are formed by relationships among data elements, composite patterns are formed by relationships between previously identified patterns treated as single entities. Such composite patterns might emerge in numerous ways, such as sequences of univariate patterns or as complex patterns formed by multiple attributes, among other possible configurations. Given the diversity in potential relationship patterns and the myriad ways they can be intertwined, the potential number of composite patterns is theoretically infinite. Researchers tend to concentrate on specific relationship types and the corresponding composite patterns that encompass these relationships.

2.4.1 Computational methods

In MVTS, univariate patterns are typically interconnected by some relationships, forming multivariate patterns. There are two methods for determining relationships, depending on whether intervals of different attributes possess identical time ranges (i.e., starting and ending times), or the time ranges differ across attributes.

Intervals sharing the exact starting and ending time can be considered as co-occurring. For instance, purchasing multiple items within a single transaction, or words appearing in the same sentence, demonstrate this concept. Techniques such as frequent item set mining or association rule mining are commonly used to detect co-occurring patterns.

When intervals do not possess identical starting and ending times, Allen's relation algebra [2] is used to determine the relationships between them. Allen's framework provides 13 interval relations that can describe any relative positioning of two intervals and widely used in formulating temporal rules involving intervals [24]. Some methods use subsets of Allen's relations such as before, after, or containment [46]. Each relation can allow for margins, such as gaps between two intervals allowed for the "before" relationship to be counted [30], or fuzzy duration of relations [17].

Relations between patterns can be depicted as matrices, where rows and columns represent individual patterns, and each element indicates a relation between those patterns [25]. Subsequently, each matrix can be viewed as a high-level pattern (i.e., pattern of patterns) made by the temporal relations between the earlier detected simpler patterns.

Allen's relations are criticized for being ambiguous and ignorant of quantitative differences in overlaps [24]. To address this issue, composite patterns that characterize overlapping patterns are introduced [24]. Using these composite patterns, information such as which univariate patterns overlap during specific time periods is revealed.

2.4.2 Interactive query interfaces

Interactive interfaces allow analysts to explore specific complex patterns. In particular, analysts can combine multiple queries to search for complex patterns.

Outflow [48] treats time intervals with events as single entities. The simultaneous occurrence of multiple events from multiple attributes, which are specific occurrences or actions, forms composite patterns. Users can filter intervals by selecting individual events to be included within these composite patterns.

2.4.3 Visualization techniques

While time intervals are typically depicted as linear elements within a one-dimensional space, Qiang et al. introduce a visualization technique to transform time intervals into points within a two-dimensional space [28]. This space is specifically configured as a triangle area. The horizontal axis depicts time while the vertical axis indicates the duration of intervals. Furthermore, temporal relations between intervals become apparent through their relative positions in the triangle. As a result, query search conditions are shown by areas of overlap, with each area representing different queries such as duration or temporal relations.

2.5 Exploring the distribution of the patterns

Detected patterns, whether basic or complex, typically possess relationships with other patterns in a distribution. Previously detected patterns are treated as entities. Possible relationships between those patterns depend on the chosen base of the distribution, i.e., one of existing data dimensions.

When using time as the base, we explore the positions of the patterns on a timeline and within time cycles, along with their temporal relationships, such as those defined by Allen's temporal relations [2]). Temporal information of intervals, such as start, end and duration, can be mapped onto a 2D plane to identify similar intervals [29]. An interactive calendar display presents the temporal distribution of data clusters throughout a year [44].

Multivariate temporal patterns can be linked with spatial locations (e.g., weather measurements) or spatial entities (e.g., cities). In such cases, the spatial distribution of the patterns may need to be explored. When focusing on spatial dimensions, provided they are present in the data, we explore the spatial positions and spatial relationships, such as proximity and directions. t-SNE projection [43] maps intervals exhibiting similar behaviors closely together, where each data point represents a distinct time interval. Other techniques use maps to represent relationships between geographical areas, in terms of values [47] and spatial links [15].

Irrespective of the chosen dimension, one might also examine patterns' relative frequencies. Visualization techniques for such frequencies include histograms [29], box plots [12], parallel coordinates plots [21], and density graphs [4].

2.6 Introduction to the general framework of MVTS abstraction

We present the general framework of MVTS abstraction, as illustrated in Figure 2.1. We consider two forms of multivariate Time Series (MVTS): discretized or continuous. The latter can be analyzed either in their inherent form or after undergoing some form of discretization. This lays the foundation for our framework that bridges prior research efforts and the contributions of this work.

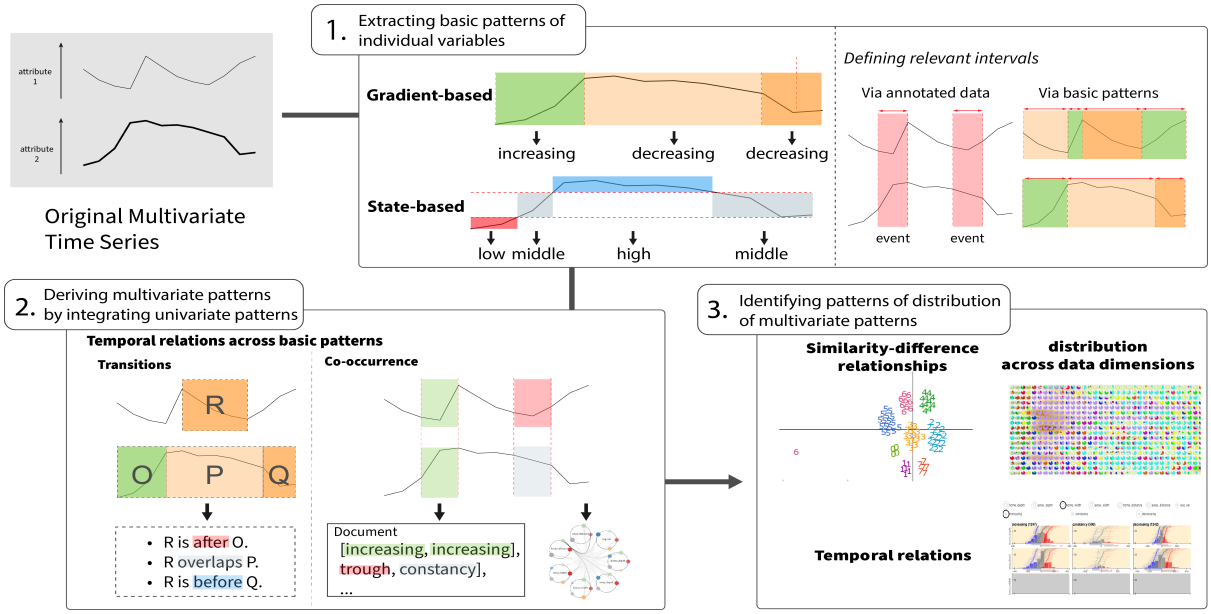


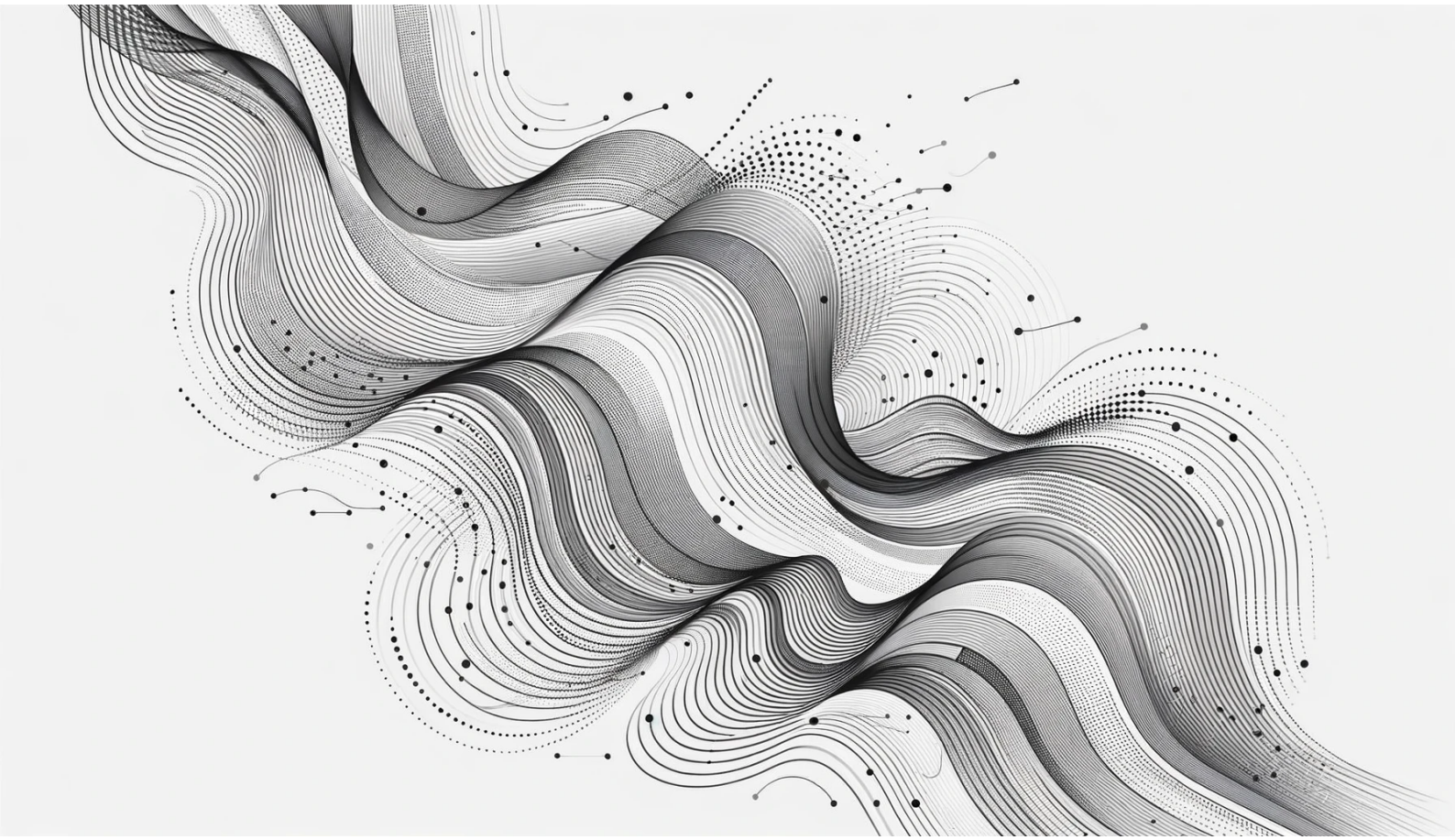
Figure 2.1: Our methodological framework consists of three steps, where the level of abstraction of the multivariate time series increases as we progress through the steps.

In the realm of discretized MVTS, the detection of patterns primarily hinges on two approaches. The first involves recognizing predefined patterns within the discretized MVTS, often termed as pattern recognition or detection. The second approach emphasizes simplifying and encoding the discretized MVTS to enhance its interpretability. Once detected, these patterns usually demand interpretation by human analysts, and in some instances, labeling.

Conversely, for continuous MVTS, the pattern detection process takes a different trajectory. It begins with pattern detection, which can adopt a supervised approach, focusing on predefined patterns, or an unsupervised approach, centering on identifying recurring motifs. The focus of this process is on the temporal relationships between patterns across multiple variables. Such exploration often leads to a frequency analysis where commonly occurring relationships are found. These identified relationships are then interpreted and, if necessary, labeled. An important point in this process is the derivation of composite patterns. In this phase, new patterns emerge by intertwining two previously recognized patterns connected by a specific relation, as captured by the formula `new_pattern := relation_X (pattern_i, pattern_j)`. Composite patterns stand as individual entities when investigating their temporal relationships. Such an examination sets off a new iteration cycle. Every cycle progressively elevates the abstraction level.

Irrespective of their origin, either from discretized or continuous MVTS, there is an examination of pattern distribution across multiple dimensions of data. Through visual analysis, the relationship between the base (e.g., time in time series data) and the overlay (i.e., associated values) results in distinct patterns.

Chapter 3
Identifying, exploring, and interpreting time series shapes in multivariate time intervals



Identifying, exploring, and interpreting time series shapes in multivariate time intervals

Gota Shirato[a,b,*], Natalia Andrienko[a,c], Gennady Andrienko[a,c]

[a] *Fraunhofer IAIS, Sankt Augustin, 53757, Germany*
[b] *University of Bonn, Regina-Pacis-Weg 3, Bonn, 53113, Germany*
[c] *City, University of London, Northampton Square, London, EC1V 0HB, UK*

Abstract

We introduce a concept of episode referring to a time interval in the development of a dynamic phenomenon that is characterized by multiple time-variant attributes. A data structure representing a single episode is a multivariate time series. To analyse collections of episodes, we propose an approach that is based on recognition of particular patterns in the temporal variation of the variables within episodes. Each episode is thus represented by a combination of patterns. Using this representation, we apply visual analytics techniques to fulfil a set of analysis tasks, such as investigation of the temporal distribution of the patterns, frequencies of transitions between the patterns in episode sequences, and co-occurrences of patterns of different variables within same episodes. We demonstrate our approach on two examples using real-world data, namely, dynamics of human mobility indicators during the COVID-19 pandemic and characteristics of football team movements during episodes of ball turnover.

Keywords: temporal patterns, multivariate time series, time intervals

1. Introduction

Everything that happens in the world can be conceptualized as a sequence of episodes representing various events or developments of dynamic phenom-

*Corresponding author.
Email addresses: gota.shirato@iais.fraunhofer.de (Gota Shirato), natalia.andrienko@iais.fraunhofer.de (Natalia Andrienko), gennady.andrienko@iais.fraunhofer.de (Gennady Andrienko)

ena. The term 'episode' means (in the context of our research) a time interval during which something happens or develops. The happening or development can be characterized by multiple time-variant attributes, or features. A data structure containing values of multiple features attained at consecutive time units is called multivariate time series. Our research presented in this paper deals with collections of episodes described by multivariate time series where all features are numeric, i.e., represent measurements rather than categories.

A chronologically ordered sequence of values of a single numeric attribute forms a certain pattern [6]. When such a sequence is represented by a polygonal line along a time axis, the pattern is visually perceived as a certain geometric shape. There are shapes, i.e., patterns, that are not only readily detectable by a human eye but also readily interpretable; moreover, their meanings are denoted by specific terms, such as 'increase', 'decrease', 'peak', etc. Temporal variation of a single feature within an interval can thus be described as one of these simple patterns or a sequence of several simple patterns. Obviously, this can be done for each individual attribute of a multivariate time series. However, the resulting description does not provide immediate holistic understanding of the joint behaviour of the attributes.

The research problem we want to solve is how to proceed from recognition of temporal development patterns of individual attributes to identifying and understanding patterns of their joint development in a set of episodes. To investigate this problem, formulate specific analysis tasks, find approaches to fulfil these tasks, and test the efficacy of these approaches, we use two real-world example datasets: mobility data upon the COVID-19 pandemic and collective movement in football games.

Our research presented in this paper aims to support the following analysis tasks.

- T1: Identify major **temporal patterns** in the variation of **individual features** within the episodes.
- T2: Study the **temporal distribution** of the identified univariate temporal patterns.
- T3: Investigate the **transitions** between univariate temporal patterns in consecutive episodes.
- T4: Investigate the **co-occurrence** of univariate temporal patterns of different features within episodes.

The tasks were defined based on the theoretical model for pattern discovery [6], which is further referred to as "the pattern theory". We do not strive to cover all possible tasks in analysing time series but consider the tasks relevant to the analysis process in which higher-level patterns are constructed from lower-level patterns. In this process, task T1 extracts lower-level patterns and tasks T2-T4 aim to discover different types of higher-level patterns formed by the lower-level patterns.

For T1, we introduce an algorithm to extract temporal patterns from univariate time series. T2 is supported by a timeline display and, when appropriate, by circular charts with the circumference representing a temporal cycle. The latter can facilitate detection of periodic re-occurrence patterns in the temporal distribution. For T3, we propose bipartite graphs showing frequencies of pattern transitions. T4 can be fulfilled by interacting with a co-occurrence network.

The rest of this paper is structured as follows. Section 2 discusses the related work. Section 3 introduces the proposed techniques and approaches using the example of the mobility data during the COVID-19 pandemic. Section 4 demonstrates the generality of our approach by example of another application using football (soccer) data. Section 5 discusses the concept, approaches, and answered research questions, identifies strengths and limitations, and proposes directions for future work. Finally, section 6 concludes our work.

2. Related Work

We introduce previous approaches to pattern detection, interpretation and visualization applicable to multivariate time series in episodes.

2.1. Conceptual foundations

Collins et al. [12] define a pattern as a holistic representation of multiple (data) items abstracted from the individual items. The concept of a data pattern and the existing definitions in different research disciplines have been extensively discussed by Andrienko et al. [6], who argued that patterns are formed by relationships between data items. A data pattern involves elements of at least two sets, for example, time units and values of a numeric attribute. A pattern is made by intrinsic relationships between the elements within these sets and the correspondences between the elements from the different sets. The former depend on the nature of the sets and the latter are defined

in the data. The intrinsic relationships between elements of one of the sets create a particular arrangement of the corresponding elements of the other set. A pattern is the manner in which the elements of the second set relate one to another throughout this arrangement.

For example, the intrinsic relationships between time units are temporal ordering and temporal distance, i.e., the amount of time that passed between two units. The intrinsic relationships of ordering and distance (i.e., difference) exist also between values of a numeric attribute. A data set specifies what attribute values corresponds to which time units. The intrinsic temporal relationships between the time units create a temporal arrangement, i.e., a sequence, of the corresponding attribute values. A pattern is the manner in which the values differ one from another along this sequence: whether values that are further in the sequence are greater or smaller than the preceding values or nearly equal to them. Depending on these relationships, we identify the pattern as increase, decrease, or constancy.

The definition of a data pattern as a system of relationships implies that visual discovery of data patterns can be enabled by visualizations satisfying two requirements: (1) appropriately represent the pattern-forming relationships according to the types of data components and (2) facilitate holistic perception of multiple data items. Thus, in a case of a numeric time series, a line chart (a.k.a. time plot) is a suitable visualization: two axes appropriately represent the ordering and distance relationships between time units and between numeric attribute values, the positions of points in this coordinate system accurately represent the correspondences between the time units and the attribute values, and holistic perception is facilitated by connecting the points by lines. A temporal pattern is thus perceived as a particular shape of the resulting polygonal chain. Hence, discovery of temporal patterns can be done by identifying shapes.

According to the pattern theory [6], data patterns that have been discovered can be treated as new elements of data to which the subsequent analysis steps are applied. The analysis involves determining relationships between the patterns throughout arrangements created by elements of other data components, e.g., how the patterns vary along time or how they are distributed over space.

2.2. Temporal pattern extraction and classification

A comprehensive overview of visual analytics approaches for temporal data can be found in monographs by Aigner et al [1], Andrienko et al [5] and

Tominski and Schumann [41].

An important pre-requisite for pattern extraction is segmentation of multivariate time series into semantically meaningful episodes. Papers by Bernard et al [9] and Gharhabi et al [16] propose visual interactive and computational approaches to segmentation. Further works propose semantic segmentation based on TimeMask [4] and its extensions [3].

There exist two major building blocks for temporal pattern extraction and classification. First, there exist methods that search for patterns specified by their shapes. Second, similarity measures (also called distance functions) are used for quantifying similarity and detecting patterns in time series.

Several papers proposed libraries of temporal patterns for univariate [31] or multivariate [44] time series. Das et al. apply a data-driven approach for identifying patterns with interpretable and recognizable shapes [14]. Algorithms for measuring similarity to pre-defined patterns were proposed for detecting time series that contain the given patterns [30] and, in contrast, for detecting dissimilar subsequences in time series [26]. Other approaches to pattern detection and analysis include representations of time series as aggregates [25] or as sequences of symbols [28].

In our work, patterns are identified by means of a new algorithm that calculates the largest triangle within a time series for determining the pattern shape. The idea of the algorithm originates from Steinarsson, who aimed at downsampling time series for visual representation [39]. Unlike the most common approaches, which are based on computing similarities to earlier defined shapes, either taken from a library or sketched by a user, our approach takes into account geometric characteristics of a time series fragment and provides a useful opportunity to represent the patterns in a highly schematic and compact manner using two or three points.

Apart from the research on extraction of predefined patterns and on recognition of pattern types, there are also works where time series patterns are identified implicitly by means of clustering assuming that each cluster defined a certain pattern. The main idea has been exemplified by val Wijk and van Selow [42], who clustered daily univariate time series and investigated the distribution of the clusters over a year. Schreck et al. [37] proposed to treat time series of two variables as trajectories in 2D space. Long time series were divided into episodes, the trajectories from the episodes were clustered, and the original time series were represented as sequences of the averaged trajectory shapes generated for the clusters. From the perspective of our work, these approaches are interesting for their focus on exploring the distribution

of temporal patterns rather than solely on pattern detection and extraction.

2.3. Visualization of multivariate time series and episodes

An obvious approach to visualization of multivariate time series is to create multiple visualizations representing the time series of the individual variables. For example, Janetzko et al. [24] create multiple horizon graphs [22] to visualize multiple time series characterizing episodes of a football match. Hao et al.[20] focus on showing the occurrences of earlier detected frequent patterns (motifs) in long time series represented by line graphs. Pham et al. [33] complement multiple area charts showing variation of singular variables with a temporally ordered sequence of radar charts showing combinations of values of the variables. Other authors strive to create a compact view, such as Kaleidomaps [8], where each time series is represented by a heat map embedded in a sector of a circle. A popular technique utilized in visual exploration of multivariate time series data is applying dimensionality reduction to the combinations of attribute values corresponding to the time steps [10, 40]. In these works, the authors are dealing with continuous time series rather than episodes.

In visualizing episodes characterized by multivariate time series, it is necessary to address:

1. **When** the episodes happened: representing their temporal references in linear [13] or cyclic time [32, 11] or structural (calendar) models [42];
2. **What** happens within each episode: temporal dynamics of attributes, usually represented either by displaying time lines [13] or animating representations such as scatter plots [35, 36, 21]. Zhao et al. proposed an interactive visualisation for episodes which facilitated comparison of timelines with different attributes [45];
3. **How** multiple episodes relate to each other: what are transitions between the episodes. This can be represented, for example, by a node-link diagram with nodes representing patterns and links - transitions between them [29].

In analysing the times of the episode occurrences (when), not only the temporal distribution of the episodes is of interest but also the temporal relationships between episodes. Allen and Ferguson systematically introduce all possible pairwise relations between time intervals [2]. These relationships can be represented graphically using triangular logic introduced by Van de

Weghe [43]. Qiang et al. [34] used this approach for representing temporal relationships between episodes. Lee and Shen [27] propose techniques for visual exploration of temporal relationships between occurrences of user-defined patterns (called "trends" by the authors) in multivariate time series. They transform the time series into a sequence of states characterized by different combinations of trends and propose a visual representation in the form of a matrix with columns corresponding to the states and rows to the trends of the different variables.

Our paper uses several visual representations that combine ideas from the mentioned earlier works. Specifically, the idea of our timeline view (Fig. 1) is similar to the visualization of state sequences by Lee and Shen [27], the circular charts (Fig. 6) utilize the idea of Ringmaps [46], and the representation of temporal patterns by colours in various displays follows the ideas of van Wijk and van Selow [42].

3. Visual analytics approach

In this section we introduce our visual analytics (VA) approach that helps analysts to explore and understand large sets of episodes characterised by multivariate numeric time series.

3.1. Essence of the approach

The key idea of our approach is to abstract each individual time series within each episode to a temporal pattern. All patterns are assigned to a finite (preferably small) set of classes, or pattern types, which can be denoted by semantically meaningful labels or somehow encoded in a symbolic form. Hence, each individual time series is represented by a reference to the corresponding pattern class, and each episode is represented by a combination of pattern classes of the multiple attributes. The following analysis is done using this representation of the episodes. For the sake of brevity, we shall henceforth use the term 'pattern' to refer to a pattern class.

According to the pattern theory [6], we treat the temporal patterns that have been obtained as new elements of data. We strive to find higher level patterns in the distributions these new elements with respect to the other components of the data, which are the set of the episodes considered as discrete objects and the time with its intrinsic relationships of temporal ordering and distances; see Section 2.1.

Hence, based on the pattern theory summarised in Section 2.1, our approach includes two stages:

1. Detect and abstract temporal patterns of singular attributes appearing in the episodes.
2. Treating the univariate temporal patterns as data elements, study the distribution of these "elements" within the set of episodes and along time.

In this approach, we deal with patterns of two levels of complexity and sophistication. The first stage discovers lower-level patterns formed by temporally ordered numeric values. The second stage aims to discover higher-level patterns formed by these lower-level patterns due to their relationships and thereby imposed arrangements within and across the episodes. Within the episodes, the univariate patterns of multiple attributes are linked by the relationships of *co-occurrence*. Across the episodes, the univariate patterns are linked by relationships of *temporal ordering and temporal distance*.

The task **T1** formulated in Section 1 refers to the first stage and the remaining tasks to the second stage. The task **T4** focuses on the relationships of co-occurrence within episodes. The expected type of higher-level patterns is which univariate patterns tend to frequently co-occur and which do not occur together. The task **T3** focuses on the temporal ordering and strives to find patterns of frequent or infrequent occurrence of one lower-level pattern immediately after another. The task **T2** focuses on more distant temporal relationships regarding the arrangement of the lower-level patterns along the time axis and, when appropriate, within temporal cycles. The expected types of higher-level patterns include tendencies to occur earlier or later in time or at certain positions in a cycle, to re-occur more or less frequently in different time periods, to occur in a particular sequence, etc.

As stated by the pattern theory [6], pattern discovery is supported by faithful visual representation of relevant relationships. Taking into account the aforementioned relationships that are relevant for tasks **T2-T4**, we propose the following visualizations to support these tasks:

- **T2**: A timeline display of the temporal patterns (Fig. 1), where the horizontal axis represents the linear ordering relationships between time intervals, plus circular diagrams (Fig. 6), where positions in circles represent the cyclic arrangement relationships.

- **T3**: A bipartite graph of immediate transitions between patterns of the same attributes (Fig. 7).
- **T4**: A co-occurrence network (Fig. 8).

The task **T1** can be fulfilled in different ways, for example, by dividing the time series into intervals and encoding the interval averages by symbols according to the value ranges in which the averages fall. The resulting codes are called SAX patterns [28]. In our paper, we propose another method, which is based on the recognition of the geometric shape that would be formed when the time series is represented graphically by a line chart. It should be noted that the visual analytics techniques we propose for the tasks **T2-T4** do not depend on the method of extracting and encoding temporal patterns and on the choice of labels to denote the patterns.

We demonstrate our approach on example of Google Mobility data [18]. Continuous time series of daily mobility indicators were divided into disjoint episodes.

3.2. Approach introduced by example

The COVID-19 pandemic has impelled local authorities and/or governments to regulate people's mobility. Such policies generate changes in mobility, which are typically sporadic across a certain period. We should distinguish those sporadic patterns from seasonal repetitions in mobility data. For example, we can expect the increasing number of people staying at home and the decreasing number of those going out during the Christmas season. Moreover, different categories of places have different patterns of mobility even during the same time interval. Here, our interest is to visualize temporal patterns across episodes and to investigate how the mobility changes over time across different categories of places.

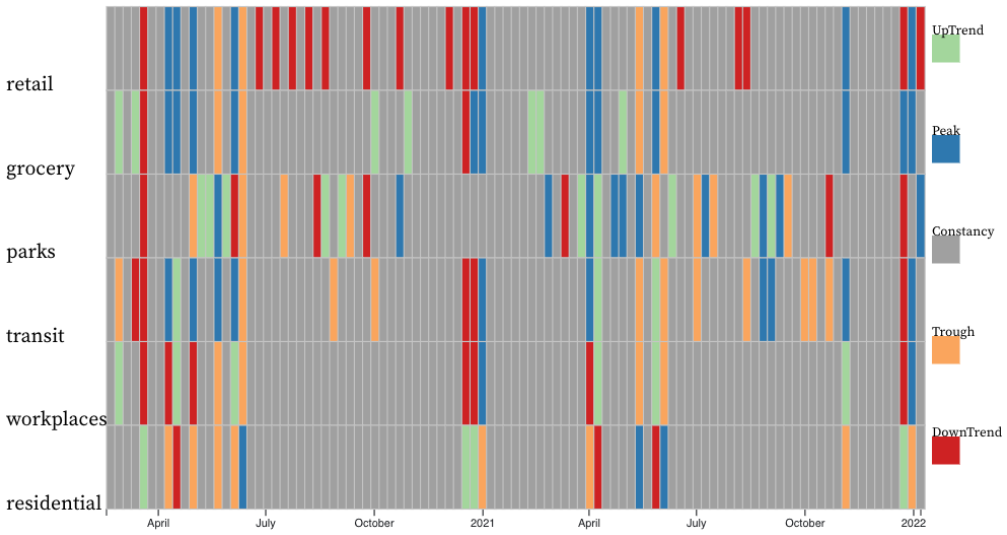


Figure 1: Timeline of temporal patterns in Google Mobility Data.

Data Description

We preprocess the mobility data provided by Google [18] to obtain multivariate time series. Since the COVID-19 outbreak around February 2020, Google has been daily publishing anonymized mobility data for 6 different categories of places (namely, retail and recreation, supermarkets and pharmacies, parks, public transport, workplaces, and residential) from different regions. The data consist of daily visitor numbers to these categories of places relative to baseline days before the pandemic outbreak. Baseline days represent a normal value for each day of the week and are given as the median value over the five-week period from January 3rd to February 6th 2020. The values in the published data are expressed as percentages of the changes from the baseline values.

From the continuous time series, we extract the time intervals of weekdays (i.e., 5 time steps for each week) with the corresponding segments of the time series. Mobility data for weekends are excluded from the analysis because changes from weekdays to weekends are very prominent and therefore obscure the longer-term changes of the mobility behaviours. We process the mobility data for Germany collected between the 17th of February, 2020 and the 7th of January, 2022 (i.e., almost for two years), which results in 99 episodes of the length of five time steps.

For validation purposes, we acquired values of eight policy indicators

(namely, closing of schools, workplace, and public transport, cancelling public events, and restrictions on internal and international movement) from the Oxford COVID-19 Government Response Tracker [19].

T1. What are the major patterns of individual attributes?

In the introduction, we mentioned the existence of simple, easily perceivable and interpretable patterns of temporal variation of numeric attributes. These patterns can be schematically represented by lines of particular geometric shapes. Let us use the term "elementary pattern" for a pattern that can be represented (in abstraction from minor fluctuations) by a single straight line. There are three elementary temporal patterns: up-trend, constancy, and down-trend. More complex patterns can be considered as sequences of these. Fig. 2 illustrates how elementary temporal patterns can make more complex temporal patterns. Any temporal pattern starts with one of the elementary patterns, and a sequence of two or more temporal patterns can make a composite pattern such as a peak or a trough.

When a sequence consists of the same kind of elementary pattern (i.e., up-trend, constancy, or down-trend), we can simply consider it as a single temporal pattern. For example, a sequence of two up-trend patterns makes a single up-trend pattern and this temporal pattern makes a peak pattern with a subsequent down-trend (i.e., up→up→down makes peak). Note that when the sequence gets longer, it can create a more complicated shape. A long time series often looks like an oscillation. It can be simplified by means of temporal smoothing. We assume that the episodes under analysis are short, so that the time series include a small number of time steps and thus can be represented by sufficiently simple patterns. Longer episodes can be subdivided into shorter ones to enable such representation. Another possibility is to downsample the time series, i.e., reduce the number of time steps by dividing a long sequence of time steps into a small number of intervals and taking a single representative value (e.g., the mean or median) from each interval.

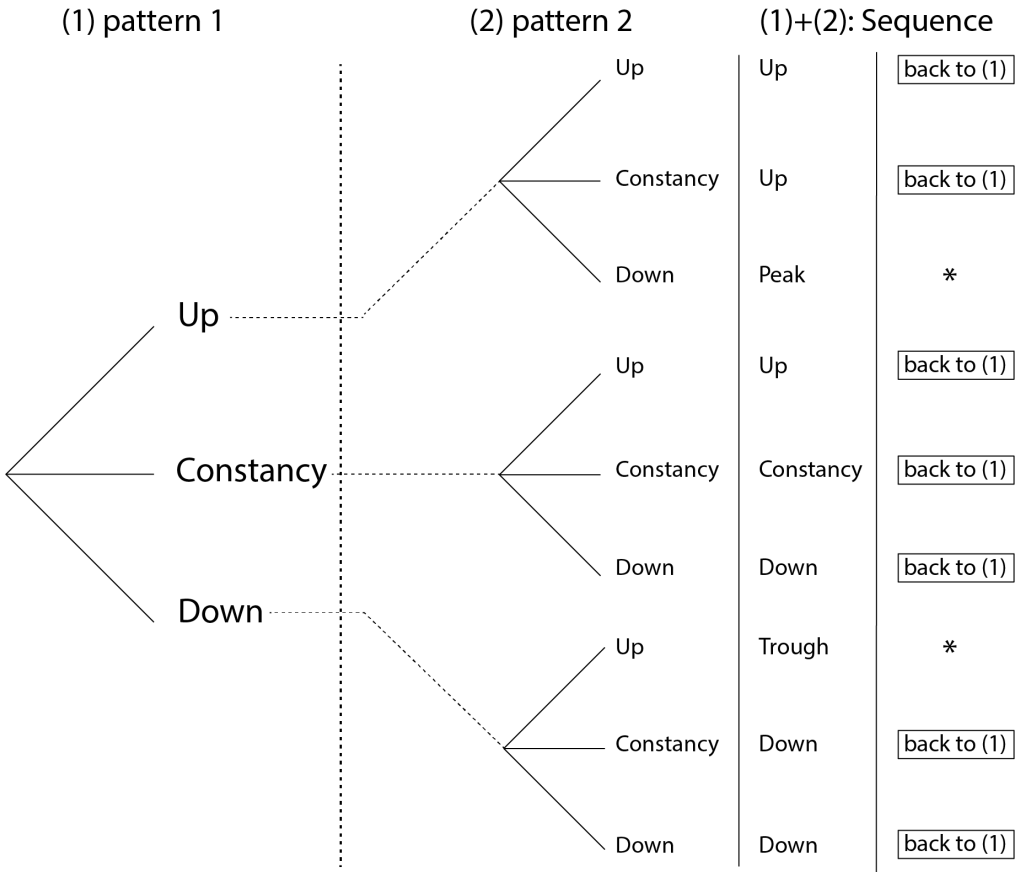


Figure 2: Possible sequences of elementary temporal patterns. For example, a temporal pattern consisting of up-up-constancy-down-down will be classified as a peak. A long time series including a peak or a trough (marked *) may require a subtle adjustment to distinguish different temporal patterns.

We assign an episode to one of the five temporal patterns to represent the most prominent shape of the time series: up-trend, peak, constancy, trough, down-trend. To determine a temporal pattern, we adapt the main idea from the algorithm of Steinarsson [39], which was devised for downsampling of time series , i.e., reducing the number of points used to represent the time series. This matches very well our goal to transform time series into simple shapes that can be represented by very few points. The method is based on finding the data point that makes the largest triangle when connected to the first and last data points in a time interval. Fig. 3 shows an example of the

largest triangle in a time series. Fig. 4 illustrates the work of the pattern determination algorithm, which is explained below.

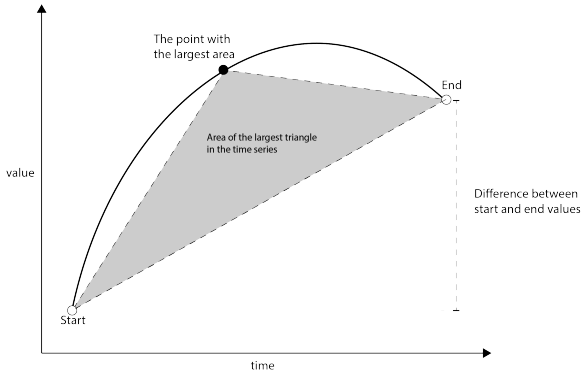


Figure 3: The largest triangle in time series.

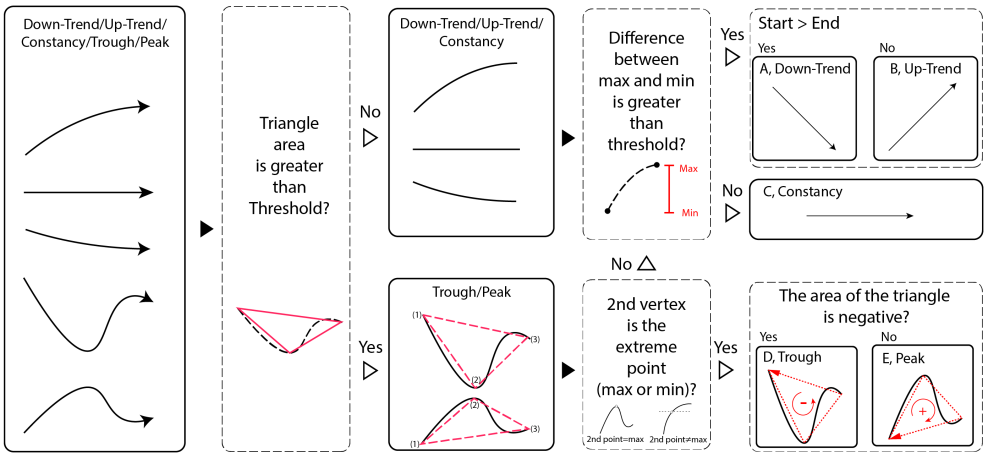


Figure 4: The process of pattern determination. Time series will be classified into either A. Down-Trend, B. Up-Trend, C. Constancy, D. Trough, or E. Peak.

A time series can be classified as peak or though when the area of the largest triangle is greater than a chosen threshold. To find the largest triangle, we take the first and the last points of the time series as the first two

vertices of a triangle and test all intermediate data points one after another as potential third vertices of the triangle. From these points, we take the one that makes a triangle with the largest area among all.

If the absolute value of the area of the largest triangle is above the threshold, the time series has either a peak or a trough; otherwise, it can be classified as a trend (up or down) or constancy. The value of the area is treated as negative when the order from the start, via the extreme, to the end points is counter-clockwise. Otherwise, the area has a positive value. The time series has a peak with a positive area and a trough with a negative area.

When the time series is neither peak nor trough, meaning that the values do not significantly deviate from the straight line connecting the first and last points, the time series has either of the following patterns: an up-trend, a down-trend, or a constancy. Imagine a time-distance graph for uniform velocity, where distance increases at the same pace. In this case, no triplet of the points makes a triangle, and we define the area to be zero. This pattern determination is relatively straightforward; when the difference between the start and end values is larger than a chosen threshold, the time series has either an up-trend or a down-trend, otherwise it has a constancy. Then the time series has a down-trend when the start value is greater than the end, and an up-trend happens when the end value is greater than the start.

Results of pattern detection depend on two thresholds that we use for determining peaks vs. troughs and identifying constancy patterns. The specific values of the thresholds are not essential for demonstrating our approach. Generally speaking, these thresholds are application-specifics, and domain knowledge may be needed for setting them properly. In our example, we've performed self-assessment to choose appropriate values based on several trials. For the assessment, we used a visualization with time series translated to a common starting point, as in Fig. 5.

Table 1 presents the distribution of temporal patterns in the mobility data. We observe constancy as the most frequent among the patterns. This observation can be confirmed by time series visualizations in Fig. 5.

The types of patterns our algorithm aims to extract can be categorised as *patterns of value change*, while, for example, SAX patterns [28] can be seen as *patterns of value magnitude*. Our algorithm ignores the magnitudes of values and considers only the differences with respect to the first value of a time series. This needs to be taken into account when assessing the suitability of our algorithm for specific analysis goals. Another important note is that the algorithm allows extraction of a more refined set of pattern types than

Table 1: The overview on temporal patterns in their frequency (Peak/Trough threshold = 0.1, Constancy threshold = 0.2). We observe constancy as the majority.

	Peak	UpTrend	Constancy	DownTrend	Trough
retail_and_recreation	10	0	76	9	4
grocery_and_pharmacy	12	4	77	2	4
parks	11	9	64	7	8
transit_stations	10	3	70	5	11
workplaces	2	6	80	7	4
residential	3	4	81	3	8

we consider in our examples. Thus, for the peak and trough patterns, it is possible to introduce subtypes based on whether the final value of the time series increased, decreased, or remained nearly the same as the first value. For the up- and down-trends, it is possible to distinguish steep and gradual increase or decrease. An appropriate level of pattern abstraction can be chosen in accord with the goals of analysis. In our examples, we extract and use highly abstracted patterns; however, the exploratory techniques we demonstrate can also be applied to an extended set of patterns.

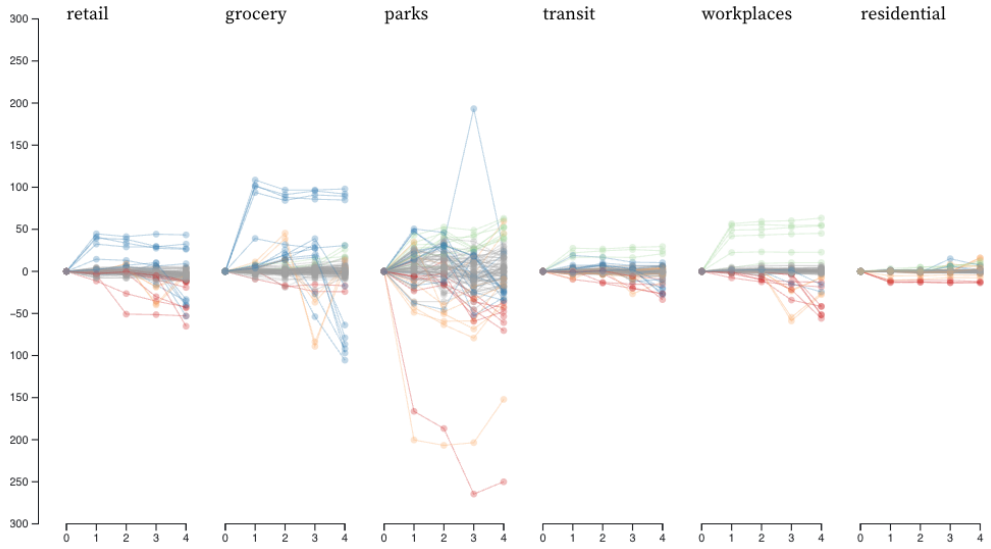


Figure 5: Actual time series with values shifted to align the start points.

T2. What is the temporal distribution of the patterns?

For this task, we propose two visualizations focusing on different types of relationships between time intervals. The timeline view (Fig. 1) focuses on the relationships of linear ordering, which are represented by positions on a straight horizontal time axis. The circular view (Fig. 6) focuses on the relationships of cyclic temporal arrangement between the episodes. In a circular chart, the years are represented by rings, and episodes (weeks of data) are blocks of the rings arranged clockwise. In both views, the temporal patterns of the individual episodes are represented by colour coding.

The timeline view (Fig. 1) reveals periods of stable mobility behaviour (i.e., prevalence of the constancy patterns) and periods of changes, in which all mobility indicators or some of them are non-constant. It shows when different patterns of the individual indicators occurred, what pattern combinations existed, and when they took place. The prevailing combination throughout the entire time span was the combination of six constancy patterns. Other combinations are rare and require more attention to be identified. For example, the combination of simultaneous down-trends of the visits of all places except homes and an up-trend of the staying at home occurred in the third week of March, when the first lockdown was issued. Similar combinations (differing by just one constituent pattern) in the Christmas periods of 2021 and 2022. These were followed by combinations of the trough in staying home and peaks in visiting all place categories except for parks.

This re-occurrence of similar patterns at the ends of two years can also be noticed by looking at the circular charts (Fig. 6). Each chart facilitates identification of seasonal and sporadic temporal patterns of a single attribute. In Fig. 6 (a), we clearly see that some temporal patterns re-occur annually. These recurrent patterns can be attributed to seasonal variations represented in the data. For example, down-trends are seen in the 'retail and recreation', 'public transport', and 'workplaces' features at the end of each year while we observe an up-trend in the 'residential' feature. We can conjecture that people travel less and prefer to stay home in the Christmas season. While the circular charts are good in revealing periodic repetitions of single-feature patterns, detection of re-occurring combinations requires integrating information from six charts; hence, holistic perception of pattern combinations is not supported by this representation. The timeline view, on the opposite, supports holistic perception of combinations but does not show periodicity as clearly as the circular charts.

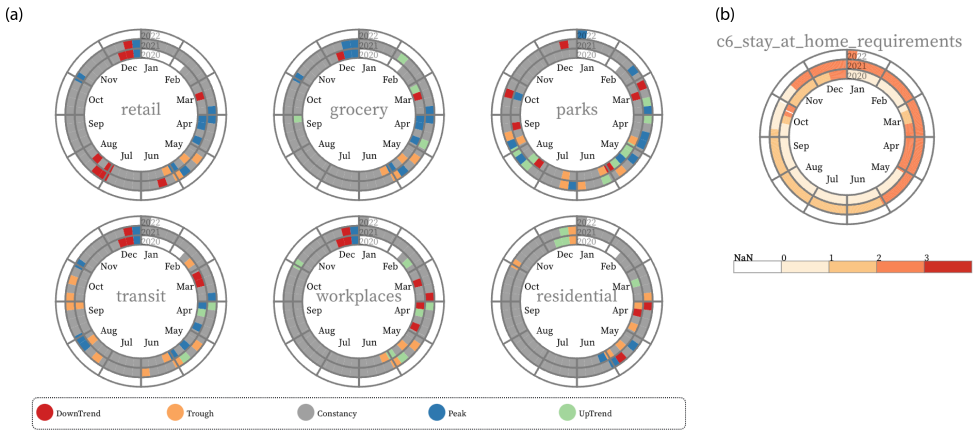


Figure 6: Circular displays of temporal patterns for features in Google Mobility Data (a) and stay-at-home requirements level in Germany (b). In all plots inner ring represents year 2020, middle - 2021, outer ring - 2022. Values in (b) mean 0: no measures announced, 1: recommended not leaving house, 2: required not leaving house with exceptions for daily exercise, grocery shopping, and 'essential' trips, 3: required not leaving house with minimal exceptions (e.g. allowed to leave once a week, or only one person can leave at a time, etc), NaN: no data [19]

The circular charts also help to detect sporadic occurrences of temporal patterns, which may be caused by factors or events that do not occur regularly. For instance, the German government required closing (or working from home) for some sectors or categories of workers. Fig. 6 (b) shows that the stay-at-home requirement level goes from 0 (no measures) to 2 (require not leaving house with exceptions for daily exercise, grocery shopping, and 'essential' trips) in the middle of March, 2020. In Fig. 6, as well as in Fig. 1, we see the effect of this measure: the residential category shows an up-trend at this time while the others have a down-trend. Moreover, we also see that the 'grocery and pharmacy' category has an up-trend in the week before the down-trend, which suggests that people went to groceries to stockpile products of everyday use (e.g., food and toilet paper) in preparation for the forthcoming restrictions or possible good shortages.

T3. Are there frequent transitions between univariate temporal patterns over sequential times?

We create bipartite graphs to represent transitions of univariate temporal patterns between consecutive time intervals. It helps to find patterns of

temporal succession and adjacency between the same and different temporal patterns of feature variation. In Fig. 7, there are six bipartite graphs, one per feature, consisting of three components: two proportionally segmented bars and curved lines linking the bar segments. The segmented bars show the overall proportions of the occurrences of the different patterns in the episodes. The segments are painted in the colours corresponding to the patterns using the same encoding as in the timeline view and the circular charts. The opacity and the stroke width of the linking lines represent the frequency of the transitions between the classes of the temporal patterns represented by the bar segments.

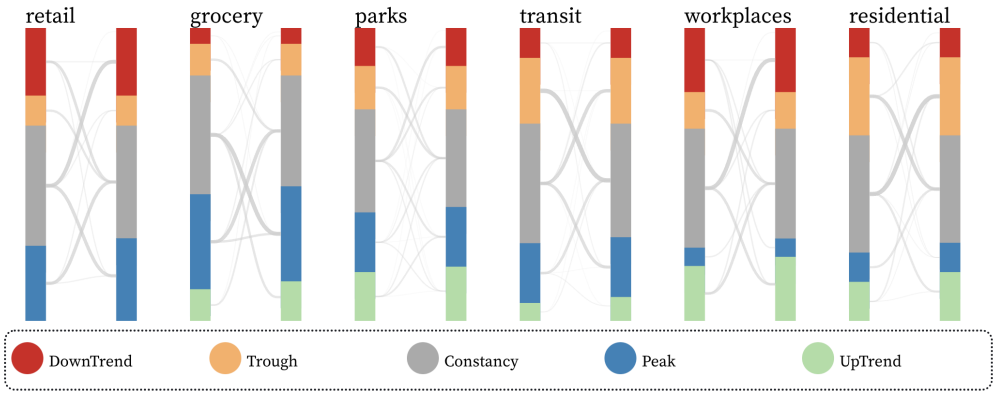


Figure 7: Bipartite graph of transitions between different patterns.

This representation can be interactively modified for focusing on selected patterns only. For example, most frequent transitions between constancy patterns are subject to be omitted for the sake of better visibility of the other transitions.

T4. Which patterns frequently co-occur?

To answer this question, we build a co-occurrence network, where nodes represent the temporal patterns of the features and edges connect patterns of different features that co-occur in the same episodes (Fig. 8, left). The size of a node represents the frequency of the temporal patterns appearing in the dataset and the opacity and the stroke width of an edge represent the frequency of co-occurrence. For example, we see that the decreasing pattern of visiting residential places and the increasing pattern of visiting workplaces frequently co-occur with the increase of the use of transit station. Note that,

same as in the transition graph, the co-occurrence between two constancy patterns is obvious and therefore omitted from the chart.

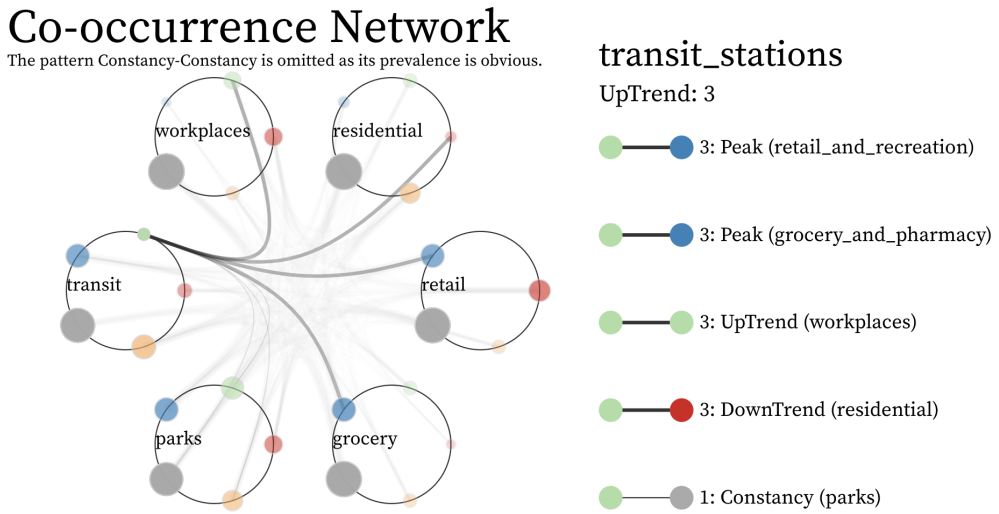


Figure 8: Co-occurrence network with the up-trend of 'transit stations' highlighted.

An analyst can interactively select a node in the network for displaying the most frequent co-occurrences of the respective temporal pattern with the temporal patterns of the other features. This interactive exploration reduces clutter in the chart and facilitates finding important relationships. Thus, the right part of Fig. 8 demonstrates the effect of selecting the node representing the up-trend pattern of 'transit stations'. It shows that this pattern occurred only three times in our data set, and in all cases it occurred together with the peak pattern of 'retail and recreation' and 'grocery and pharmacy', the up-trend pattern of 'workplaces', and the down-trend of 'residential'. This reveals a re-occurring multivariate temporal pattern (i.e., a combination of univariate patterns) in the data set.

4. Case Study: Teams' behaviours in football

We demonstrate the generality of our approach by applying it to episodes around ball possession change from a professional football (or soccer) match. Different types of changes of possession exist in football, each of which forces both teams to switch their tasks from attacking to defending or vice versa.

The team can apply different tactics. For example, after regaining the possession possible options are either to approach the opponent's goal (i.e., execute a counter-attack) or to remain at own side to protect the possession.

While different types of transitions are typically visible to a human eye, experts such as video analysts often have to watch the game to sub-categorize the transitions (e.g., label them as counter-attacks or securing the possession), which is a time-consuming and daunting task. Our intention in this study is to investigate which multivariate temporal patterns appear in transition episodes in football. We characterize these episodes by spatial features of collective movement.

Data Description

We extract episodes from positional data of players in one professional football match. We choose time intervals based on the occurrence of a specific event, i.e., change of possession. Each time interval consists of players' positions for ten seconds around transitions and the change of possession occurs exactly in the middle of the episode. As a consequence, we acquire 115 episodes, each lasting 10 seconds (i.e., 250 timesteps, given that the raw data has a sampling rate of 25 Hz), with 63 episodes seeing the home team gaining the possession and 52 episodes featuring the away team. Next, we characterize time intervals by spatial features that can be computed from positional data: compactness of the team, distance from their own goal, and velocity. For each team, we compute **team width** (i.e., distance perpendicular to the side line, between the most left-positioned field player and the most right-positioned one), **team depth** (i.e., distance parallel to the side line, between the farthest player from the goal and the nearest one, except the goalkeeper), and **distance from the center of the team to the own goal**. Since we observe a strong correlation of average velocities between players of the two teams, we calculate a common **average velocity** for the 20 infield players of both teams.

4.1. T1. What are the major patterns of individual attritbutes?

As Fig. 9 shows in grey, episodes consisting of many time steps (250 in our case) may have complex temporal patterns consisting of multiple fundamental patterns. As discussed in T1 in Section 3, temporal patterns need to be sufficiently simple to allow easy interpretation. Complex patterns can be simplified by omitting excessive details, which can be achieved through downsampling of the time series. We use the same algorithm [39] introduced

in T1 to downsample the episodes. The red dot lines in Fig 9 show how time series with 250 timesteps are downsampled into 5 timesteps. We begin by applying the downsampling technique to each half of the episode in order to get a representative value that would form the greatest triangle with two ends in the divided half. Then, using our algorithm on the downsampled episode, we classify temporal patterns. Fig. 10 illustrates all downsampled time series with colors.

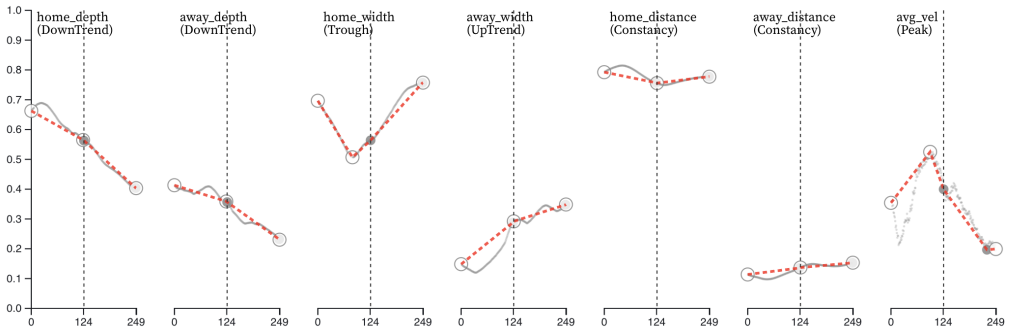


Figure 9: An overlay of the downsampled time series and the original time series for the home depth, away depth, home width, away width, home distance to their goal, away distance to their goal, and average velocity. The larger three points including the start and end points indicate the points used to classify the temporal pattern, and together with the other smaller two points they form the downsampled time series. The downsampled time series is colored to indicate the classified temporal pattern while the original time series is grey. The dotted line in the middle means the middle point of the episode. Time is shown on the horizontal axis in frames (1/25th of a second) while the normalized attribute values ranging from 0 to 1 are shown on the vertical axis.

Table 2: Frequency of temporal patterns for each feature in the football data set. Two numbers in each cell represent two types of episodes where the home team begins by defending (left) and when the away team begins by defending (right). (Peak/Trough threshold = 0.05, Constancy threshold = 0.1).

	Peak	UpTrend	Constancy	DownTrend	Trough
home_depth	14, 16	14, 13	19, 6	10, 6	6, 11
away_depth	21, 12	10, 15	12, 9	14, 2	6, 14
home_width	6, 11	10, 4	10, 6	10, 21	27, 10
away_width	11, 2	7, 13	12, 15	23, 5	10, 17
home_distance	0, 6	8, 7	33, 30	21, 9	1, 0
away_distance	1, 0	23, 10	30, 25	9, 10	0, 7
avg_vel	31, 26	5, 5	1, 0	7, 0	19, 21

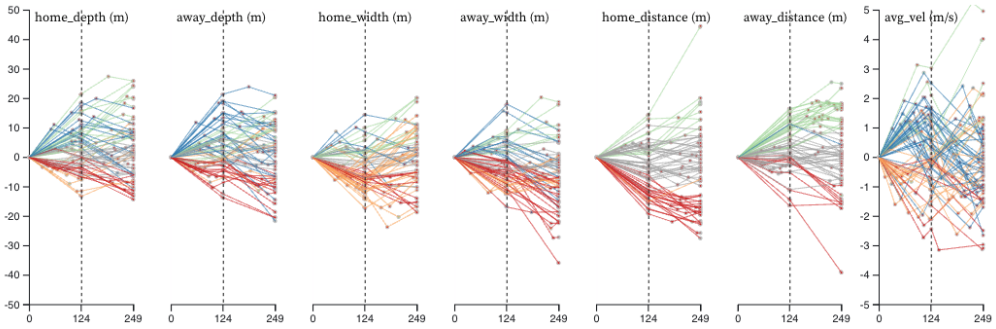


Figure 10: All downsampled time series. Colors indicate classified temporal patterns. Time is shown on the horizontal axis in frames (1/25th of a second) while the changes of the attribute values with regard to the initial point are represented by the vertical positions. The axes are labelled according to the measurement units of the original (not normalized) attributes.

Table 2 summarizes the detected patterns. For the attribute `home_width`, we observe a prevalence of patterns with increase towards the episode end (i.e., up-trends and troughs) over decreasing patterns (37 vs. 16) in the episodes when the home team begins the episode by defending (left side of the cell). This means that the home team tends to expand after they gain the possession, which is a known behaviour in football [15]. We find the opposite patterns (i.e., down-trends and peaks) to be the majority in `away_width` (34 out of 63). Second, we observe a similar number of up-trend patterns in `home_distance` as down-trend patterns in `away_distance`, as well as the similar number of down-trend patterns in `home_distance` as up-trend

patterns in `away_distance`. Fig. 11 confirms this finding with the centroids of both teams following similar trajectories. Third, we see most of patterns appear as peaks or trough (50 out of 63) in `avg_vel`. We can assume that the change of possession can accelerate or decelerate players abruptly rather than monotonically. Finally, the significant difference between the both teams may be the trough pattern of the distance from the goal. We observe only one trough pattern in `home_distance` while seven in `away_distance`. Different tactics, such as having the away side attempt more counter-attacks than the opponent, can account for this variation.

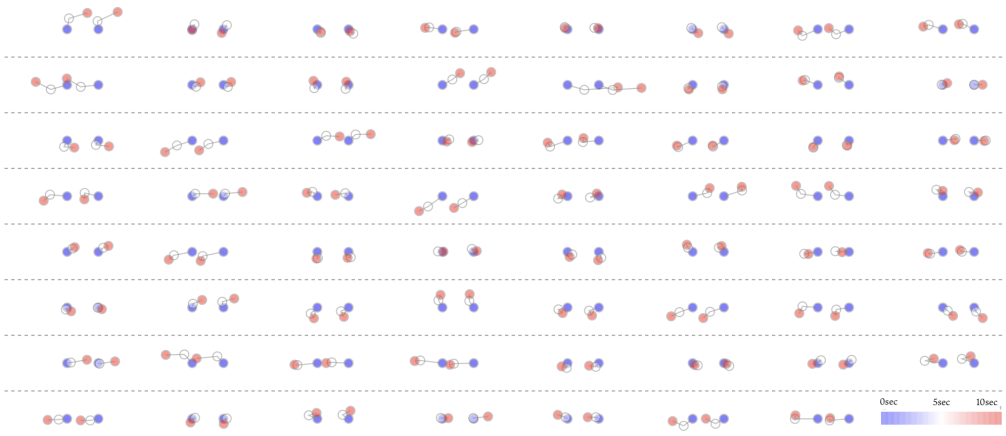


Figure 11: The team centroid shifts during the episode. Each row depicts the shift in the centroid for both teams over the course of eight episodes (left: home, right: away). The colors reflect the progression of time, from blue to white to red.

4.2. T2. What is the temporal distribution of the patterns?

We use a linear ordering to represent the temporal distribution of the temporal patterns. In Fig. 12, rectangles that represent episodes are aligned more sparsely than in Fig. 1 since the time intervals are selected according to specific events.

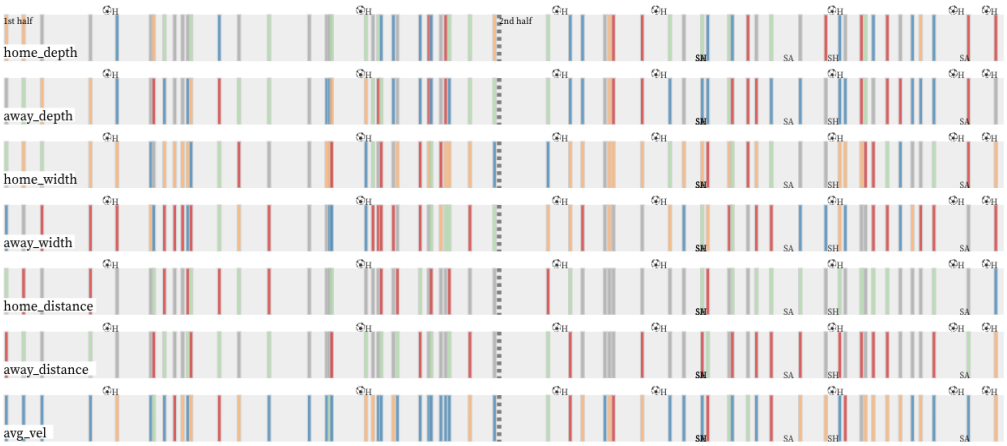


Figure 12: Timeline of temporal patterns in football data. Markers at the top and bottom of each row indicate goals and substitutions (Ball: goal, S: substitution. H: home, A: away).

Fig. 13 shows a circular view of the temporal distribution of the patterns. Two arcs in each chart represent the temporal axes, where inner arcs represent the first half of the match and outer arcs represent the second half. Although no periodic repetitions can be expected, this view can facilitate understanding the data as a circle refers to a clock face, which allows to compare the first and second halves.

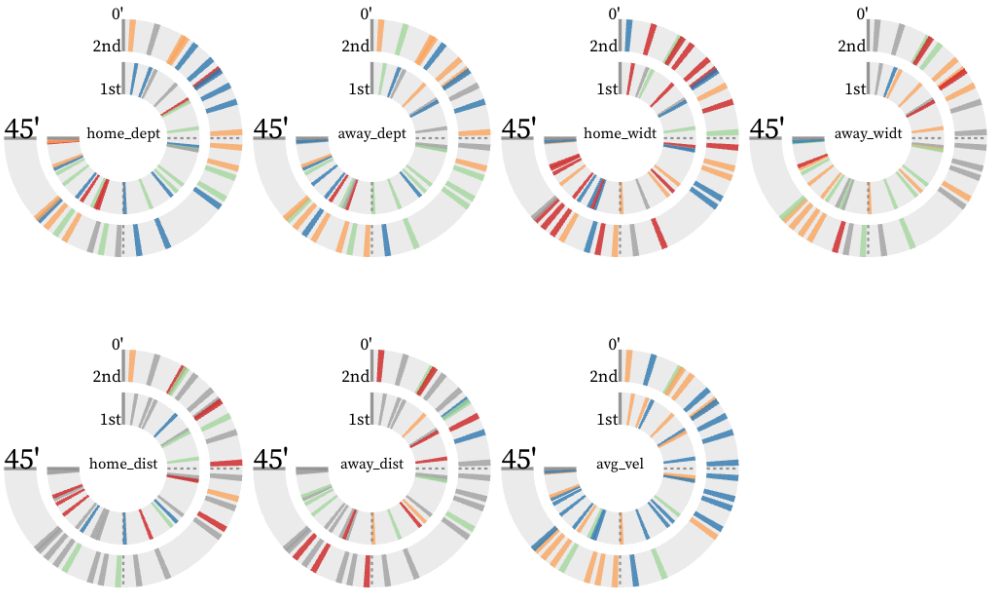


Figure 13: Circular time view of temporal patterns in football data.

4.3. T3. Are there frequent transitions between univariate temporal patterns over sequential times?

This task is not applicable to this dataset since episodes appear sporadically.

4.4. T4. Which patterns frequently co-occur?

Fig. 15 shows a co-occurrence network applied to the episodes (left) and the five multivariate temporal patterns that most frequently co-occur with the up-trend pattern of `home_distance` (right), where the home team gains the ball possession in the middle (at 5 seconds).

One third of the patterns with increasing `home_width` toward the episode end (i.e., up-trends and troughs) co-occur with the combination of `avg_vel`'s peak, `home_distance`'s down-trend, and `away_distance`'s up-trend. Fig 14 illustrates the movement of the team centroids in these episodes. We further identify from the footage that the defending team slowly rebuilds the attack after collecting long balls deep in their own side. Other combinations such as with `avg_vel`'s trough or `home_distance`'s up-trend mainly consist of counter-attacks, collecting balls relatively near to the opponent's goal, or

immediate regains of possession by the defending team. Similar tendency is found in the co-occurrence of increasing patterns of `away_width` when the away team is defending. However, we observe more counter-attacks with `avg_vel`'s trough (21% vs 14%), which implies that the away team tends to attack fast after they gain the possession.

The fact that the home side finished the season in the top three and the other team in the relegation zone explains these distinct tactics. While the away side may have preferred long balls to possession, the home team may have felt secure in controlling the ball against the opponent.

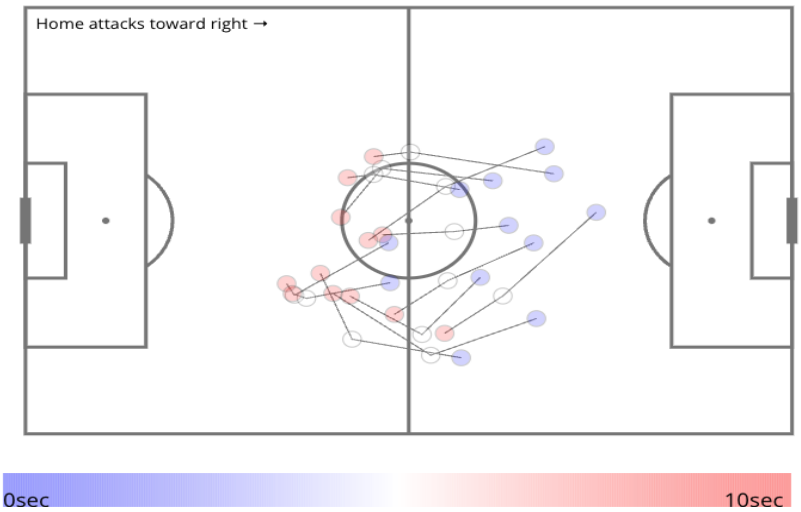


Figure 14: Movement of centroids during episodes with `avg_vel`=peak, `home_distance`=down-trend, and `away_distance`=up-trend when the home team is defending

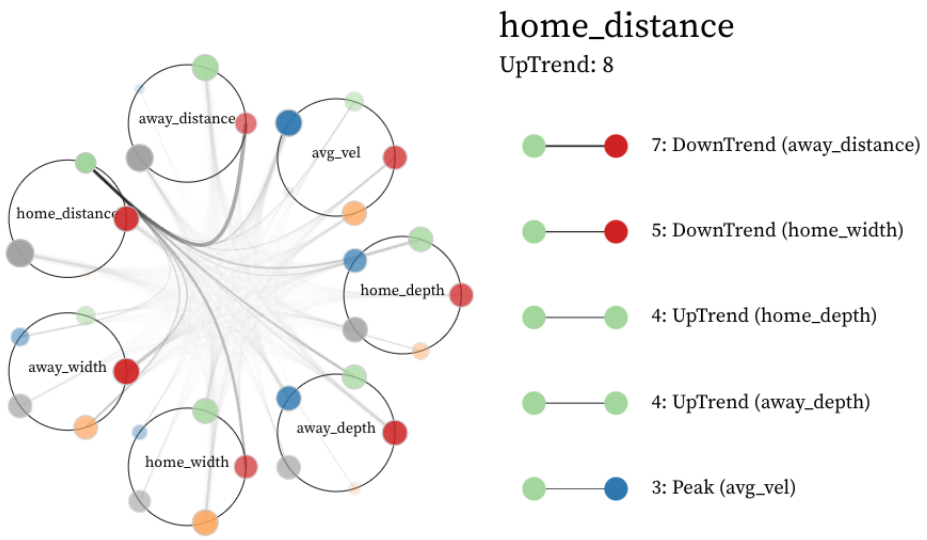


Figure 15: Features that have co-occurrence with the up-trend pattern of 'home_distance' (left) and the top 5 co-occurrent features (right) in episodes from Match 1, where the home team defends.

4.5. Summary of findings

Our approach enabled us to identify similar and distinctive behaviours for the two teams. Temporal patterns show players often play wide when they are attacking and narrow when they are defending. Additionally, quick acceleration and deceleration in response to a change of possession is observed. The co-occurrence chart reveals two typical tactics used by both sides when they gaining possession of the ball: either executing counter-attacks or gradually rebuilding the attack. After obtaining possession of the ball, the home team often carefully connects passes while the away team typically attempts quick counter-attacks.

5. Discussion

With this paper, we are proposing a view of time-varying phenomena as a sequence of episodes, i.e., time intervals encapsulating fragments of the temporal behaviours of the phenomena. The term "behaviour" here refers to any kinds of changes. Episodes can be described by values of multiple attributes specified for different time slices within the intervals and thus forming multivariate time series. The rationale for introducing episodes as

units of behaviour is that they can be short enough to allow abstractive perception and representation of each time series as a single easily interpretable temporal pattern. Hence, the behaviour encapsulated in an episode can be represented by a combination of patterns made by the multiple attributes.

Based on the premise that simplification and abstraction are essential for understanding a phenomenon, i.e., building a mental model of it [7], we explored in our research the analytical potential of computer-supported abstraction of time series to temporal patterns and explicit representation of these patterns for involvement in subsequent analysis. The idea is that the patterns substitute the original elementary data [6] and are themselves treated as data to be analysed. We considered several analysis tasks that can be posed when dealing with such data and defined visual analytics techniques that can support these tasks.

In our exploratory study, we neither tried to create a complete task taxonomy for analysis of temporal patterns of episodes nor strove to design novel visualisations. The goal was to investigate the principal possibility of analysing data transformed into temporal patterns. Our study showed that this approach can be quite useful. By using abstractions of elementary data, it allows considering the behaviour of a phenomenon at a yet higher level of abstraction, namely, at the level of relationships between the patterns. This contributes to obtaining an overall understanding of the behaviour or revealing its essential features. It can be noted that the very idea of the approach is generic, i.e., potentially applicable to any type of data.

Given that transformation of data to patterns can be beneficial, a valid question is what kinds of patterns should be considered and how to obtain them from data. This question requires a specific answer for each distinct type of data, because patterns are formed by type-specific intrinsic relationships between data elements [6]. We have proposed an answer to this question for data consisting of time series of values of numeric attributes. We wanted to represent such data by patterns that are well understood by humans and, preferably, denoted by commonly understandable terms. We considered a set of basic patterns that can be represented graphically as particular geometric shapes and are commonly labelled as up-trend, peak, constancy, trough, and down-trend. We propose an algorithm for automatic recognition of these patterns and representation of episodes by combinations of patterns. We acknowledge the possibility to consider other sets of patterns requiring other algorithms for extraction, but we would like to note that the same visualisation and exploration techniques may be applied to transformed

data regardless of the specific pattern "vocabularies" used for encoding the data.

The visual analytics techniques that we described in this paper are intended to support exploration of (A) the temporal distribution of the different types of patterns and (B) relationships between the temporal patterns, namely, temporal ordering of patterns in a sequence of episodes and co-occurrence of patterns within episodes. A and B are the two major classes of analytical tasks relevant to time-referenced data in general. The most common representation of such data is by some kind of visual marks along a time axis, and we apply it in our timeline view. A circular representation of time is also frequently used, particularly, to reveal and explore cyclic changes. We also propose two time-abstracted and aggregated representations of the data in the form of graphs showing sequential ordering relationships between patterns of the same attribute and co-occurrence relationships between patterns of different attributes. Using graphs to visualise relationships is one of the most common design choices along with the use of a time axis-based display to visualise a temporal distribution. The visualisations we describe in the paper should be considered as mere examples of numerous possible implementations of these fundamental designs.

Thus, there are many methods for laying out nodes of a graph [17]. Most of the existing algorithms are not suitable for visualising relationships between patterns, which requires the nodes representing the patterns of the same attribute to be grouped together and separated from nodes referring to other attributes. We address this requirement in our design of the co-occurrence network (Figs. 8 and 15) by arranging groups of nodes in circles. A more usual design that could satisfy this requirement is the chord diagram [23] using a circular layout, where groups of nodes are arranged in arcs and separated from other groups by gaps. Figure 16 demonstrates how the same data as in Figs. 8 and 15 can be visualised in the form of chord diagrams. In our design, the grouping of nodes is much better noticeable than in a chord diagram. A disadvantage of our design is intersections between some of the graph edges and the circles that visually link nodes belonging to groups. The circular layout, as in a chord diagram, is potentially suitable for visualising hierarchical networks by increasing the number of outer circles; however, this is not needed in our case. The circular layout may also be more scalable to a greater number of nodes given its simple structure; however, showing a large number of node groups with sufficient separation between them may be problematic. Since there is no universally effective

layout, the choice should depend on properties of data and user preferences.

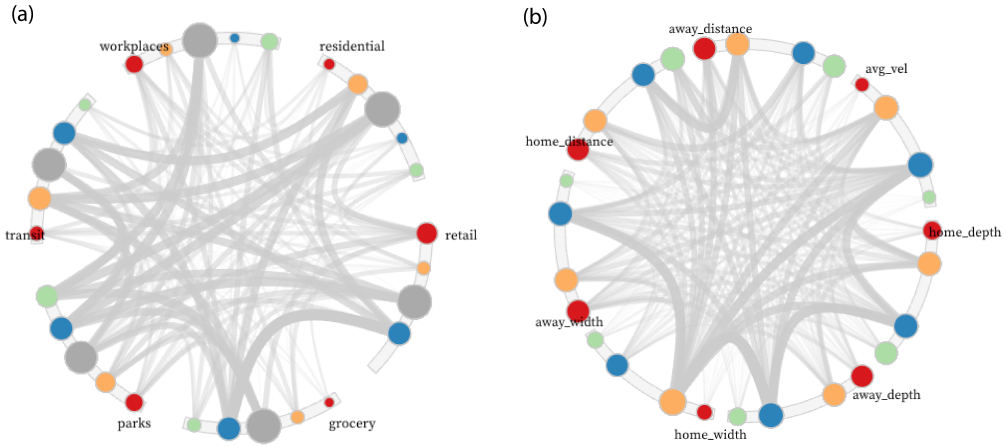


Figure 16: An alternative design of co-occurrence chart with two data sets: (a) Google Mobility Data (2) Football Data

We consider our work as just a first step in the research on analysis of episodes as a way of representing complex dynamic phenomena. We envisage continuous systematic research in this direction. Our exploratory study shows how this representation can be utilised leveraging the possibility of condensing and abstracting elementary data. While we see that this approach has good potential, we admit that the set of techniques we have developed is not yet sufficiently powerful. In particular, it provides quite limited opportunities for exploration of multi-attribute temporal patterns, i.e., combinations of single-attribute patterns. The co-occurrence network shows only pairwise co-occurrence relationships but does not support joint perception and analysis of multiple patterns occurring together in episodes. We see the problem of representing and analysing multivariate temporal patterns as a challenge for future research that requires significant attention and concentration of effort.

Hence, one of the next steps in the future research should be towards finding methods for the integration of multiple single-attribute temporal patterns into composite multi-attribute patterns that can be perceived and treated as units. We see a possibility to achieve this goal with the help of topic modelling. Our experiments [38] showed that this idea deserves further investigation. Another step should be towards methods for comprehensive analysis of

temporal relationships between patterns not limited to co-occurrence and sequential ordering. Our initial idea is to consider temporal neighbourhoods of patterns and try to find re-occurring combinations of patterns whose neighbourhoods overlap.

6. Conclusion

We have introduced a concept of episode as a relatively short fragment in temporal development or behaviour of a dynamic phenomenon. We have suggested that data describing episodes may have the form of time series of values of multiple attributes. Limiting our focus to numeric attributes, we have presented an approach to analysis of such data by means of automated abstraction of the time series to temporal patterns represented as categorical labels. We have demonstrated possible ways of visualising abstracted data for analysing the temporal distribution of the patterns and relationships between patterns within and across episodes. Our study has shown that decomposition of complex behaviours into episodes and characterising episodes by temporal patterns of multiple attributes is a promising approach to analysis of dynamic phenomena. We call for further research in this direction, particularly, to find ways to consider and analyse combinations of single-attribute patterns holistically as integrated patterns incorporating multiple aspects of the behaviour.

7. Acknowledgements

This work was partly supported by Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the *Lamarr Institute for Machine Learning and Artificial Intelligence* (Lamarr22B), by EU in projects *SoBigData++* and *CrexData*, and by DFG within priority research program *SPP VGI* (project *EVA-VGI*).

References

[1] Aigner, W., Miksch, S., Schumann, H., Tominski, C., 2011. Visualization of time-oriented data. Springer. doi:10.1007/978-0-85729-079-3.

[2] Allen, J.F., Ferguson, G., 1994. Actions and events in interval temporal logic. J. Logic Comput. 4, 531–579.

[3] Andrienko, G., Andrienko, N., Anzer, G., Bauer, P., Budziak, G., Fuchs, G., Hecker, D., Weber, H., Wrobel, S., 2021a. Constructing spaces and times for tactical analysis in football. IEEE Transactions on Visualization and Computer Graphics 27, 2280–2297. doi:10.1109/TVCG.2019.2952129.

[4] Andrienko, N., Andrienko, G., Camossi, E., Claramunt, C., Cordero-Garcia, J.M., Fuchs, G., Hadzagic, M., Jousselme, A.L., Ray, C., Scarlatti, D., Vouros., G., 2017. Visual exploration of movement and event data with interactive time masks. Visual Informatics 1, 25 – 39. doi:https://doi.org/10.1016/j.visinf.2017.01.004.

[5] Andrienko, N., Andrienko, G., Fuchs, G., Slingsby, A., Turkay, C., Wrobel, S., 2020. Visual Analytics for Data Scientists. Springer.

[6] Andrienko, N., Andrienko, G., Miksch, S., Schumann, H., Wrobel, S., 2021b. A theoretical model for pattern discovery in visual analytics. Visual Informatics 5, 23–42. doi:https://doi.org/10.1016/j.visinf.2020.12.002.

[7] Andrienko, N., Lammarsch, T., Andrienko, G., Fuchs, G., Keim, D., Miksch, S., Rind, A., 2018. Viewing visual analytics as model building. Computer Graphics Forum 37, 275–299. doi:10.1111/cgf.13324.

[8] Bale, K., Chapman, P., Barraclough, N., Purdy, J., Aydin, N., Dark, P., 2007. Kaleidomaps: A new technique for the visualization of multivariate time-series data. Information Visualization 6, 155–167. doi:10.1057/palgrave.ivs.9500154.

[9] Bernard, J., Dobermann, E., Bögl, M., Röhlig, M., Vögele, A., Kohlhammer, J., 2016. Visual-interactive segmentation of multivariate time series, in: Proceedings of the EuroVis Workshop on Visual Analytics, Eurographics Association, Goslar, DEU. p. 31–35.

[10] Bernard, J., Wilhelm, N., Scherer, M., May, T., Schreck, T., 2012. Timeseriespaths : Projection-based explorative analysis of multivariate time series data, in: Journal of WSCG, pp. 97–106.

[11] Buono, P., 2016. A circular visualization technique for collaboration and quantifying self, in: Proceedings of the International Working Con-

ference on Advanced Visual Interfaces, Association for Computing Machinery, New York, NY, USA. pp. 348–349.

[12] Collins, C., Andrienko, N., Schreck, T., Yang, J., Choo, J., Engelke, U., Jena, A., Dwyer, T., 2018. Guidance in the human—machine analytics process. Visual Informatics 2, 166 – 180. doi:10.1016/j.visinf.2018.09.003.

[13] Crisan, A., Fisher, S.E., Gardy, J.L., Munzner, T., 2021. GEViTRec: Data reconnaissance through recommendation using a Domain-Specific visualization prevalence design space. IEEE Trans. Vis. Comput. Graph. PP.

[14] Das, G., Lin, K.I., Mannila, H., Renganathan, G., Smyth, P., 1998. Rule discovery from time series., in: KDD, pp. 16–22.

[15] Fonseca, S., Milho, J., Travassos, B., Araujo, D., Lopes, A., 2013. Measuring spatial interaction behavior in team sports using superimposed voronoi diagrams. Int. J. Perform. Anal. Sport 13, 179–189.

[16] Gharghabi, S., Yeh, C.C.M., Ding, Y., Ding, W., Hibbing, P., LaMunion, S., Kaplan, A., Crouter, S.E., Keogh, E., 2019. Domain agnostic online semantic segmentation for multi-dimensional time series. Data mining and knowledge discovery 33, 96–130.

[17] Gibson, H., Faith, J., Vickers, P., 2013. A survey of two-dimensional graph layout techniques for information visualisation. Information Visualization 12, 324–357. doi:10.1177/1473871612455749.

[18] Google, 2022. COVID-19 community mobility reports.

[19] Hale, T., Angrist, N., Goldszmidt, R., Kira, B., Petherick, A., Phillips, T., Webster, S., Cameron-Blake, E., Hallas, L., Majumdar, S., Tatlow, H., 2021. A global panel database of pandemic policies (oxford COVID-19 government response tracker). Nat Hum Behav 5, 529–538.

[20] Hao, M., Marwah, M., Janetzko, H., Sharma, R., Keim, D.A., Dayal, U., Patnaik, D., Ramakrishnan, N., 2011. Visualizing frequent patterns in large multivariate time series, in: Visualization and Data Analysis 2011, SPIE. pp. 194–203. doi:10.1117/12.872169.

[21] Haroz, S., Kosara, R., Franconeri, S.L., 2016. The connected scatterplot for presenting paired time series. IEEE Trans. Vis. Comput. Graph. 22, 2174–2186.

[22] Heer, J., Kong, N., Agrawala, M., 2009. Sizing the horizon: the effects of chart size and layering on the graphical perception of time series visualizations, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA. pp. 1303–1312. URL: http://idl.cs.washington.edu/papers/horizon.

[23] Holten, D., 2006. Hierarchical edge bundles: visualization of adjacency relations in hierarchical data. IEEE Trans. Vis. Comput. Graph. 12, 741–748.

[24] Janetzko, H., Sacha, D., Stein, M., Schreck, T., Keim, D.A., Deussen, O., 2014. Feature-driven visual analytics of soccer data, in: 2014 IEEE Conference on Visual Analytics Science and Technology (VAST), pp. 13–22. doi:10.1109/VAST.2014.7042477.

[25] Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S., 2001. Dimensionality reduction for fast similarity search in large time series databases. Knowl. Inf. Syst. 3, 263–286.

[26] Keogh, E., Lin, J., Lee, S.H., Van Herle, H., 2007. Finding the most unusual time series subsequence: algorithms and applications. Knowl. Inf. Syst. 11, 1–27.

[27] Lee, T.Y., Shen, H.W., 2009. Visualization and exploration of temporal trend relationships in multivariate time-varying data. IEEE transactions on visualization and computer graphics 15, 1359–66. doi:10.1109/TVCG.2009.200.

[28] Lin, J., Keogh, E., Wei, L., Lonardi, S., 2007. Experiencing SAX: a novel symbolic representation of time series. Data Min. Knowl. Discov. 15, 107–144.

[29] Liu, L., Mei, S., 2016. Visualizing the GVC research: a co-occurrence network based bibliometric analysis. Scientometrics 109, 953–977.

[30] Lu, Y., Wu, R., Mueen, A., Zuluaga, M.A., Keogh, E., 2022. Matrix profile XXIV: Scaling time series anomaly detection to trillions of datapoints and ultra-fast arriving data streams, in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA. pp. 1173–1182.

[31] Mauceri, S., Sweeney, J., Nicolau, M., McDermott, J., 2021. Feature extraction by grammatical evolution for one-class time series classification. Genet. Program. Evolvable Mach. 22, 267–295.

[32] Peuquet, D.J., 1994. It's about time: A conceptual framework for the representation of temporal dynamics in geographic information systems. Annals of the Association of american Geographers 84, 441–461.

[33] Pham, V.V., Nguyen, N.V.T., Li, J., Hass, J., Chen, Y., Dang, T., 2019. Mtsad: Multivariate time series abnormality detection and visualization. 2019 IEEE International Conference on Big Data (Big Data) , 3267–3276.

[34] Qiang, Y., Delafontaine, M., Versichele, M., De Maeyer, P., Van de Weghe, N., 2012. Interactive analysis of time intervals in a two-dimensional space. Inf. Vis. 11, 255–272.

[35] Robertson, G., Fernandez, R., Fisher, D., Lee, B., Stasko, J., 2008. Effectiveness of animation in trend visualization. IEEE Trans. Vis. Comput. Graph. 14, 1325–1332.

[36] Rosling, H., 2006. The best stats you've ever seen. https://www.ted.com/talks/hans_rosling_the_best_stats_you_ve_ever_seen. Accessed: 2021-11-5.

[37] Schreck, T., Tekušová, T., Kohlhammer, J., Fellner, D., 2007. Trajectory-based visual analysis of large financial time series data. SIGKDD Explor. Newsl. 9, 30–37. URL: https://doi.org/10.1145/1345448.1345454, doi:10.1145/1345448.1345454.

[38] Shirato, G., Andrienko, N., Andrienko, G., . What are the topics in football? Extracting time-series topics from game episodes. IEEE VIS 2021 poster URL: http://geoanalytics.net/and/papers/vis21poster.pdf.

[39] Steinarsson, S., 2013. Downsampling Time Series for Visual Representation. Ph.D. thesis. University of Iceland.

[40] Tanisaro, P., Heidemann, G., 2019. Dimensionality reduction for visualization of time series and trajectories, in: Felsberg, M., Forssen, P., Sintorn, I., Unger, J. (Eds.), Image Analysis, Springer International Publishing. pp. 246–257. doi:10.1007/978-3-030-20205-7_21.

[41] Tominski, C., Schumann, H., 2020. Interactive Visual Data Analysis. CRC Press.

[42] Van Wijk, J.J., Van Selow, E.R., 1999. Cluster and calendar based visualization of time series data, in: Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99), pp. 4–9.

[43] Van de Weghe, N., Docter, R., De Maeyer, P., Bechtold, B., Ryckbosch, K., 2007. The triangular model as an instrument for visualising and analysing residuality. J. Archaeol. Sci. 34, 649–655.

[44] Yeh, C.C.M., Kavantzas, N., Keogh, E., 2017. Matrix profile VI: Meaningful multidimensional motif discovery, in: 2017 IEEE International Conference on Data Mining (ICDM), IEEE. pp. 565–574.

[45] Zhao, J., Drucker, S.M., Fisher, D., Brinkman, D., 2012. TimeSlice: interactive faceted browsing of timeline data, in: Proceedings of the International Working Conference on Advanced Visual Interfaces, Association for Computing Machinery, New York, NY, USA. pp. 433–436.

[46] Zhao, J., Forer, P., Harvey, A.S., 2008. Activities, ringmaps and geovisualization of large human movement fields. Information Visualization 7, 198–209. doi:10.1057/PALGRAVE.IVS.9500184.

Chapter 4
Exploring and visualizing temporal relations in multivariate time series

Exploring and visualizing temporal relations in multivariate time series

Gota Shirato[a,b,*], Natalia Andrienko[a,c], Gennady Andrienko[a,c]

[a] *Fraunhofer IAIS, Sankt Augustin, 53757, Germany*
[b] *University of Bonn, Regina-Pacis-Weg 3, Bonn, 53113, Germany*
[c] *City, University of London, Northampton Square, London, EC1V 0HB, UK*

Abstract

This paper introduces an approach to analysing multivariate time series (MVTS) data through progressive temporal abstraction of the data into patterns characterizing behavior of the studied dynamic phenomenon. The paper focuses on two core challenges: identifying basic behavior patterns of individual attributes and examining the temporal relations between these patterns across the range of attributes to derive higher-level abstractions of multi-attribute behavior. The proposed approach combines existing methods for univariate pattern extraction, computation of temporal relations according to the Allen's time interval algebra, visual displays of the temporal relations, and interactive query operations into a cohesive visual analytics workflow. The paper describes application of the approach to real-world examples of population mobility data during the COVID-19 pandemic and characteristics of episodes in a football match, illustrating its versatility and effectiveness in understanding composite patterns of interrelated attribute behaviors in MVTS data.

Keywords: temporal relations, temporal abstraction, multivariate time series, time intervals

*Corresponding author.
Email addresses: gota.shirato@iais.fraunhofer.de (Gota Shirato), natalia.andrienko@iais.fraunhofer.de (Natalia Andrienko), gennady.andrienko@iais.fraunhofer.de (Gennady Andrienko)

1. Introduction

Temporal abstraction means representing sequences of time-referenced data items as unified entities called patterns [4]. To comprehend the underlying dynamics and interrelationships among various attributes within multivariate time series (MVTS), it is crucial to uncover and explore temporal relations between patterns of individual attribute variations. Although numerous methods exist to address specific tasks in abstracting and analyzing MVTS, there is currently no overarching framework that consolidates these tasks and their corresponding methods into a comprehensive analysis workflow. Such a framework would help researchers to synergistically use different methods, leveraging the variety of existing techniques and enhancing their understanding of dynamic phenomena.

This paper presents a framework that aims to bridge this gap by supporting progressive abstraction of MVTS, from defining relevant intervals with basic behavioral patterns of individual attributes to exploring temporal relations between previously extracted patterns, which may differ in their levels of abstraction. Our primary objective is not to replace existing methods with new ones; instead, we show how to organize existing methods supporting different tasks into a cohesive visual analytics workflow [3]. We present examples of computational and visualization techniques capable to support different analysis steps, while the framework is conceptual and therefore allows the use of any appropriate methods. Thus, analysts can choose the types of patterns to search for, pick one of suitable existing methods that can detect these patterns in time series, and choose or design a time-oriented visualization technique that will show the positions of the detected patterns along the time line. To visualize relationships between the patterns, analysts may use node-link diagrams instead of matrices.

The proposed framework addresses two key problems: (a) identifying basic behavioral patterns of individual attributes, and (b) examining the temporal relations between these patterns across multiple attributes to derive higher-level abstractions. In this context, a basic pattern refers to an interpretable symbol or expression representing a sequence of values for a single variable, such as "increasing" or "decreasing" [28]. Deriving complex patterns of joint behavior of multiple variables is particularly challenging, especially when there are time lags between the starting points of individual behavior patterns. Our framework is designed to visualize temporal relations with lags, facilitating the comprehension of complex interactions among mul-

tiple attributes.

The framework focuses on three main tasks:

- T1: Defining relevant intervals with basic patterns in univariate time series
- T2: Deriving complex patterns by computing temporal relations between time intervals in multivariate time series
- T3: Exploring occurrence patterns of temporal relations through an interactive visual interface.

For T1, we apply an algorithm to define relevant time intervals from univariate time series using the geometric pattern extraction technique [30]. T2 can be fulfilled by using Allen's interval algebra [1]. T3 is supported by a visual exploration interface designed following B.Shneiderman's mantra of "Overview first, zoom and filter, then details-on-demand" [31].

We argue that these tasks serve as essential components in a comprehensive MVTS analysis workflow, demonstrating the cohesive nature of our framework. Our framework accommodates various types of patterns and temporal relationships, allowing analysts to apply existing methods for pattern detection. Example of such methods include trend-based [18] and state-based techniques [20].

The rest of this paper is structured as follows. Section 2 discusses the related work. Section 3 describes selected methods suitable for each task using the example of the mobility data during the COVID-19 pandemic. Section 4 demonstrates the effectiveness and versatility of our framework by example of another application using football (soccer) data. Section 5 discusses the concept, approaches, and answered research questions, identifies strengths and limitations, and proposes directions for future work. Finally, Section 6 concludes our work.

2. Related Work

In this section, we review the literature related to the analysis of multivariate time series, temporal abstraction, and visualization techniques for exploring temporal relations.

2.1. Multivariate Time Series Analysis

A variety of methods have been proposed for the analysis of multivariate time series (MVTS) data. These methods can be broadly categorized into statistical approaches (e.g., Granger causality [10], vector autoregression [22]), machine learning techniques (e.g., recurrent neural networks [12], Bayesian networks [26]), and matrix and tensor factorization methods [16]. While these approaches are effective in modeling and predicting various aspects of MVTS data, they often do not provide an intuitive understanding of the temporal relations between different attributes.

2.2. Temporal Abstraction

Temporal abstraction involves transforming the raw data into higher-level concepts that are easier to understand and interpret, thus creating interval-based representations from time-stamped data [29] (i.e., basic temporal abstraction [28]), and abstracting intervals into other intervals with a higher level of abstraction (i.e., complex temporal abstraction [28]). Techniques like time series segmentation [14], time periodization [2], motif discovery [27], and frequent episode mining [24] have been used to identify meaningful patterns in univariate and multivariate time series. Joint behavior of multiple variables are also derived from basic patterns by using co-occurrence [19] and simultaneity of different temporal patterns [30]. While these methods are effective in extracting temporal patterns, they do not explicitly address the problem of exploring and analyzing different types of temporal relations between the identified patterns.

2.3. Visualization Techniques for Temporal Relations

Several visualization techniques have been proposed to explore temporal relations in time series data. TimeMatrix [35] and TimeNotes [33] are examples of visualizations that present the temporal relations between events in the form of a matrix. Moreover, co-occurrences of pairs of different abstract patterns can be visualized by a network [30]. EventFlow [25] and Outflow [34] are visual analytics tools that enable users to explore temporal patterns in event sequences by providing interactive visualizations of event data. Techniques have also been introduced for visually specifying, combining, and querying complex temporal patterns [6]. Recent work supports the validation of causal relationships by showing correlations between time intervals in matrix [17] and network [7, 21]. An approach exists for investigating relationships between one or more time series within specified time

frames [36]. However, these methods generally lack direct support for exploring patterns of temporal relations that include time lags.

In summary, the related work highlights a large variety of methods and techniques available for analyzing and visualizing multivariate time series data and temporal relations. Our proposed framework distinguishes itself by introducing an integrated workflow consisting of three tasks: identifying relevant intervals containing patterns, deriving complex patterns by computing temporal relations, and exploring occurrence patterns of temporal relations through an interactive visual interface . We propose a selection of methods that can be employed for each task but do not exclude the use of alternative techniques.

3. Visual analytics approach

In this section, we present a workflow composed of methods suitable for identifying time intervals with basic behavior patterns, computing temporal relations between time intervals in multivariate time series, and revealing patterns of joint behavior by visually presenting the relations between earlier extracted patterns.

3.1. Essence of the approach

The key idea of our approach is to conduct a progressive abstraction process from identifying basic patterns in univariate time series to discovering higher-level patterns formed by temporal relations between the basic patterns. This process is designed to simplify and distill complex multivariate time series data into meaningful and interpretable components that can be easily understood and analyzed. The workflow of the process is presented in Figure 1.

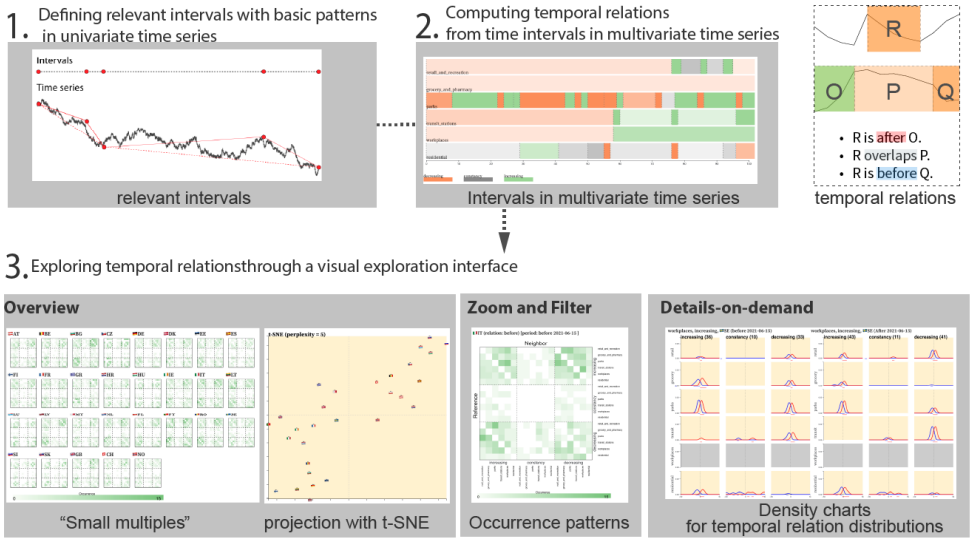


Figure 1: The workflow of the progressive abstraction process consists of three steps with increasing the level of abstraction at each step.

We demonstrate our approach by utilizing time series of continuous daily mobility indicators from Google Mobility Data [9].

3.2. Approach introduced by example

During the COVID-19 pandemic, measures were taken to curb infection spread by restricting people's movement. In mobility data, we observe temporal relations between different mobility patterns that warrant attention. When a lockdown measure is announced, for example, we can expect an `increase` pattern in coming to workplaces to occur `before` an `increase` pattern of staying at home since, many people may go to their offices to prepare for remote work. The mobility patterns may differ among different countries due to their varying policies against the pandemic. Here, our interest is to visualize temporal relations between mobility patterns within each country and to investigate the distributions of temporal relations across countries.

Data description

Following the COVID-19 outbreak in February 2020 [9], Google began publishing anonymized mobility data for six different categories of places:

`retail and recreation`, `grocery and pharmacies`, `parks`, `transit stations`, `workplaces`, and `residential` from various regions. The data consists of daily visitor counts to these categories, compared to baseline days prior to the pandemic's onset. Baseline days represent a normal value for each day of the week, calculated as the median value over a five-week period from January 3rd to February 6th, 2020. The values in the published data are presented as percentages of changes from these baseline values. For our analysis, we utilize daily time series for 29 countries across Europe, collected between the 15th of February, 2020 and the 15th of October, 2022.

T1. Defining relevant intervals containing abstract patterns

The first step in our framework involves dividing a univariate time series into a set of time intervals of varying lengths, each featuring a distinct pattern of value variation (e.g., a trend). Analysts can define the pattern types of interest and adjust thresholds for identifying them. In our example, we consider a set of basic patterns that can be represented as trends and typically labelled as `increase`, `constancy`, and `decrease`. To distinguish these trends, we employ the algorithm by Shirato et al. [30], which treats a time series as a graph of a function $V(t)$ in a Cartesian coordinate system. It determines a peak or trough pattern by identifying the largest triangle in a time interval $[t_1, t_2]$ formed by points $(t_1, V(t_1)), (t', V(t')), and (t_2, V(t_2))$, where $t_1 < t' < t_2$. In other words, the algorithm identifies a peak or trough point that forms the largest triangle with the starting and ending points of a given interval (Figure 2). If the area of the triangle is sufficiently large, the peak or trough point is taken as a break point to divide the time series into two segments each containing a simpler pattern of temporal variation that can be considered as increase, decrease, or constancy.

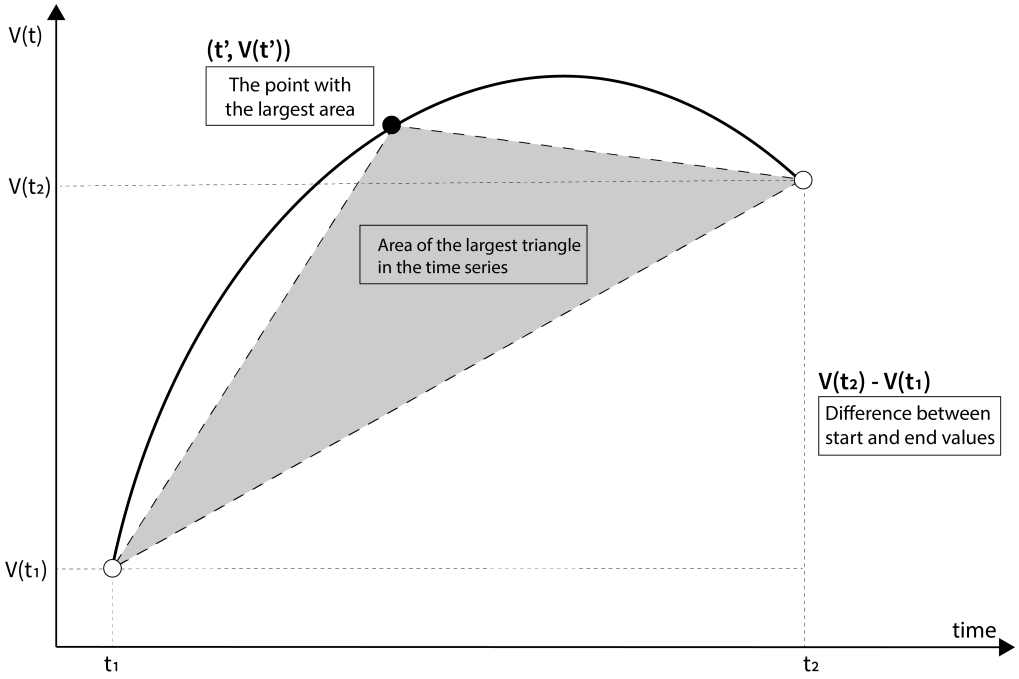


Figure 2: Illustration of the idea of the pattern detection algorithm [30]. The curve portrays the evolution of values over time as a function $V(t)$. The algorithm identifies a point $(t', V(t'))$ within an interval $[t_1, t_2]$ that forms the largest triangle with the points $(t_1, V(t_1))$ and $(t_2, V(t_2))$. The point $(t', V(t'))$ is used to divide the time series in two intervals. In this example, the first interval contains an increasing trend. Depending on a threshold for the difference between the first and last values, the attribute behavior in the second interval can be considered as a decreasing trend or as constancy.

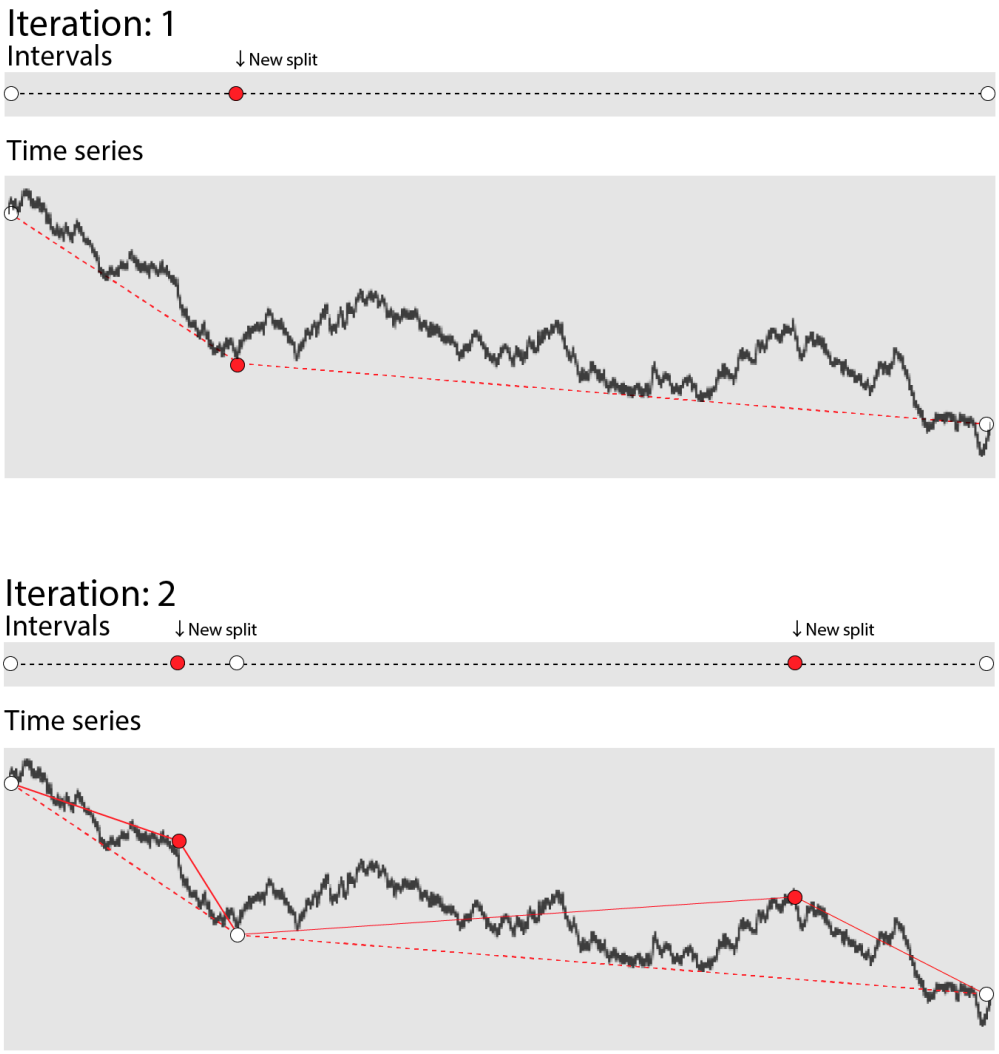


Figure 3: Dividing time intervals by iterative application of the largest triangle algorithm [30]. In the initial iteration (top), the algorithm finds the peak or trough point that forms the largest triangle in the time series, then segments the time interval at this identified time point. Subsequently, in the second iteration (bottom), the algorithm discovers peak or trough points in the time intervals obtained in the first iteration. The iterative process continues until the area of the largest triangle falls below a chosen threshold.

The operation of finding the largest triangle is applied to a time series in

an iterative manner as illustrated in Figure 3. Initially it is applied to the whole time span of the time series $[t_0, t_{last}]$ (Figure 3, top). After finding the vertex of the largest triangle $(t', V(t'))$, the time step t' is used for dividing the entire time span into intervals $[t_0, t']$ and $[t', t_{last}]$. The operation is then applied to each of these two intervals, which can be, in turn, further divided into sub-intervals (Figure 3, bottom). The decision whether a given interval needs to be subdivided depends on the area of the largest triangle on this interval. A large area implies the presence of a substantial peak or trough, which is a composition of increasing and decreasing trend patterns. Hence, the interval needs to be subdivided for obtaining simpler patterns. When the triangle is small, it suggests that this fragment of the time series can be treated as a simple trend pattern with inessential noise.

With an aim to obtain a set of elementary trends, namely `increase`, `constancy`, and `decrease`, we recursively segment each time interval until its pattern is sufficiently distinct, i.e., the largest triangle within the segment is smaller than a threshold. A larger threshold allows for larger triangles, representing more significant peaks or troughs, to exist within the segment, resulting in fewer intervals as the partitioning process is less stringent. In other words, increasing the threshold value leads to a coarser segmentation of a time series and a higher level of abstraction, where segments are considered as simple trends, and internal deviations are ignored. Figure 4 demonstrates the effect of the threshold on the final division of a time series.

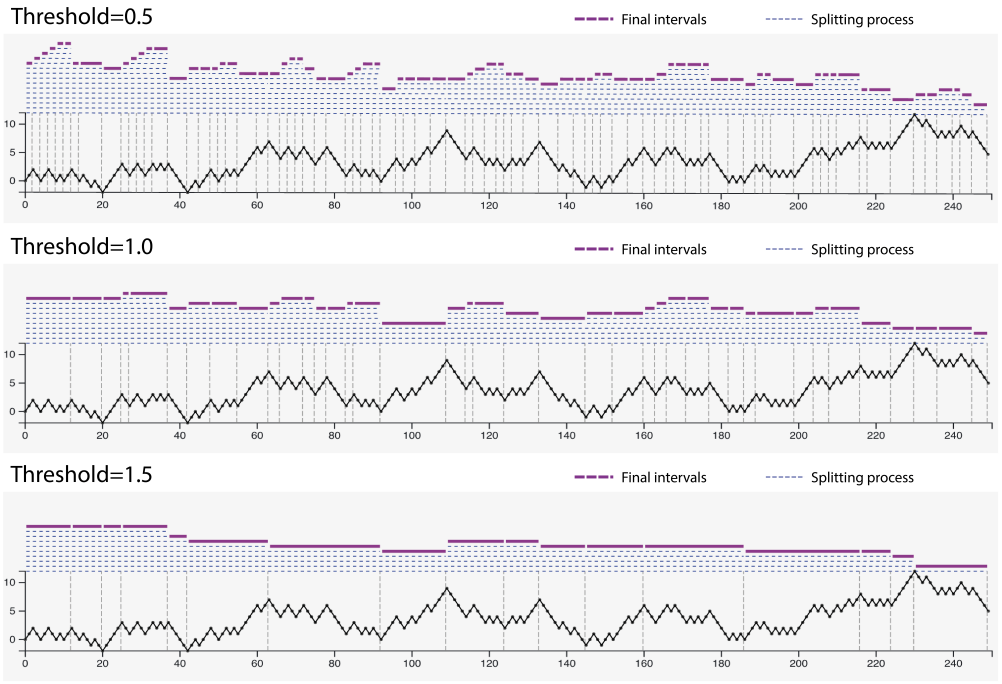


Figure 4: Division of a time series into intervals using different thresholds for the size of the largest triangle (top: threshold=0.5, middle: 1.0, bottom: 1.5). Each view consists of a raw data series (bottom) and the resulting time intervals (top). Blue dotted lines represent the recursions of the splitting process and bold purple lines represent the final intervals while grey dotted lines represent the end of each interval. A larger threshold indicates a greater tolerance for variations within the data, resulting in a coarser segmentation that represents more pronounced trends. Conversely, a smaller threshold refines the segmentation capturing subtler variation.

T2. Deriving complex patterns by computing temporal relations between time intervals in multivariate time series

After identifying the time intervals containing basic patterns in each univariate time series (Figure 5), we proceed to compute the temporal relations between the time intervals across multiple attributes by employing a subset of Allen's interval algebra consisting of the relations `before`, `after`, and `overlap`. We allow a certain margin of overlapping ω to be present in the `before` and `after` relation. Any relation where two intervals share a sufficiently long ($> \omega$) period of simultaneous existence is considered as an instance of the `overlap` relation. The threshold ω is specified as percentage of the duration of the shorter interval.

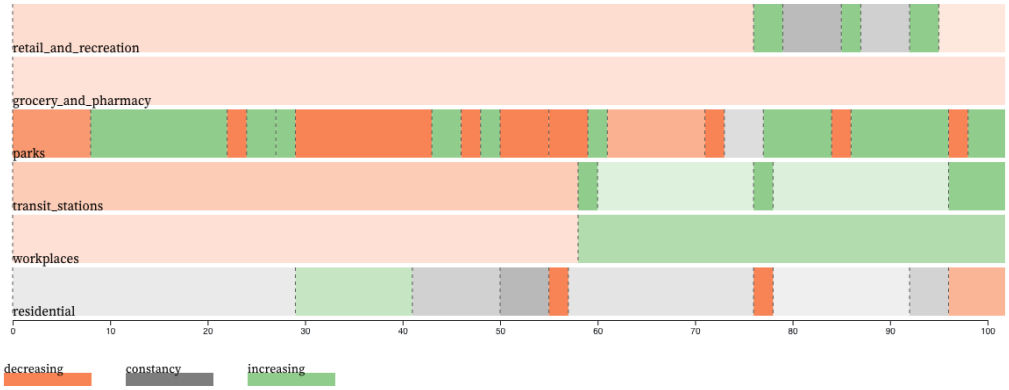


Figure 5: A segment of an univariate time series from the mobility dataset distilled into basic trend patterns. Colors denote the trend directions: orange for `decreasing`, grey for `constancy`, and green for `increasing`. Opacity signifies the change rate, i.e., the amount of change of the value divided by the interval length.

To identify relations, we employ the following algorithm. Firstly, for each interval in one time series that contains a pattern (referred to as the reference interval), the algorithm searches for its temporal neighbors in the other time series. Two intervals are considered temporal neighbors if they either overlap or if the temporal distance between the end of the earlier interval and the start of the later interval does not exceed a predefined threshold, denoted as δ. If the neighboring interval overlaps with the reference interval by more than a specified value of the threshold ω, the relation is labeled as `overlap`. Otherwise, if the neighbor starts earlier, the relation is labeled as `before`, and if the neighbor starts later, the relation is labeled as `after`. Analysts

have the flexibility to adjust the parameters δ and ω based on their specific requirements. In the provided examples, we have chosen δ to be 1 day and ω to be 20% of the shorter interval's duration. Figure 6 demonstrates an example of the relation `after` between two patterns.

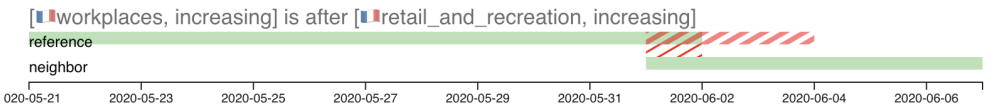


Figure 6: An example of a temporal relation between two trend patterns. The `increasing` pattern in `workplaces` (bottom) is `after` the `increasing pattern` in `retail and recreation` (top). As the two intervals overlap by less than the threshold ω (bold hash pattern in red), the relation is identified as `after` rather than `overlap`.

T3. Exploring occurrence patterns of temporal relations through an interactive visual interface.

In this task, the objective of an analyst is to understand the occurrence patterns of temporal relations by examining their frequency distributions. This task is meant to be performed separately for each type of temporal relation, i.e., `before`, `after`, or `overlap`. We shall call the relation that is currently explored the target relation. This task is structured in accordance with the Visual Information-Seeking Mantra of "overview first, zoom and filter, then details-on-demand" [31].

T3.1 Overview: Matrix visualization of occurrence patterns

To begin our exploration, we first introduce a matrix visualization that aids in understanding the occurrence patterns of temporal relationships. An example is demonstrated in Fig. 7. Each cell in the matrix represents the frequency of the target relation (`before` in Fig. 7) occurring between two patterns across different attributes. In this matrix, rows and columns are divided by dashed lines into three blocks corresponding to the increase, constancy, and decrease patterns. Within these blocks, the rows and columns correspond to the different attributes. The color intensity of a cell indicates the frequency of the target temporal relation: darker shades represent more frequent occurrences, while lighter ones signify fewer.

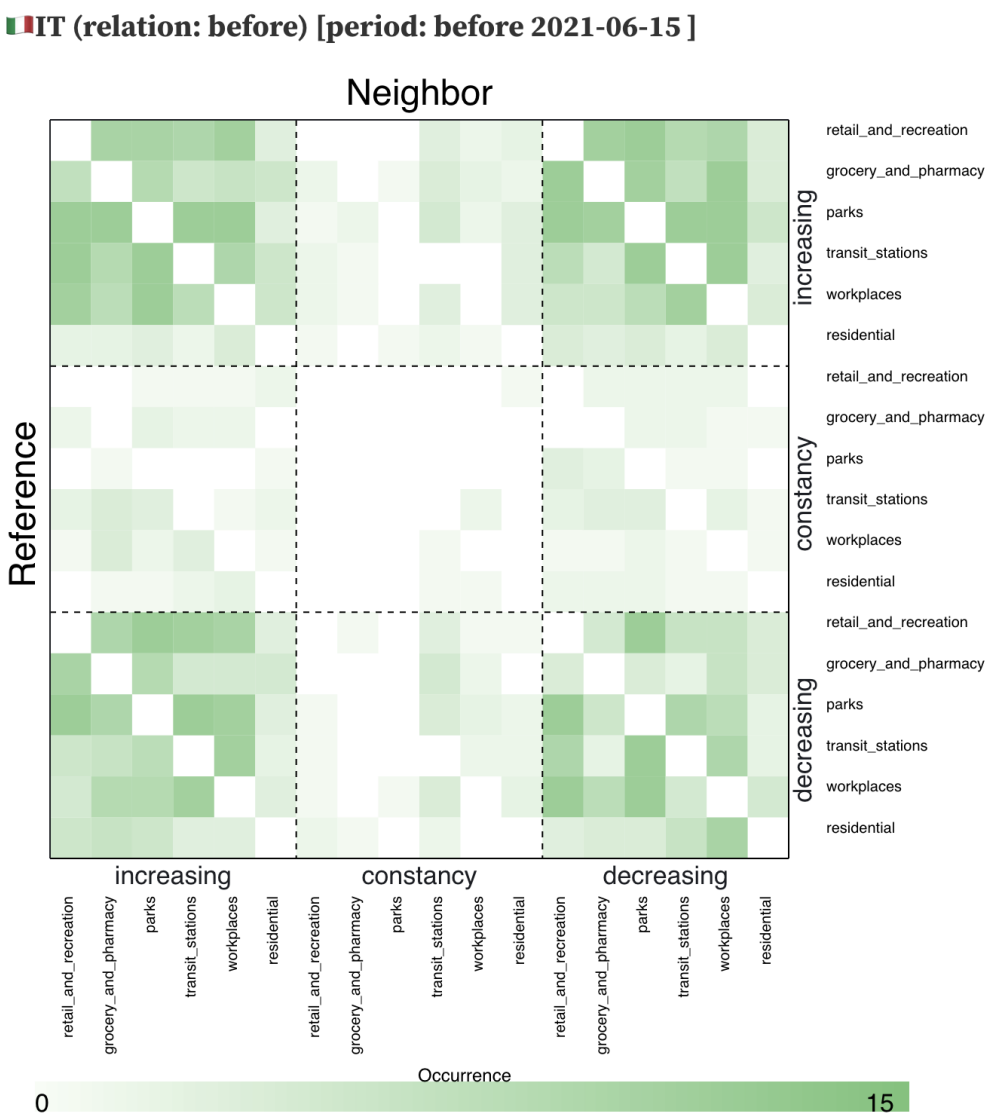


Figure 7: Relation occurrences matrix: Each cell represents the frequency of the target relation (`before` in this example) between two patterns across various attributes. The matrix rows and columns are divided into three blocks corresponding to increase, constancy, and decrease patterns. Darker shades indicate more frequent occurrences, while lighter ones signify fewer.

The investigation is done using a display with multiple matrices (Figure 8), i.e., we apply the "small multiples" technique considered by Edward

Tufte as the best design solution for a wide range of problems [32, p.67]. In accordance with the Jacques Bertin's concept of an image as "the meaningful visual form, perceptible in the minimum instant of vision" [5, p.11], each matrix can be perceived holistically as a single object. This allows for an at-a-glance comparison between matrices, without involving minute details.

The multi-matrix display in Figure 8 visualizes the occurrence patterns for the `before` relation for each country. Initial observation of these matrices reveals certain similarities, such as comparable white crossing lines in the Czech Republic, Hungary, Luxembourg, Slovenia, and Slovakia.

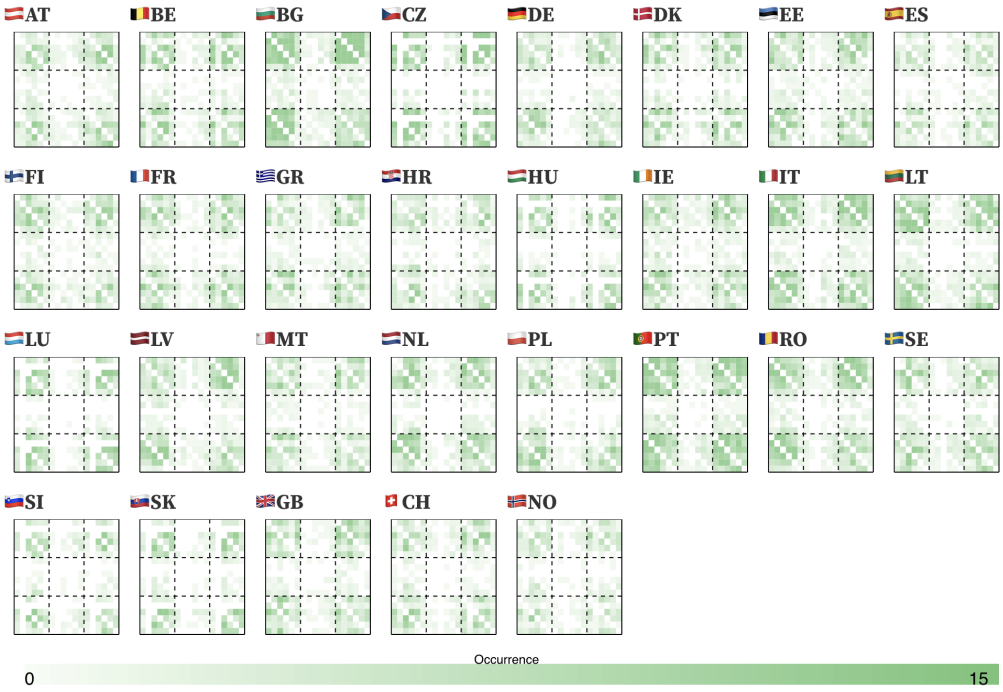


Figure 8: Grid of matrix views for 29 European countries. Each matrix within the grid represents the occurrences of the `before` relation for each pair of patterns within the respective country.

For more effective comparison, the user can set the multi-matrix view to show normalized deviations from the average (Figure 9). Within these matrices, each matrix cell displays the difference between a normalized occurrence value for a given country and the average normalized value for corresponding pair of patterns across all countries. The normalized occurrence value is

computed as the ratio of the occurrence count of a specific pattern pair to the total number of occurrences of the target relation within the matrix. We can represent this concept with the following formula:

$$diff = NOV_{ref,neigh,rel,c} - ANOV_{ref,neigh,rel} \tag{1}$$

where ref and $neigh$ mean a pair of a reference interval and its neighbor, rel means a relation, and c means a country.

The Normalized Occurrence Value (or NOV) is calculated by dividing the count of occurrences for each pattern pair by the total occurrences of the target relation in the matrix:

$$NOV_{ref,neigh,rel,c} = \frac{count_{ref,neigh,rel,c}}{totalOccurrences_{rel,c}} \tag{2}$$

The Average Normalized Occurrence Value (or ANOV) is the average of the NOVs across all countries:

$$ANOV_{ref,neigh,rel} = \frac{\sum_c NOV_{ref,neigh,rel,c}}{N} \tag{3}$$

In this formula, N is the total number of countries.

Differences are shown through diverging colors, making similarities in occurrence patterns more readily observable.

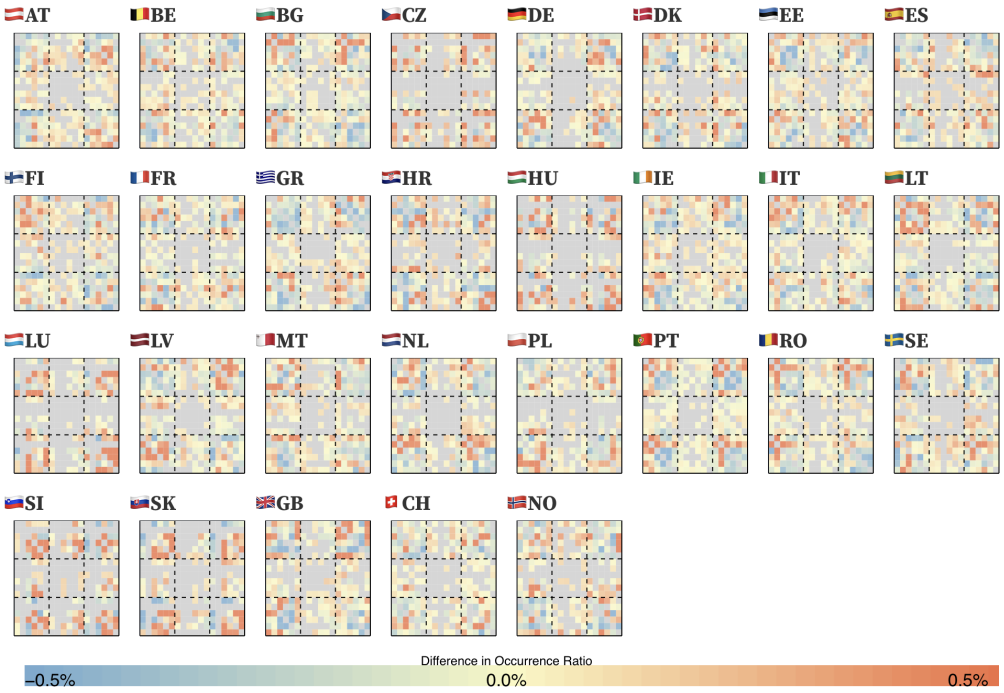


Figure 9: A multi-matrix view of the relations between mobility trends for 29 European countries. Each matrix illustrates the deviation of the normalized frequencies of the relation occurrences from the average of the normalized frequencies for the `before` relation. These differences are expressed using diverging colors.

For an overview of the similarity relationships between countries, analysts can apply dimensionality reduction to the set of matrices. We recommend using a dimensionality reduction algorithm from the class known as neighbour embedding algorithms, which give priority to preserving local neighborhoods at the cost of higher distortion of longer distances between embedded data items. Hence, highly similar items (i.e., neighbors in the multidimensional space) receive close positions in the low-dimensional projection space. One of such algorithms is t-SNE [23], which we use in our example. In Figure 10, t-SNE was utilized with two different values of the parameter `perplexity`, 5 and 25. This parameter approximately defines the number of neighbours to be considered when placing points in the projection space. Since it is not known in advance how many neighbours, in terms of similarity of the relation occurrence distribution, a country may have, it is reasonable to consider projections obtained with smaller and larger values of the perplexity parameter.

In our example, both projections exhibit clusters of countries with similar matrices, for example, the aforementioned five countries (i.e., the Czech Republic, Hungary, Luxembourg, Slovenia, and Slovakia), as well as the Baltic countries (i.e., Lithuania, Latvia, and Estonia). We do not observe significant difference between the results obtained with the two perplexity values.

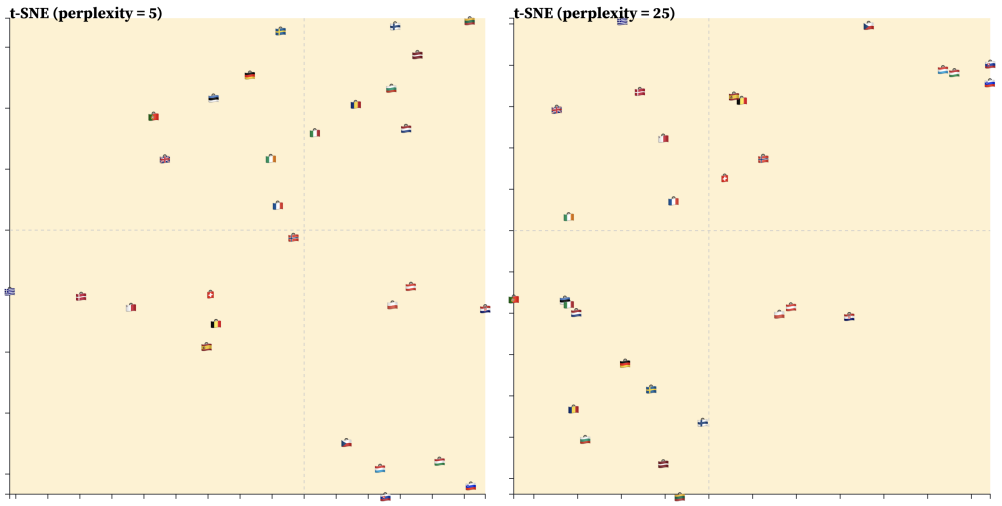


Figure 10: t-SNE projections of the set of countries obtained with perplexity values of 5 (left) and 25 (right), based on the similarity between the matrices of the normalized occurrence frequencies of the `before` relation. The projection provides a spatial representation of the similarities.

T3.2 Zoom, Filter, and details-on-demand

A. Selecting a matrix of interest and a pair of patterns

The subsequent step encompasses zooming and filtering, permitting analysts to narrow down their analysis to a particular matrix of interest ("filter"), which is shown in a larger format ("zoom") to facilitate focusing on specific patterns and their temporal relations. Analysts can then choose a target temporal relation, and the frequency of the selected relation between each pair of patterns will be shown ("details-on-demand"). In our illustrations, the analyst chooses the `before` relation to explore the frequencies of neighboring patterns that occur prior to reference patterns.

Taking Malta as an example (Figure 11, left), the matrix visualization shows the frequencies of the relation occurrences for different pattern pairs over the period between February 15, 2020 and June 15, 2021 (i.e., in first half

of the time span of the available data set, including the beginning of the pandemic). We observe that decreasing patterns of different attributes often precede increasing patterns of other attributes. For instance, we observe a large number of instances where a decreasing trend in `retail and recreation` precedes an upswing in `parks`, `transit stations`, and `workplaces` (Figure 11, left, block 1). Similarly, decreasing patterns in these three categories of places often precede an increase in `retail and recreation`. On the other hand, there are fewer instances of an increasing pattern in `retail and recreation` preceding a decrease in these place categories (Figure 11, left, block 2). Likewise, the matrix for Italy shown on the right of Figure 11 also displays similar patterns but includes another category of place, namely `grocery and pharmacy` (Figure 11, right, block 1).

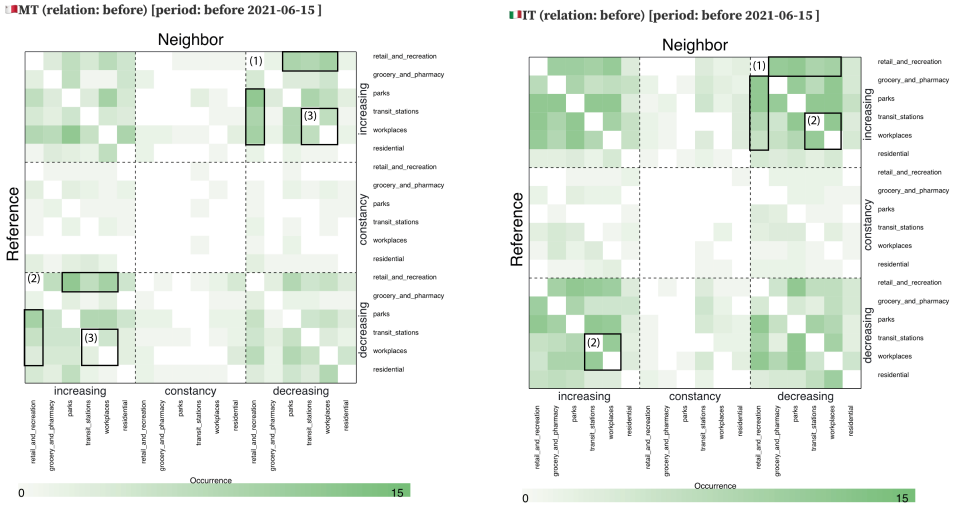


Figure 11: Matrix visualizations of the occurrence frequencies of the temporal relation `before` between trend patterns of different attributes for Malta (left) and Italy (right).

Notably, Italy has higher frequencies of the `before` relation between decreasing trends in `transit stations` and increasing trends in `workplaces` and vice versa, indicating a different pattern of mobility in Malta (Figure 11, left, block 3 and right, block 2) as compared to Italy (Figure 11).

These matrix visualizations serve to observe and compare the overall frequency distribution of temporal relations between different patterns across various contexts or regions, providing initial insights for further detailed analysis.

B. Density chart grid view for temporal relation distributions

To exhibit the distributions of temporal relations between different abstract patterns, we utilize a grid view of density charts. Differently from the matrix view, which shows information for a chosen target relation, the grid view presents information for a chosen reference pattern. For each other pattern, there is a density chart representing the distribution of the relative (with respect to the reference pattern) times of occurrence of this other pattern. The chart includes two density plots, one for the `before` relation (blue) and another for the `after` relation (red). The `overlap` relation is intentionally excluded from this visualization due to its significantly higher frequency, which decreases the visibility of the distributions of the other relations. It is worth noting that our framework provides the flexibility to exclude any relation, allowing for a more focused examination of the other specific relations. The density charts are arranged in a grid with rows corresponding to the attributes and columns to the different patterns, i.e., increase, constancy, and decrease.

Figure 12 illustrates the grid view. The figure includes two grids representing data from two time periods: before and after the 15th of June, 2021 (midpoint of the available data). The reference pattern is the increase of the attribute `workplaces`. In each grid view, the rows correspond to six attributes (`retail and recreation`, `grocery and pharmacies`, `parks`, `transit station`, `workplaces`, and `residential`) and columns to three trend patterns (`increase`, `constancy`, and `decrease`), resulting in a total of 18 grid cells. The cells in all but one rows contain density charts showing the distributions of the relative times of the occurrences of the neighboring patterns with respect to the reference pattern. The row of the attribute whose pattern is chosen as the reference is empty, because only relations between patterns of distinct attributes are considered in our framework.

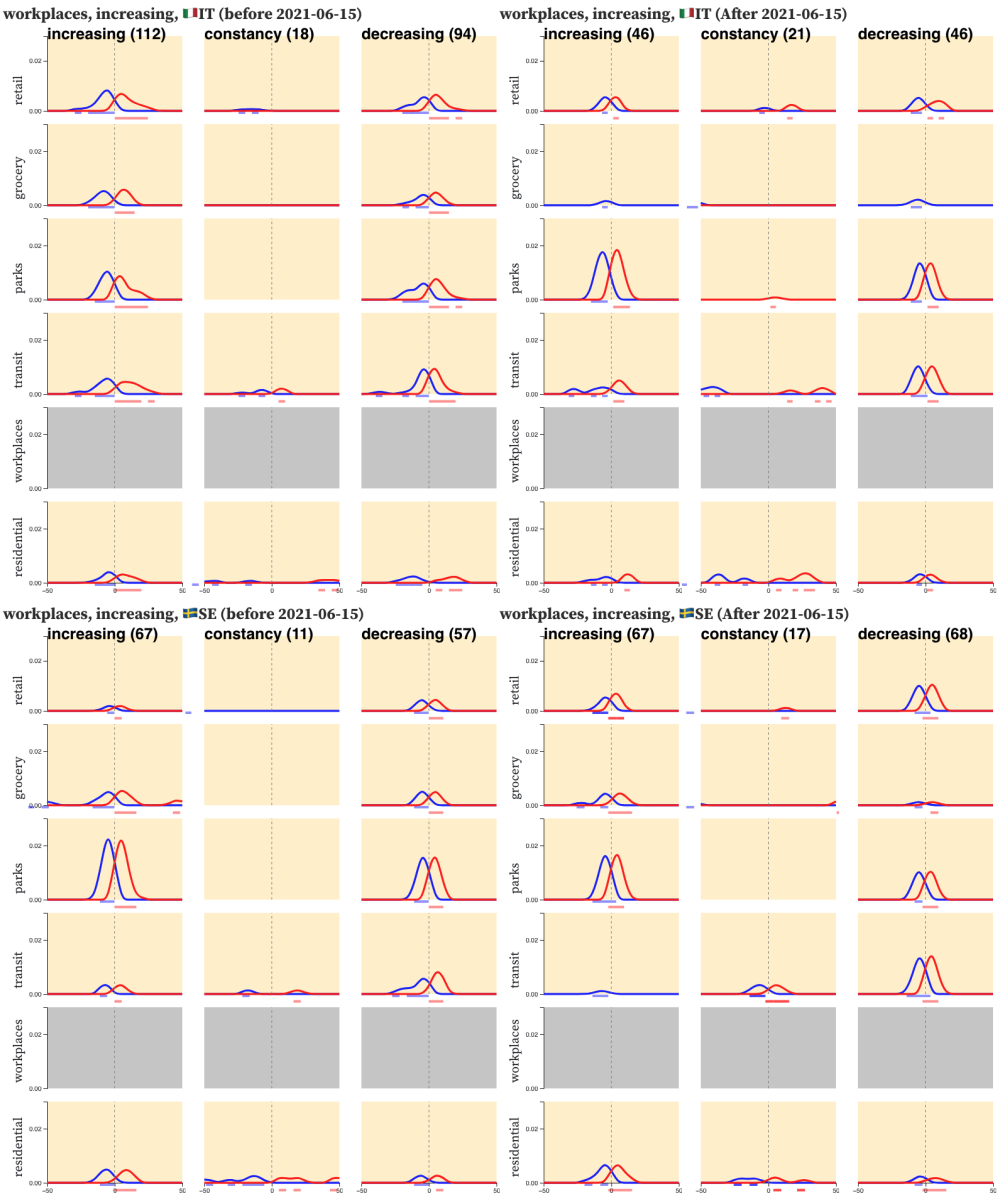


Figure 12: Grid views of density charts for the distributions of the neighbors' relative times for the `increase` pattern of `workplaces` in Italy (top) and Sweden (bottom). The grids on the left include the data from the time period before 15/06/2021, and the grids on the right show the relation distributions after this date. Colors denote different temporal relations: blue for `before` and red for `after`.

Comparison of relations of the `increase` *in the visits to* `workplaces` *between Italy and Sweden*

In this case study, we focus on the relations of the `increase` pattern in visiting `workplaces` with different trend patterns of the other attributes occurring in the temporal neighborhood of the reference pattern defined using a temporal threshold of $\delta = 1$ day. We compare the distributions of the neighboring patterns for Italy and Sweden. We segment the data into two subsets based on whether they fall before or after a chosen midpoint date June 15, 2021, for comparative analysis of two time periods.

In Italy (Figure 12, top), more occurrences of the reference pattern in neighborhoods of other patterns are noted before the midpoint date than afterwards. This suggests that the `increase` pattern of `workplaces` was more prominent at the early stages of the pandemic. Given that Italy implemented lockdown measures relatively early [13], this observation aligns with the expectation that people's mobility would have been significantly affected by these measures. In contrast, Sweden (Figure 12, bottom), known for not imposing any form of lockdown [13], exhibits fewer occurrences of the reference pattern in relation to others before the midpoint date than after it.

Upon closer inspection, we find that the same pattern before the midpoint in Italy has more relations with the `decrease` pattern of `transit station` than its counterpart in Sweden, implying different mobility patterns in these countries.

4. Case study: Team behaviours in football

In this section, we present another case study using data from professional football (or soccer) matches. Understanding collective movements is crucial for interpreting tactical behaviors in football. For example, the team that has gained the ball possession tends to extend its width while the team without possession tends to get more compact [8]. Revealing temporal relations between such kinds of trends of different attributes can enhance understanding of the data. For example, an `increase` in `average velocity` (i.e., average speed of players on both teams) `before` an `increase` in `goal distance` (i.e., distance between a team's own goal and the mean position of the outfield players on that team, excluding the goalkeeper) implies a quick attack such as a counter-attack.

Data description

We use continuous time series of the teams' collective movements computed from players' positions. We have data from two matches, labeled as BB and BN, in which the same home team, denoted as D, competes against two distinct opponents, referred to as O. One match can be divided into four subsets of game episodes distinguished by two factors: the stage of the match (either the first half or the second half) and the team possessing the ball, i.e., D or O. Each half contains around 67500 timesteps (i.e., 45 minutes given that the raw data has a sampling rate of 25 Hz). For each team and each time step, we compute `team width` (horizontal distance between the leftmost and rightmost players on a team excluding the goalkeeper), `team depth` (vertical distance between the frontmost and rearmost players on a team excluding the goalkeeper), and `goal distance` (distance between a team's own goal and the mean position of the players on that team excluding the goalkeeper). The attributes of the two teams are distinguished by the prefixes `home` and `away` in the attribute names, for example, `home width` and `away width` As there exists strong correlation between the average velocities of the players of the two teams, we compute `average velocity` on both teams (excluding the goalkeepers).

Comparative analysis of team strategies in first and second halves of the BB match

In defining temporal neighborhoods and identifying relations, we use the threshold values $\delta = 25$ frames (i.e., 1 second) and $\omega = 0.3$; see section 3.2, task T2.

Figure 13 enables a comparative study of relation occurrence patterns under eight situations, i.e., two matches, the first and second halves of each match, and different teams (D and O) in possession of the ball. Similarly to Section 3, each cell within these matrices displays the occurrence frequency of the relation `before` between the corresponding pattern pairs across all attributes.

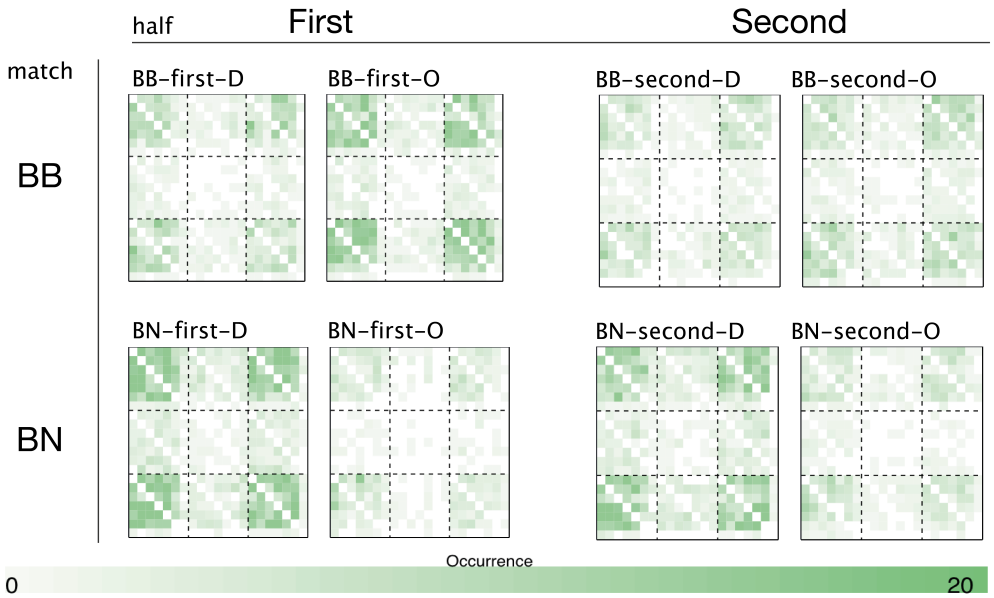


Figure 13: A 2 ×4 grid of matrix views. The matrices in the top row correspond to the first match (BB) and those in the bottom row to the second match (BN). Each matrix within the row represents the occurrences of the `before` relation for each pair of patterns in a distinct subset of game episodes: the first or the second half of a match and the team D or O in possession of the ball.

Upon initial observation, we identify remarkable similarities in the patterns of relation occurrence in the subsets of episodes in both matches when team D is in control of the ball, i.e., the matrices with labels that end with `-D`, such as BB-first-D. These patterns are also similar to the pattern of relations when team O is in possession in match BN, evident in both BN-first-O and BN-second-O. There is higher similarity between the matrices BB-first-O, BN-first-D, and BN-second-D, which indicates that the team D behaved in the match BN similarly to the behavior of their opponents in the first half of the match BB. In the first half of the match BB, team D had lower relation frequencies than team O, whereas in the second half the frequencies were nearly equal.

The apparent differences between the absolute frequencies can be explained by the differences in the total duration of the ball possession between the teams. Therefore, to reveal possible differences in team tactics, it makes sense to transform the absolute frequencies to normalized values, as in Fig. 9.

The normalized deviations from the average, as depicted in Figure 14,

enhance the visibility of certain patterns of occurrences, although some similarities with the matrices in Figure 13 still persist. Specifically, a distinct difference in color (blue and yellow) for the top-left vertical line between the BN-second-D and the BN-first-D matrices can be observed. However, this difference is merely expressed by color intensity in Figure 13.

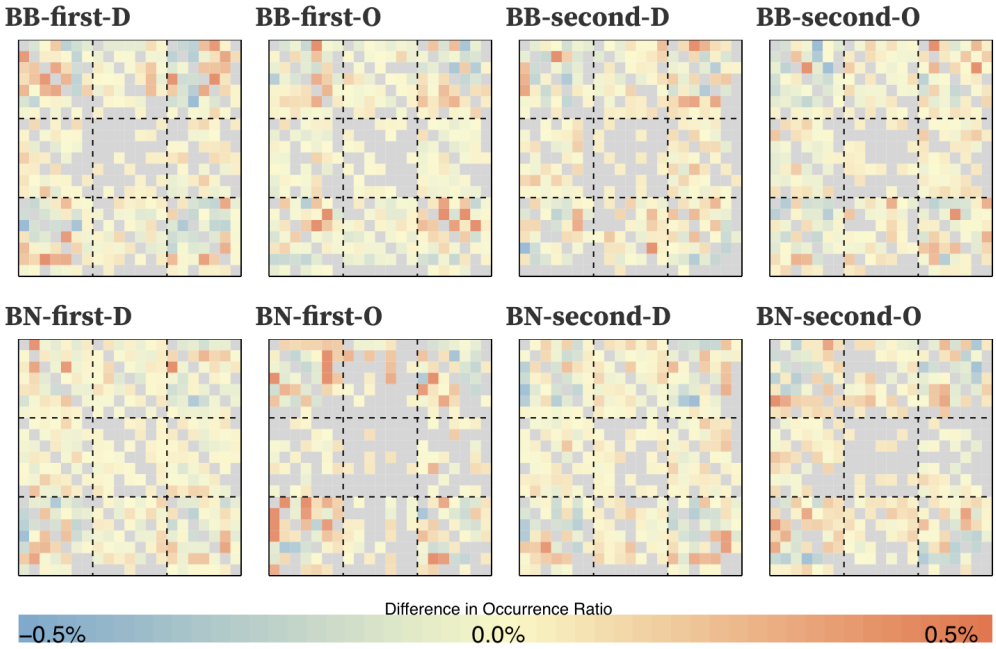


Figure 14: A 2 ×4 grid of matrix views. The matrices in the top row correspond to the first match (BB), while those in the bottom row correspond to the second match (BN). Each matrix within the row illustrates the difference between the average of normalized occurrences of the `before` relation for each pair of patterns and the corresponding normalized value. Diverging colors are used to represent these differences. Each matrix corresponds to a distinct subset of game episodes, either the first or second half of a match, and whether the team D or O in possession of the ball.

To compare two halves of one match, analysts can subtract the normalized occurrence values of the second half from those of the first half to identify which pair of patterns appears more frequently in each half. Figure 15 demonstrates the result of this operation for the ball possession of O in the match BB.

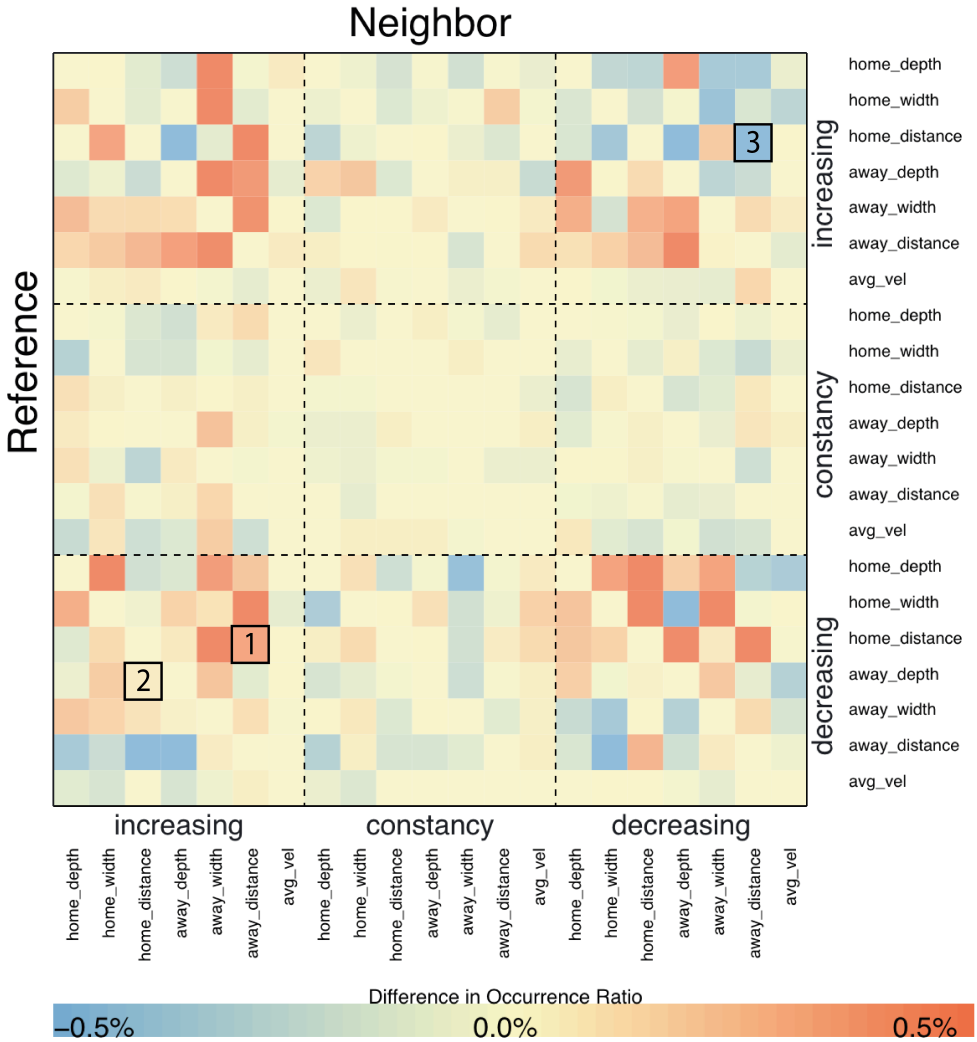


Figure 15: The matrix represents the difference in normalized occurrence values of the neighbor patterns preceding the reference patterns between the first and second halves of match BB when team O possesses the ball. The values are calculated by subtracting the occurrence frequencies in second half from those in the first half. Red indicates a higher frequency of occurrences in the first half while blue signifies more occurrences in the second half.

The `increase` pattern of `away distance` followed by the `decrease` pattern of `home distance` indicates that the away team is moving further away from their own goal (1 in Figure 15), suggesting an offensive strategy, while

the home team is moving closer to their goal, suggesting a defensive strategy. This pattern is more noticeable in the first half, concurring with the match report's statement that the away team initiated a strong offensive from the start of the game [15].

Similarly, we observe the `increase` pattern of `home distance` followed by the `decrease` pattern of `away depth` (2 in Figure 15), suggesting that the home team advances before the away team becomes more compact. We also observe that the `decrease` pattern of `away distance` preceding the `increase` pattern of `home distance` is more prevalent in the second half (3 in Figure 15), suggesting that the away team adopts a more defensive posture while the home team defends more aggressively. This suggests that the home team is preparing for an aggressive strategy, potentially anticipating a turnover or looking to exploit any gaps in the away team's formation, while the away team is playing compact. The increased prevalence of this pattern in the second half suggests an offensive shift of strategy by the home team, aligning with the match report that mentioned that the home team exerting considerable pressure on the away team's defense [15].

Density charts: Changes in behaviour between first and second halves

In this section, we use grids of density plots for a more detailed investigation of the temporal relations between patterns of different attributes. To compare two halves of a football match, we juxtapose two grids representing the corresponding data. Each grid comprises seven attributes (`home depth`, `away depth`, `home width`, `away width`, `home distance`, `away distance`, and `average velocity`) and three trend patterns (`increase`, `constancy`, and `decrease`), resulting in a total of 21 grid cells.

First, we focus on the `increase` pattern in `average velocity`. We observe a generally higher number of various neighboring patterns in the first half (Figure 16, top-left) compared to the second half (Figure 16, top-right), which may be a consequence of the higher number of occurrences o the `increase` pattern of `average velocity` during the first half.

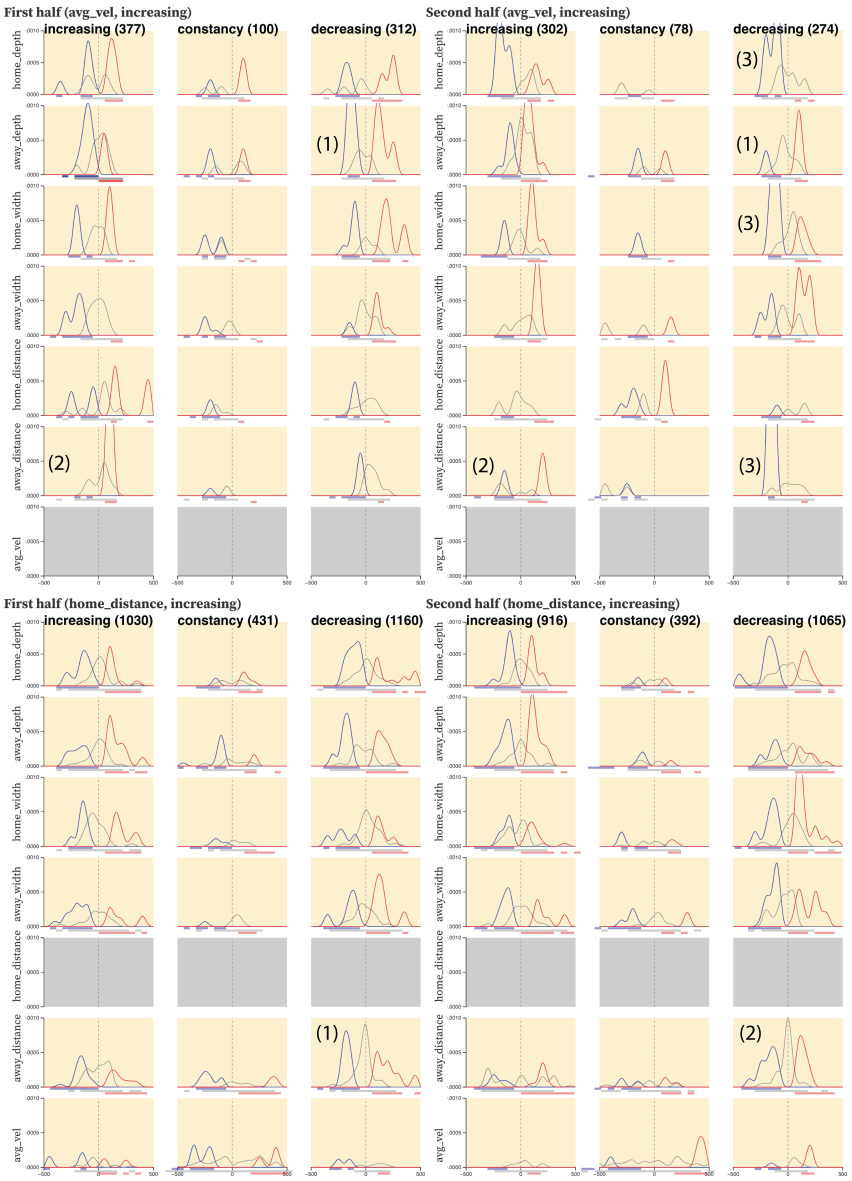


Figure 16: Comparison of two grid views from the first half (left) and second half (right) of a football game, with the reference attribute `average velocity` and pattern `increasing` (top) and with the reference attribute `home distance` and pattern `increasing` (bottom). In each grid, three density plots depict the distributions of the relative times of the neighboring patterns compared to the reference patterns, with blue representing the `before` relation, grey representing the `overlap` relation, and red representing the `after` relation.

This seems logical since players typically experience less fatigue in the first half, enabling them to change speed more frequently. We also observe less obvious differences, such as much higher frequencies of the `decrease` of `away depth` and `increase` of `away width` occurring before the `increase` of modify`average velocity` in the first half than compared to the second half (1 in Figure 16). The `increase` of `away distance` after the `increase` of `average velocity` is also observed more often in the first half (2 in Figure 16, top). These indicate that the first half included a larger number of active attacks by the away team in which they moved with increasing velocity towards the opponent's goal. To prepare the attack, the team tended to increase the width (i.e., across the pitch) while decreasing the distances between the lines, i.e., the team's depth. The second half had prominently higher frequencies of the `decrease` of `home depth`, `home width`, and `away distance` before the increase in the `average velocity` (3 in Figure 16, top). This indicates that the away team was often retreating to their goal before the increase of the `average velocity` and, at the same time, the home team was getting more compact, which usually happens in preparation to an attack of the opponents.

Next, we examine the relations of the `increase` pattern in `home distance` (Figure 16, bottom). It is apparent that this pattern has numerous relations with the `decrease` pattern of `away distance` as a neighboring pattern, particularly during transitions of ball possession, as observed earlier by Shirato et al. [30]. However, we can verify that in the first half, there are more neighboring patterns of the `decrease` of `away distance` preceding the reference pattern than those succeeding it (1 in Figure 16, bottom). In contrast, during the second half, we observe more of the same neighboring patterns following the reference pattern than those preceding it (2 in Figure 16, bottom). These observations suggest that the away team tends to move more quickly in the first half, while the reverse occurs in the second half. This implies that the home team has greater control over the match in the first half compared to the second half.

5. Discussion

In this study, we developed a framework for analysis of multivariate time series (MVTS) data. The framework includes abstraction of value sequences into instances of basic variation patterns and exploration of temporal relations between these pattern instances across different variables.

Our framework has several strengths. As its core, it offers flexibility and generality to meet a wide range of data analysis needs in MVTS.

Its flexibility emerges from the proposed approach to pattern extraction. The framework does not require to pre-define pattern duration, i.e., the length of the time interval containing a pattern. This provides flexibility for finding pattern instances of variable duration and adapting to data with diverse properties, such as sampling rate, rate of changes, and amplitudes of changes.

On the other hand, the generality of our framework manifests in its ability to handle different types of temporal patterns. While our pilot studies focused on extracting basic trend patterns for their easy interpretability, the framework is not limited to these. It is capable of extending to any other types of temporal patterns depending on the character of studied changes and the analysis goals. For example, there may be pattern types reflecting states, such as high, medium, and low values. Moreover, patterns may be composed of values of categorical attributes.

Our framework can be extended, offering additional or alternative methods for pattern definition and extraction. This accommodates the diverse needs of analysts, who may opt to sketch a pattern, use an interface like Time Searcher [11] to define the pattern, or even define composite patterns built of basic patterns like a peak followed by a trough. The system, in response, identifies patterns similar to the sketch or template provided, opening a door to more customized and insightful analysis.

Another aspect of extensibility comes in the form of temporal relations considered in the analysis. While we used a subset consisting of three relations, before, overlap, and other, other relations from the Allen's algebra of time intervals can also be considered in the analysis.

We have demonstrated the application of our framework in two distinct use cases: exploring mobility patterns during the COVID-19 pandemic and analyzing team behaviors in professional football matches. In both cases, our framework was able to provide valuable insights into the temporal relations between different patterns of attribute variation.

In the COVID-19 mobility data case, our framework was able to identify and visualize temporal relations between different mobility patterns within each country and to investigate the distributions of temporal relations across countries. This analysis revealed interesting patterns, such as the increase in workplace mobility preceding an increase in residential mobility, likely due to people preparing for remote work during lockdowns. Moreover, the

framework was able to highlight differences in mobility patterns between countries, reflecting the varying policies against the pandemic.

In the football data case, our framework was able to identify and visualize temporal relations between different team behaviors. For example, it was able to detect a change in team behavior between the first and second halves of a match, which aligned with the match report.

Despite its strengths, our framework also has some limitations. It requires a number of thresholds and parameters to adjust (e.g., ω, δ, and perplexity), which could potentially confuse analysts and require sensitivity analysis. In terms of trend patterns, the results may not be expressive enough as it does not consider a combination of univariate temporal patterns such as peaks and troughs. For temporal relations, our framework does not provide a holistic understanding of pairwise relations, as it only calculates the distribution. The importance of relations is thus expressed only through frequency.

A major limitation is that the framework does not scale well to the numbers of attributes, pattern types, and types of relations. A possible approach to alleviate this is to develop a guiding system that suggests potentially interesting selections to explore. During the analysis, the analyst can interactively construct a knowledge graph or several graphs for different conditions or classes of situations, such as fast attack or gradual approach in football. A knowledge graph contains pattern types of different abstraction levels and relations between them, including temporal and hierarchical.

We also foresee several potential improvements and directions for future work to enhance our framework. For trend patterns, one possible direction is to construct combined univariate patterns from basic patterns, thus increasing expressiveness of the results. For temporal relations, future work could incorporate approaches capable of dealing with multiple pairwise relations, such as a network-based approach where nodes represent temporal patterns and edges represent relations between them. This could provide a more holistic understanding of the relations and allow for the identification of important nodes using graph centrality measures.

An important consideration is the accessibility and user-friendliness of any tools developed to support this framework. Our primary objective was the development and validation of the framework itself, while the prototype tools we implemented served mainly as proof of concept. These tools were not optimized for end-user adoption and would require further user-centered design for broad accessibility. Conceptually, our framework is simple enough to understand to understand and adopt without specialized technical skills.

However, the practical implementation of the framework for end-users would necessitate domain-specific enhancements to facilitate its use. Our proof-of-concept implementation has demonstrated the types of analyses possible with the framework, but further development is needed to make these operations user-friendly in specific domains of application.

6. Conclusion

This paper presented a framework that unifies various methods for the abstraction of multivariate time series (MVTS) data. The unification is achieved by integrating these different methods into a *cohesive workflow*, which allows to understand dynamic phenomena through the lens of temporal relations, the identification of basic behavior patterns and the examination of temporal relations among these patterns. Our framework is designed to identify basic behavior patterns and examine the temporal relations among these patterns, taking into account temporal lags and varying duration of the patterns. This feature enhances the understanding of complex interactions among multiple attributes, making the framework valuable for analysts.

The effectiveness and versatility of our framework were demonstrated through its application to mobility data during the COVID-19 pandemic and football (soccer) data. Despite its strengths, the framework has some limitations, such as the need for further enrichment to handle intricate variable interactions and the integration of more complex patterns. These limitations provide avenues for future work.

In conclusion, our framework offers an approach to abstracting MVTS, with a focus on understanding temporal relations. By integrating various methods into a single workflow, it enables analysts to effectively explore and comprehend complex temporal relations in MVTS data.

7. Acknowledgements

This work was partly supported by Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the *Lamarr Institute for Machine Learning and Artificial Intelligence* (Lamarr22B), and by EU in projects *SoBigData++* and *CrexData* (grant agreement 101092749).

References

[1] Allen, J.F., 1983. Maintaining knowledge about temporal intervals. Communications of the ACM 26, 832–843. doi:10.1145/182.358434.

[2] Andrienko, N., Andrienko, G., 2023. It's about time: Analytical time periodization. Computer Graphics Forum doi:10.1111/cgf.14845.

[3] Andrienko, N., Andrienko, G., Fuchs, G., Slingsby, A., Turkay, C., Wrobel, S., 2020. Visual analytics for data scientists. Springer.

[4] Andrienko, N., Andrienko, G., Miksch, S., Schumann, H., Wrobel, S., 2021. A theoretical model for pattern discovery in visual analytics. Visual Informatics 5, 23–42. doi:10.1016/j.visinf.2020.12.002.

[5] Bertin, J., 1983. Semiology of graphics. University of Wisconsin Press.

[6] Combi, C., Oliboni, B., 2012. Visually defining and querying consistent multi-granular clinical temporal abstractions. Artificial intelligence in medicine 54, 75–101. doi:10.1016/j.artmed.2011.10.004.

[7] Deng, Z., Weng, D., Xie, X., Bao, J., Zheng, Y., Xu, M., Chen, W., Wu, Y., 2022. Compass: Towards better causal analysis of urban time series. IEEE transactions on visualization and computer graphics 28, 1051–1061. doi:10.1109/TVCG.2021.3114875.

[8] Fonseca, S., Milho, J., Travassos, B., Araújo, D., 2012. Spatial dynamics of team sports exposed by voronoi diagrams. Human movement science 31, 1652–1659. doi:10.1016/j.humov.2012.04.006.

[9] Google, 2020. COVID-19 community mobility reports. https://www.google.com/covid19/mobility/. Accessed: 2022-4-24.

[10] Granger, C.W.J., 1969. Investigating causal relations by econometric models and cross-spectral methods. Econometrica: journal of the Econometric Society 37, 424–438. doi:10.2307/1912791.

[11] Hochheiser, H., Shneiderman, B., 2001. Visual queries for finding patterns in time series data. University of Maryland, Computer Science Dept. Tech Report, CS-TR-4365 .

[12] Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9, 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[13] Islind, A., Óskarsdóttir, M., Steingrímsdóttir, H., 2020. Changes in mobility patterns in europe during the COVID-19 pandemic: Novel insights using open source data. arXiv.org doi:10.48550/arXiv.2008.10505.

[14] Keogh, E., Chu, S., Hart, D., Pazzani, M., 2004. Segmenting time series: a survey and novel approach, in: Data Mining in Time Series Databases. WORLD SCIENTIFIC. volume 57 of *Series in Machine Perception and Artificial Intelligence*, pp. 1–21. doi:10.1142/9789812565402_0001.

[15] kicker, 2018. Furioser BVB! nach Robert-Lewandowski-Doppelpack drehen reus & co. auf - borussia dortmund baut vorsprung auf bayern münchen auf insgesamt sieben punkten aus. https://www.kicker.de/dortmund-gegen-bayern-2018-bundesliga-4243356/analyse. Accessed: 2023-6-20.

[16] Kolda, T.G., Bader, B.W., 2009. Tensor decompositions and applications. SIAM Review 51, 455–500. doi:10.1137/07070111X.

[17] Köthur, P., Witt, C., Sips, M., Marwan, N., Schinkel, S., Dransch, D., 2015. Visual analytics for correlation-based comparison of time series ensembles. Computer graphics forum: journal of the European Association for Computer Graphics 34, 411–420. doi:10.1111/cgf.12653.

[18] Lee, T.Y., Shen, H.W., 2009. Visualization and exploration of temporal trend relationships in multivariate time-varying data. IEEE transactions on visualization and computer graphics 15, 1359–1366. doi:10.1109/TVCG.2009.200.

[19] Li, J., Chen, S., Zhang, K., Andrienko, G., Andrienko, N., 2019. Cope: Interactive exploration of co-occurrence patterns in spatial time series. IEEE Transactions on Visualization and Computer Graphics 25, 2554–2567. doi:10.1109/TVCG.2018.2851227.

[20] Lin, J., Keogh, E., Wei, L., Lonardi, S., 2007. Experiencing SAX: a novel symbolic representation of time series. Data mining and knowledge discovery 15, 107–144. doi:10.1007/s10618-007-0064-z.

[21] Liu, S., Weng, D., Tian, Y., Deng, Z., Xu, H., Zhu, X., Yin, H., Zhan, X., Wu, Y., 2023. ECoalVis: Visual analysis of control strategies in coal-fired power plants. IEEE transactions on visualization and computer graphics 29, 1091–1101. doi:10.1109/TVCG.2022.3209430.

[22] Lütkepohl, H., 1991. Introduction to Multiple Time Series Analysis. Springer Berlin Heidelberg. doi:10.1007/978-3-662-02691-5.

[23] van der Maaten, L., Hinton, G., 2008. Visualizing data using t-sne. Journal of Machine Learning Research 9, 2579–2605. URL: http://jmlr.org/papers/v9/vandermaaten08a.html.

[24] Mannila, H., Toivonen, H., Inkeri Verkamo, A., 1997. Discovery of frequent episodes in event sequences. Data mining and knowledge discovery 1, 259–289. doi:10.1023/A:1009748302351.

[25] Monroe, M., Lan, R., Lee, H., Plaisant, C., Shneiderman, B., 2013. Temporal event sequence simplification. IEEE transactions on visualization and computer graphics 19, 2227–2236. doi:10.1109/TVCG.2013.200.

[26] Murphy, K.P., 2002. Dynamic Bayesian Networks: Representation, Inference and Learning. Ph.D. thesis. University of California.

[27] Patel, P., Keogh, E., Lin, J., Lonardi, S., 2002. Mining motifs in massive time series databases, in: 2002 IEEE International Conference on Data Mining, 2002. Proceedings., pp. 370–377. doi:10.1109/ICDM.2002.1183925.

[28] Sacchi, L., Larizza, C., Combi, C., Bellazzi, R., 2007. Data mining with temporal abstractions: learning rules from time series. Data mining and knowledge discovery 15, 217–247. doi:10.1007/s10618-007-0077-7.

[29] Shahar, Y., 1997. A framework for knowledge-based temporal abstraction. Artificial intelligence 90, 79–133. doi:10.1016/S0004-3702(96)00025-2.

[30] Shirato, G., Andrienko, N., Andrienko, G., 2023. Identifying, exploring, and interpreting time series shapes in multivariate time intervals. Visual informatics 7, 77–91. doi:10.1016/j.visinf.2023.01.001.

[31] Shneiderman, B., 2002. The eyes have it: a task by data type taxonomy for information visualizations, in: Proceedings 1996 IEEE Symposium on Visual Languages, IEEE Computer Society Press. pp. 336–343. doi:10.1109/vl.1996.545307.

[32] Tufte, E.R., 1990. Envisioning Information. Graphics Press, Cheshire, CT.

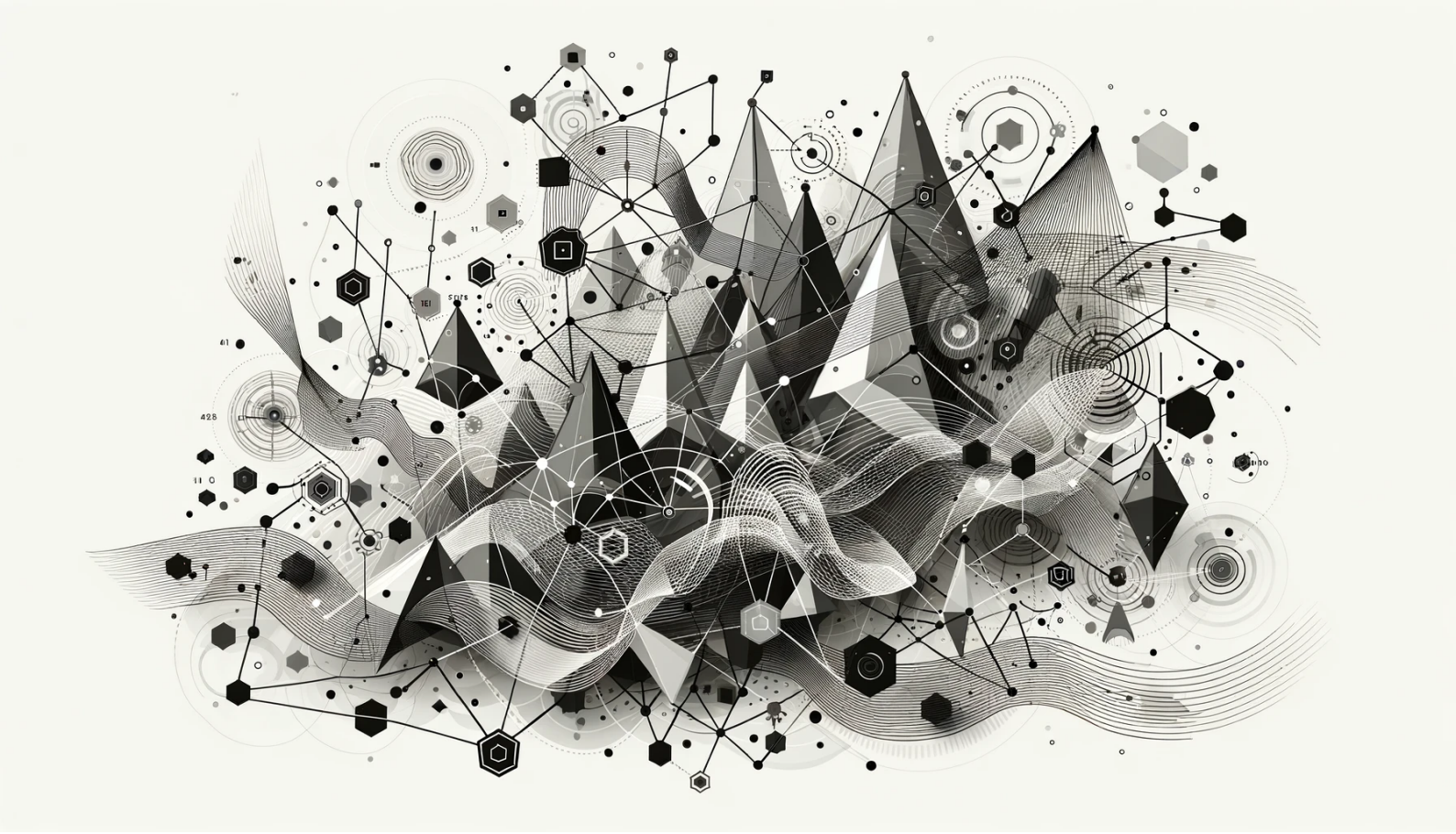
[33] Walker, J., Borgo, R., Jones, M.W., 2016. TimeNotes: A study on effective chart visualization and interaction techniques for Time-Series data. IEEE transactions on visualization and computer graphics 22, 549–558. doi:10.1109/TVCG.2015.2467751.

[34] Wongsuphasawat, K., Gotz, D., 2011. Outflow : Visualizing patient flow by symptoms and outcome. IEEE VisWeek Workshop on Visual Analytics in Healthcare , 25–28.

[35] Yi, J.S., Elmqvist, N., Lee, S., 2010. TimeMatrix: Analyzing temporal social networks using interactive Matrix-Based visualizations. International journal of human-computer interaction 26, 1031–1051. doi:10.1080/10447318.2010.516722.

[36] Zhao, J., Chevalier, F., Pietriga, E., Balakrishnan, R., 2011. Exploratory analysis of time-series with ChronoLenses. IEEE transactions on visualization and computer graphics 17, 2422–2431. doi:10.1109/TVCG.2011.195.

Chapter 5
Episodes and topics in multivariate temporal data



Volume 00 (1981), Number 0 pp. 1–21 COMPUTER GRAPHICS *forum*

Episodes and topics in multivariate temporal data

Natalia Andrienko[1,2], Gennady Andrienko[1,2], Gota Shirato[1,3]

[1]Fraunhofer Institute IAIS, Sankt Augustin, Germany, gennady.andrienko@iais.fraunhofer.de
[2]City University of London, UK
[3]University of Bonn, Germany

Abstract

The term 'episode' refers to a time interval in the development of a dynamic process or behaviour of an entity. Episode-based data consist of a set of episodes that are described using time series of multiple attribute values. Our research problem involves analysing episode-based data in order to understand the distribution of multi-attribute dynamic characteristics across a set of episodes.

To solve this problem, we applied an existing theoretical model and developed a general approach that involves incrementally increasing data abstraction. We instantiated this general approach in an analysis procedure in which the value variation of each attribute within an episode is represented by a combination of symbols treated as a 'word'. The variation of multiple attributes is thus represented by a combination of 'words' treated as a 'text'. In this way, the the set of episodes is transformed to a collection of text documents. Topic modeling techniques applied to this collection find groups of related (i.e., repeatedly co-occurring) 'words', which are called 'topics'. Given that the 'words' encode variation patterns of individual attributes, the 'topics' represent patterns of joint variation of multiple attributes. In the following steps, analysts interpret the topics and examine their distribution across all episodes using interactive visualisations.

We test the effectiveness of the procedure by applying it to two types of episode-based data with distinct properties and introduce a range of generic and data type-specific visualisation techniques that can support the interpretation and exploration of topic distribution.

Categories and Subject Descriptors (according to ACM CCS): [Human-centered computing → Visual analytics]: Visualization application domains—Visual analytics

1. Introduction

Everything in the world changes over time. Data describing changes often consist of time-referenced values of one or more attributes. The process or succession of changes along time can be divided into *episodes* each of which occurs over a specific time interval and is described by attribute values referring to different time steps within this interval. Fig. 1 demonstrates one of possible ways to divide continuous time series of attribute values into episodes using a sliding time window. Our research aims at finding ways to help humans understand dynamic phenomena or behaviours by analysing episode-based data.

While a line graph or other visual representation of a time series of attribute values can help a person understand the overall character of the development and identify different patterns of change, it can be difficult to get a holistic understanding of what is happening when changes are characterised by multiple attributes. To address this problem, we aim to extract interpretable patterns of change for singular attributes and then derive meaningful patterns of their joint changes.

The approach we develop and test is based on explicit representation of single-attribute temporal patterns as elements of data. Given a collection of episodes, we represent the variation of values of each individual attribute in each episode by a combination of symbols, which is treated as a 'word'. Hence, the variation of all attributes within an episode is represented by a combination of such 'words', which can be treated as a 'text'. The entire set of episodes is thus transformed to a collection of 'texts'. In natural language processing, there are topic modeling methods [VK20, AEG*23] that extract interpretable groups of semantically related words based on their co-occurrence in texts. By analogy, we expect that applying topic modeling methods to the set of 'texts' de-

rived from the episode-based data will result in finding interpretable groups of related 'words' encoding single-attribute variation patterns that tend to occur together in episodes. If successful, these groups can be considered as integrated multi-attribute temporal patterns representing components of complex behaviours or stages of complex processes.

The next goal after the extraction of multi-attribute patterns is to understand when, where, and under what circumstances different patterns occur, which is essential for understanding the dynamic phenomenon or behaviour as a whole. To facilitate this understanding, we want to provide an overview and enable the exploration of the distribution of the patterns in context, including space, time, and any conditions that may affect or be affected by the process or behaviour being studied.

With this paper, we intend to make the following contribution to the visual analytics research dealing with temporal data:

- Propose a conceptual framework and a general workflow for the analysis of dynamic phenomena described by episode-based data involving time series of multiple attributes.
- Explore the opportunities for analysis of dynamic phenomena given by explicit representation of temporal patterns of attribute variation.
- Investigate the potential of using topic modelling techniques for revealing relationships between patterns and finding patterns of pattern co-occurrence.
- Demonstrate examples of visual exploration of pattern distribution for data of distinct nature.

We begin with introducing the conceptual background in Section 2 followed by an overview of the related work in Section 3. We describe the work of the investigated analysis approach in two case studies in Section 4 and then discuss our experiences and findings (Section 5). Section 6 concludes the paper.

2. Background

2.1. Key concepts

In this research, we use the term *'episode'* to refer to a short period of time that has distinct properties while being a part of a larger series or process. This aligns with common definitions of an episode in dictionaries (e.g., [MW22]). Our research focuses on data describing changes occurring within episodes with non-zero duration. The data consist of series of attribute values referring to different times between the beginnings and ends of the episodes. While the attributes can be of any type (numeric, categorical, or spatial), our current research is focusing on attributes with numeric values. We refer to this type of data as *episode-based data*.

Our research goal is to find methods and develop a visual analytics workflow for analysing episode-based data

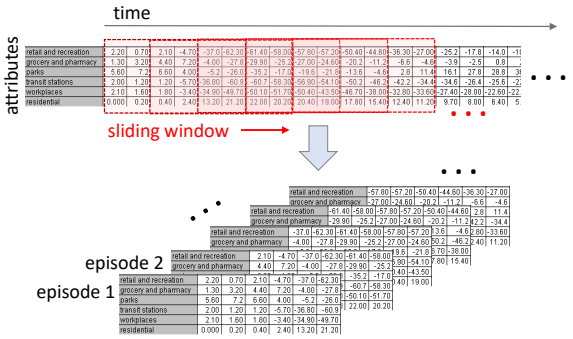


Figure 1: *Illustration of the division of a continuous multivariate time series into multiple episodes using a sliding time window.*

in order to understand the overall process or behaviour the episodes are parts of. Gaining a general understanding of a whole by observing its multiple parts requires abstraction. According to the theoretical model known as "pattern theory" [AAM*21], abstraction in data analysis is achieved by finding *patterns* in data distributions. A pattern is a combination of relationships between multiple data items that allows us to consider and represent all of these items jointly as a unit, which can be described without referring to any individual items.

Episode-based data have a hierarchical structure. The entire dataset consists of descriptions of multiple episodes, each of which is composed of time series of multiple attributes. Each time series consists of multiple attribute values and their corresponding time references. To gain a general understanding, we must perform abstraction from the elementary data items (attribute values and time references) up to patterns at the highest level of the hierarchy (the distribution of dynamic properties across the entire set of episodes). We believe this requires a step-wise ascent from the bottom to the top of the hierarchy. At the lowest level, patterns are made up of data elements; at higher levels, patterns are made up of patterns from the previous levels.

At the lowest level, patterns are jointly formed by (1) temporal relations between the time steps, (2) correspondences between time steps and attribute values, and (3) relations between the attribute values. The possible types of patterns include increase, decrease, peak, trough, constancy, and fluctuations.

At the second level of the hierarchy, patterns are formed by relations of co-occurrence of temporal patterns of individual attributes, i.e., the single-attribute patterns appear together in episodes. For example, increase of attribute A1 may tend to co-occur with constancy of attribute A2 and decrease of attribute A3.

At the third level of the hierarchy, one needs to consider

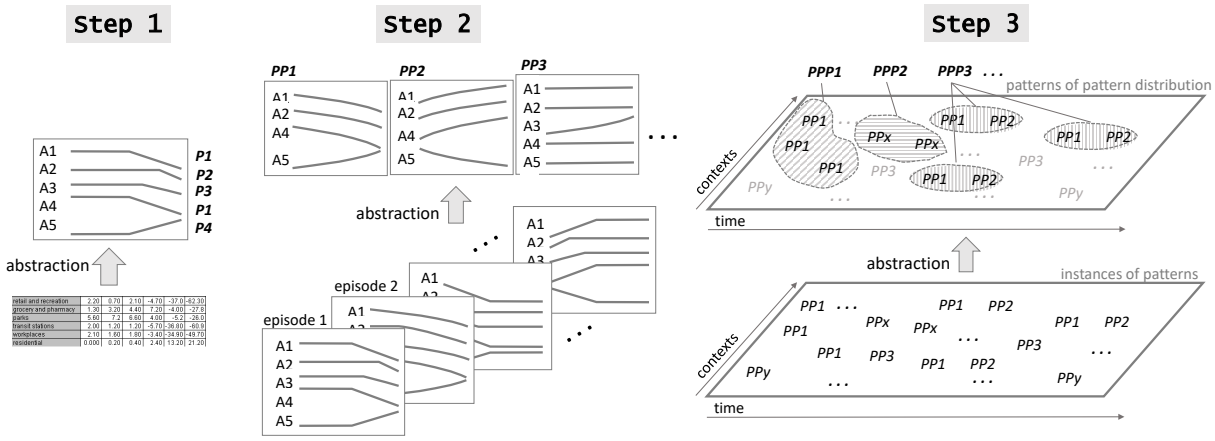


Figure 2: *A schematic representation of the idea of progressive abstraction. Step 1: Sequences of values of individual attributes are abstracted to single-attribute patterns of value variation. Step 2: Combinations of single-attribute patterns co-occurring in multiple episodes are abstracted to multi-attribute combination patterns, Step 3: Instances of combination patterns occurring throughout the data are abstracted to patterns of distribution of the combination patterns.*

the distribution of the second-level patterns over relevant data dimensions, one of which is always time (since the data are temporal). Third-level patterns of the temporal distribution are formed by the temporal relations between the positions of the second-level patterns in time. Other relevant dimensions depend on the nature of the phenomenon reflected in the data. The data describing episodes may include contextual information about the circumstances in which the episodes occurred, their spatial positions, and/or the actors involved. The set of all such contexts is the relevant data dimension. The distribution of the second-level patterns with respect to the set of contexts needs to be analysed. Third-level patterns are formed by the links of the second-level patterns to various context properties. For example, there may be tendencies for the combination patterns to occur or not occur in certain parts of space or under specific external conditions.

This theory-based analysis workflow is schematically represented in Fig. 2. In the first step of the workflow, the temporal sequences of attribute values in each episode are abstracted to single-attribute patterns $P1$, $P2$, ... representing the character of the value variation. In the second step, the combinations of the single-attribute patterns co-occurring in the episodes are analysed to find multi-attribute combination patterns denoted $PP1$, $PP2$, ... The notation PP emphasises that the combination patterns are super-patterns (i.e., patterns of a higher level of abstraction) with respect to the single-attribute patterns. After extracting the set of PP_i, the episodes are represented in terms of combinations of these super-patterns. In the third step, the distribution of the super-patterns PP_i over the set of episodes is analysed to find super-super-patterns PPP_j formed by relationships between the super-patterns resulting from their positions in the distribution. The distribution is schematically represented in Fig. 2, right, as a plane where one dimension is time and the other stands for the set of relevant contexts.

Fig. 2 represents the general idea of the progressive abstraction approach. It does not specify what methods can be utilised to fulfil the three steps of the workflow. In the following, we describe one of many possible ways to implement the approach. In Step 1, single-attribute value variations are represented by SAX patterns [LKWL07]. In Step 2, multi-attribute combination patterns are extracted by means of topic modelling methods [VK20]. In Step 3, patterns of the distribution of the combination patterns are discovered with the help of interactive visualisations.

2.2. Ideas for implementing the approach

According to pattern theory, one possible operation on discovered data patterns is to represent them in an aggregated manner so that they can be treated as single elements of data. This means that the combinations of data elements making up the patterns are replaced by aggregated representations, which can then be used in further analysis.

The pattern aggregation operation may be a part of an approach to finding multi-attribute combination patterns. After identifying single-attribute patterns, we can treat them as units and represent them with tokens. We can then replace the original time series of attribute values with tokens denoting the temporal patterns formed by these values. As a result, each episode is represented by a combination of pattern tokens. We can then apply a method that is suitable for analysing combinations of tokens and can find patterns formed by the tokens, such as recurring associations. A potentially suitable class of methods is topic modeling. The

topics generated by these methods are, in essence, multi-token patterns, which in our case can be interpreted as multi-attribute temporal variation patterns.

Once we have an aggregated representation of the multi-attribute combination patterns in the form of topics, we can use them in further analysis. We need to analyse the distribution of the integrated patterns (topics) across the dataset to find patterns at an even higher level of abstraction and establish relationships between them. To support this by visual analytics techniques, we need to find appropriate methods to visualise the distribution of the topics over the set of episodes.

2.3. Deriving single-attribute patterns

A temporal pattern of attribute values represents, in an aggregated form, the relationships between values arranged in a chronological sequence. These relationships may be *similar* or *different*, *larger* or *smaller*, *close* or *distant*, etc. To efficiently implement our workflow, we need a method for automatically transforming value sequences into aggregated representations of the patterns of value variation along the sequences. Essentially, we need a compact and simplified machine-readable representation of a time series that can be treated as a single object (a token) in the following steps of analysis. This means that the representation must be symbolic rather than numeric.

For time series of numeric values, there is a suitable representation called Symbolic Aggregate approXimation (SAX) [LKWL07]. SAX divides each time series into a specified number w of equal-sized segments and calculates the mean value in each segment. The probability or frequency distribution of the mean values is divided into α equiprobable parts, where α is the desired size of the alphabet, i.e., the number of symbols to be used for encoding the time series. Each part of the distribution is given a distinct symbol from the alphabet. The mean values of the time series segments are mapped to the symbols corresponding to the parts of the value distribution in which they fit. As a result, each time series is represented by a sequence consisting of w symbols from the alphabet.

The basic idea of the SAX representation method can also be applied to values of other types of attributes if it is possible to divide the value distribution into a small number of meaningful parts that can be represented by symbols from an alphabet. For example, when attribute values represent positions in space, the space can be divided into regions, the mean positions in segments of the time series can be calculated, and these positions can be encoded with symbols corresponding to the regions containing them. This allows for the creation of a compact and simplified machine-readable representation of a time series that can be treated as a single object in the following steps of analysis.

3. Related work

Related to our work are researches in the following areas: visual exploration of multivariate time series (MVTS) and event sequences, segmentation of MVTS, simplification of numeric time series, abstraction of temporal data, and application of topic modelling methods to non-textual data.

3.1. Visual exploration of multivariate time series

The most obvious approach to visualisation of MVTS is representation of the time series of the individual variables along a common time axis in a juxtaposed, superposed, stacked, or intertwined way [JME10, BHR*19]. Another widely used approach is to apply dimensionality reduction (DR) to the combinations of values of the variables and represent the time steps by points in a two-dimensional projection space, as, for example, in MotionTrack [HWX*10] or TimeCurves [BSH*16]. Bernard et al. [BWS*12] assign colours to the positions in the projection space and represent the temporal variation of the value combinations by variation of colours along the time axis.

Fujiwara et al. [FSS*21] deal with data consisting of multiple MVTS, such as measurements recorded in different geographic locations. Originally, the data have the form of a 3D tensor. It is transformed to a matrix where each row corresponds to one time step of one MVTS. After applying DR to this matrix and obtaining a 2D projection, selected clusters of points in the projection plot are represented by colour coding in various additional views supporting interpretation of the DR results. To analyse multiple MVTS of air pollution data from different locations, Kuo et al. [KFC*22] apply non-negative matrix factorisation (NMF) as a DR technique. NMF extracts combinations of chemicals that can be attributed to different sources of air pollution. The visual displays are designed to enable interpreting the results of NMF.

Algorithmic clustering of time steps with subsequent representation of the clusters by colours is also used in analysis of MVTS [GCML06]. DR can be applied to clustered and aggregated data [BWK*13].

It is worth noting that DR or clustering methods in all these works are applied to data associated with individual time steps, i.e., with time points lacking duration. Hence, each data item includes a single value of each attribute. Our approach to analysis of multivariate temporal data is based on dividing the data into episodes of non-zero duration. Each episode includes a sequence of values of each attribute. Analysis of such data requires different approaches.

In essence, episodes are *events*; hence, episode-based data comprise one or more sequences of events. Analysis of event sequences is an established research topic in visual analytics. Guo et al. [GGJ*22] present a comprehensive survey of the existing approaches and systems. Well known examples include LifeFlow [WGGP*11], OutFlow [WG12], and

EventFlow [MLL*13]. In all these works, events are considered as atomic objects without internal structure. The main focus of analysis is arrangement of events relative to each other, which is different from our focus on the variation of values of multiple attributes *within the episodes*. Eventpad [CvW18] is designed for analysis of sequences of multivariate events, where multiple attributes characterise each event as a whole. Differently from our work, dynamic attributes whose values vary during the event life times are not considered.

3.2. Segmentation of multivariate time series

Episodes can be obtained from time series data in many different ways. The simplest approach is to use a sliding time window that defines episodes of equal length (e.g., [STKF07]). The episodes may partly overlap in time [WG11] thus smoothing transitions between consecutive patterns of value variation. This method of deriving episodes is illustrated in Fig. 1. The resulting pieces of time series can be treated as multidimensional vectors to which clustering and/or dimensionality reduction (projection) methods can be applied [vWvS99, STKF07, WG11]. Other approaches define episodes based on events, for example, by taking temporal buffers before and/or after detected events or time intervals from one event to another; see Monroe et al. [MLL*13] for examples in basketball data analysis. TimeMask technique [AAC*17] proposes a powerful set of query operations for defining episodes. In this paper, we apply the sliding window approach in Section 4.1 and event-based definition of episodes in Section 4.2.

Episodes can also be obtained by dividing time series into semantically meaningful segments. There exist segmentation algorithms [GYD*19], which can be combined with interactive visual techniques [BDB*16, BBB*18]. Segmentation may also be done based on clustering of time steps [BWK*13]. An earlier work [AA23] considers the problem of dividing the time span of a complex dynamic phenomenon described by *multiple MVTS* into meaningful periods that enclose different relatively stable states or development trends. Here, the segmentation is applied to all MVTS taken together. The task is supported by a combination of clustering, aggregation, projection, and interactive visual tools for time division.

Statistical science develops methods for detecting multiple change points in a long univariate time series [NHZ16]. To utilise such methods for dividing MVTS, it is necessary to define the way of setting common breaks for all time series so as to take into account their individual change points.

3.3. Simplification of numeric time series

In our approach, short time series encapsulated in episodes are simplified and represented in an abstracted symbolic form using the SAX pattern method [LKWL07]. The method involves aggregation (averaging) of attribute values by subintervals and discretisation of the domain of the aggregated attribute values. Alternatively to aggregation, simplification of time series can be achieved by downsampling [CS10, Ste13], which represents a time series using a smaller number of time points while striving to preserve its shape. The Douglas-Peucker algorithm originally proposed for cartographic generalisation [DP73] can be used for the same purpose.

To discretise a numeric attribute, its value range is divided into bins by introducing several breaks. The ways to do this have been studied extensively in cartography (see a review by Slocum [SMKH22]) for designing classified choropleth maps. The most common approaches include natural breaks, equal length intervals, and equal size divisions. Jenks [Jen77] developed a method for calculating a statistically optimal classification. The geovisualisation research community developed various interactive procedures for human-controlled discretisation [AAK*21, SMKH22]. A number of discretisation techniques have been developed in data mining, see a review by Garcia et al. [GLS*13]. Beyond discretisation, an extensive survey of methods for simplification and compact representation of numeric time series that can be used in visualisation was done by Shurkhovetskyy et al. [SAAF18].

The authors of the SAX method [LKWL07] noted that symbolic representation of time series had not received much attention in the data mining research. More recently, Bondu et al. [BBC16] proposed a more advanced variant of the SAX method that optimises the division of the time series into sub-intervals. Shirato et al. [SAA21] divide episodes into equal intervals, as in the SAX method, but apply symbolic encoding to value trends (increase, decrease, or constancy) on the intervals rather than value aggregates.

Instead of directly working with time series as sequences of values, it may be suitable for many analysis tasks to extract features from them, i.e., derive attributes characterising some aspects of the entire time series. Lubba et al. [LSK*19] evaluated about 5000 diverse features that can be computed from time series and selected a subset of 22 features that exhibit strong classification performance across a given collection of time-series problems and are minimally redundant. There are multiple software libraries for feature extraction, e.g., [BFF*20]. A recent trend is automatic extraction of time series features by means of artificial neural networks [ZZYG21, CTMMB22]. While such features may work very well in machine learning tasks, they are not interpretable by humans. In contrast, our goal is to extract variation patterns that are meaningful to humans.

3.4. Abstraction of temporal data

While the term "abstraction" is often treated as a syn-

onym to "simplification" or "compression", we use this term in the sense of transforming data into meaningful higher-level concepts [Sha97]. According to Aigner et al. [AMM*08, AMST11], data abstractions are qualitative values or patterns that convey key ideas. Abstraction of temporal data may be supervised or unsupervised [HÖ2]. The former means finding time intervals where data have a priori defined properties and assigning corresponding labels from a given set. Thus, there are methods for detecting predefined shapes, such as increase, decrease, constancy, peak, etc. [HÖ2, SAA23]. Abstraction can be done using domain-specific rules [CC99, AMM*08] or definitions from a domain ontology [Sha97]. Unsupervised time abstraction means that concepts are not defined in advance but need to be learned from data. A general approach is to identify similar parts in the time series by means of clustering [vWvS99, HÖ2]. In our work, unsupervised abstraction is supported by visual representations of time series transformed to SAX patterns.

Progressive abstraction means deriving higher-level concepts from earlier extracted patterns based on relationships between the latter [AAM*21]. Complex patterns can be detected automatically according to predefined rules specifying patterns to be used as building blocks and temporal relationships between these blocks [SLCB07]. Shirato et al. [SAA23] propose interactive visual interfaces to identify patterns in the temporal distribution and co-occurrences of patterns of different attributes and transitions between patterns of the same attributes. In that work, higher-level patterns are constructed in the mind of the analyst. In our current work, the derivation of multi-attribute patterns from single-attribute patterns is supported by topic modelling techniques.

3.5. Applications of topic modelling beyond text analysis

Regardless of the encoding method, the output of the procedure is a set of tokens (words) that represent single-attribute patterns. Each episode is described by a combination of tokens for multiple attributes. To find recurrent associations of tokens, we apply topic modelling methods [VK20], such as Latent Dirichlet allocation (LDA) [BNJ03] and Non-negative Matrix Factorization (NMF) [LNC*17]. In essence, these methods perform dimensionality reduction: similarly to PCA [JC16], they generate higher level features (considered as "topics") as weighted combinations of input features, which may be, in particular, frequencies or measures of relevance of different words. Beyond analysing texts in natural languages, topic modelling methods have been successfully applied to different types of data, for example, DNA codes [LTD*16], software repositories [CTH16], taxi trips [CSZ*14], user activities in interaction with software systems [CAA*20], and team tactics in football [SAA21]. These examples demonstrate the versatility of the topic modelling methods. However, we are not aware of the uses of topic modelling for finding patterns in multivariate time series.

While there have been numerous comparative evaluation studies of different topic modelling methods in application to text data (e.g., [AYB20, EY22, AEG*23], we are not aware about studies comparing effectiveness of these algorithms for other data types. In our work, we did not plan to compare different methods. However, LDA, which worked quite well in the first case study, produced unsatisfactory results in the second case study. This lead us to try NMF, which fulfilled its task successfully.

The two case studies in which we tested our ideas and analysis workflow are described in the following section.

4. Implementation and testing of the approach

In the first case study, we deal with data on population mobility behaviours in different countries during the COVID-19 pandemic. In the second case study, we apply our approach to episodes extracted from football game tracking data.

4.1. Case study 1: COVID-19 mobility trends

We use a subset of the publicly available data set of mobility trends available in the Google's COVID-19 Open Data Repository [Goo22]. Google collects anonymised data from apps such as Google Maps to track changes in people's movements during the pandemic. The data consist of daily visitor numbers to specific categories of places (e.g., grocery stores, parks, train stations, etc.) relative to baseline days before the pandemic outbreak. Baseline days represent a normal value for each day of the week and are given as the median value over the five-week period from January 3^{rd} to February 6^{th} 2020. The rationale for using a specific normal value for each day of the week is that people usually have different routines on weekends versus weekdays. The data thus consist of the deviations from the normal values expressed in percent of the normal values. Positive values signify increased numbers of visits to a certain category of places and negative values have the opposite meaning.

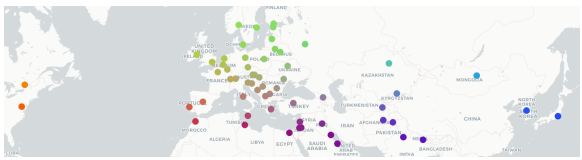


Figure 3: *The countries involved in the case study are represented on a map by their capitals.*

For this case study, we used a subset of data containing information on 60 countries in Europe, Asia, and North America (as shown in Fig. 3). The data covers a 496-day period from February 17, 2020 (Monday) to June 27, 2021 (Sunday); see Fig. 4. It is divided into 69 overlapping episodes of

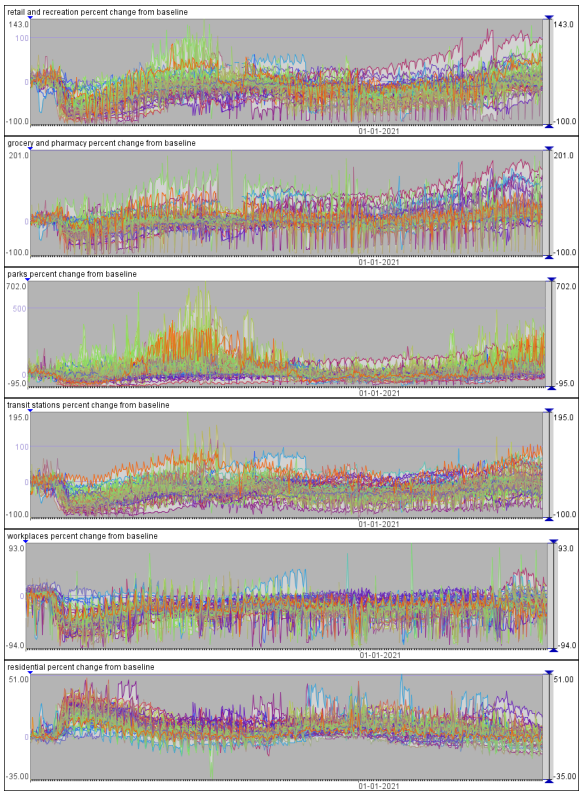


Figure 4: *The entire time series of the mobility indicators by the countries are shown on line plots. The lines have the same colours as the corresponding dots on the map showing the positions of the countries capitals (Fig. 3).*

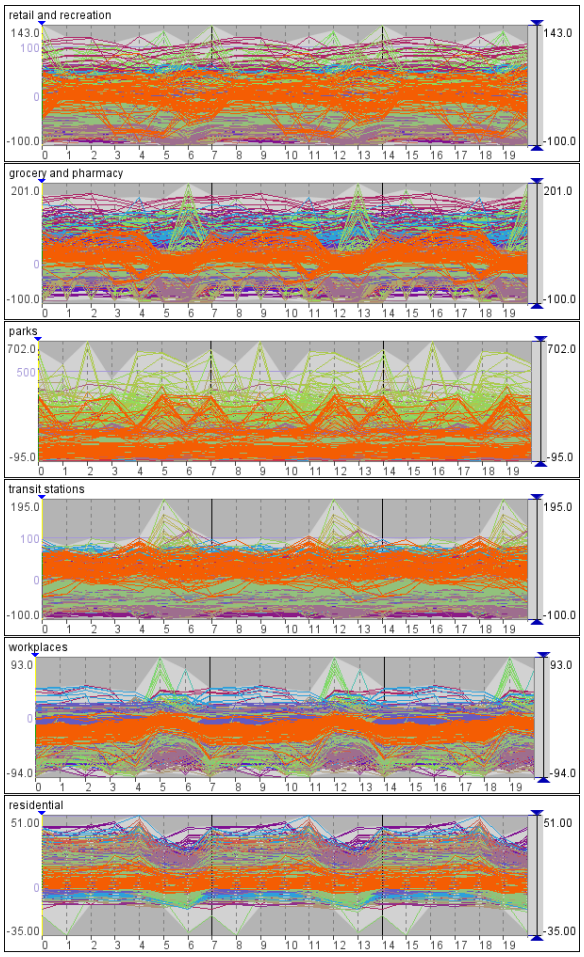


Figure 5: *The line plots show the time series of the mobility indicators by the episodes. The lines have the colours that have been earlier assigned to the countries (Fig. 3). The time steps are from 0 to 20 according to the number of days passed since the beginning time of each episode.*

21 days (3 weeks) each, with a 7-day shift between consecutive episodes. This means that each episode overlaps with the previous one by 14 days. Each episode represents data for one country. The full set consists of 4130 episodes, while 10 are missing due to gaps in the data. Fig. 5 shows the 21-day time series for the mobility indicators within each episode.

4.1.1. Step 1: generating single-attribute patterns

We represent the episode-based time series by SAX patterns of length 5 using the alphabet $\{a,b,c,d,e\}$, where a corresponds to the lowest value interval and e to the highest value interval. In doing that, we skip the values for Saturdays and Sundays to disregard the irrelevant weekly variations of the mobility behaviours and consider the general trends over the 3-weeks time periods. Table 2 shows the breaks by which the value ranges of the attributes have been automatically divided into 5 bins so that each bin includes approximately 20% of the values. To visualise the patterns, we apply colour coding to the symbols of the alphabet that has been used in pattern generation. We use a diverging colour scale [HB03] from dark blue to dark red to encode the symbols corresponding to the value intervals ordered from the lowest to the highest. Fig. 6 demonstrates the representation of the patterns.

4.1.2. Step 2: obtaining multi-attribute patterns

We create pseudo-texts composed of the SAX patterns preceded by the abbreviated attribute names (e.g., 'residential' is abbreviated as 'home') and apply the topic modelling algorithm LDA to the resulting strings. After experimenting with the parameter k (number of topics), we find that $k = 9$ gives an acceptable result in terms of topic interpretability while the topics are not too numerous. The table display in Fig. 7 shows the patterns that have certain weights in the topics, the minimal weight (in our example it is 0.005) being set through the slider at the bottom of the display. The interpre-

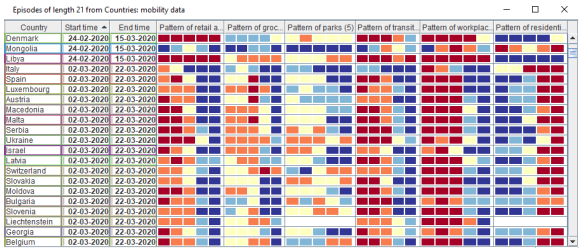


Figure 6: *A fragment of a table showing colour-coded SAX patterns of the mobility indicators in the episodes. The patterns of length 5 have been generated using the alphabet $\{a,b,c,d,e\}$, where a corresponds to the lowest value interval and e to the highest value interval. The symbols are represented using a diverging colour scheme from dark blue for a to dark red for e.*

tations of the topics that can be derived from this display are listed in Table 1.

Table 1: *Interpretation of topics.*

N	Interpretation
0	more or less usual life
1,3,4	average to high mobility, reduced staying at home
2,5	average to low mobility, increased staying at home
7	decreasing mobility, increasing staying at home
6,8	strict "stay at home"

Figure 7: *The topics resulting from applying the LDA algorithm.*

It can be noted that there are groups of topics that can be interpreted similarly. The explanation is that there are multiple symbolic codes corresponding to pattern variants that have very close meanings for a human. For example, the codes *cdddd*, *cccdd*, *ddeee* and quite many others represent different variants of the pattern of increase. However, for the topic modelling algorithm, these are distinct and unrelated terms, which may belong to different topics. Therefore, running the algorithm with a lower value of the parameter k will not automatically unite semantically similar topics.

Another observation is that the topics are not "clean" in terms of including similar or consistent patterns of the same attribute. Consider, for example, the patterns of the attribute 'home' in the topic 7. Almost all patterns represent average to high values and only one pattern represents values from the lowest range. Such inconsistent mixtures of patterns occur irrespective of the chosen number of topics k. To investigate this phenomenon in more detail, we select (by means of interactive filtering) the episodes with the symbolic pattern *aaaaa* of the attribute 'home' for which topic 7 has the highest weight among all topics. The 20 episodes satisfying the query are shown in a table view in Fig. 8. We see that there are 9 episodes where the lowest values of presence at home co-occurred with quite low levels of presence in the other categories of places, which seems counter-intuitive. Interestingly, 8 of these episodes took place in Moldova. A possible reason may be that people did not frequently use their mobile devices while staying at home, which lead to under-estimation of the people's presence. This reminds us that data may be biased and require caution in interpreting analysis results and making inferences.

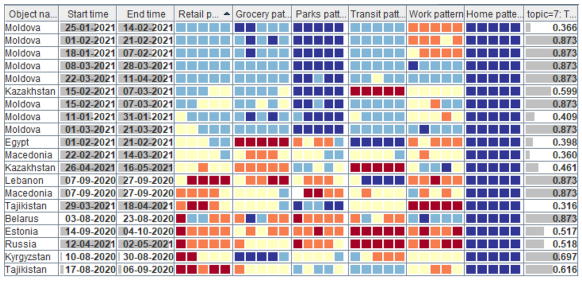


Figure 8: *A selected subset of episodes where topic 7 has the highest weight and the pattern of attribute 'home' is aaaaa. The rows of the table are arranged in the increasing order of the patterns of the attribute 'retail'.*

To verify and, if appropriate, refine the interpretations of the topics, it is useful to consider the groups of episodes with different dominant topics (i.e., having the highest weight) and the attribute patterns occurring in these groups of episodes. Since the episodes are numerous, the patterns need to be represented in an aggregated way. A possible way of aggregated representation of SAX patterns is demonstrated in Fig. 9. The patterns of one attribute are aggregated by counting the occurrences of each symbol $\{a,b,c,d,e\}$ on each position from 1 to 5. The result is represented by a segmented bar chart where each bar corresponds to one position of a pattern and its segments represent the proportions of occurrences of the symbols $\{a,b,c,d,e\}$ in this position. The

segments are painted in the colours that have been assigned to the symbols.

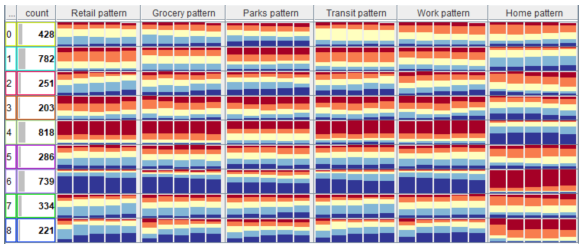


Figure 9: *The mobility patterns in the groups of episodes with different dominant topics are shown in an aggregated way by segmented bars.*

The display in Fig. 9 confirms the interpretations of the topics and topic groups. Thus, the bar charts of the groups of episodes with dominant topics 1, 3, and 4 have large proportions of red (representing high values) for all place categories except home, which has large proportions of blue. The bars for the groups of episodes with dominant topics 6 and 8, on the opposite, have high amounts of blue for all places except home and high amounts of red for home. We also see differences between the topics within the groups in terms of the proportions of different colours; however, these differences can be treated as inessential for the interpretation.

4.1.3. Step 3: understanding the distribution of the multi-attribute patterns

After obtaining and interpreting the topics, we want to investigate the contexts in which they occur. This includes the distribution of the topics over the set of countries and the time and their relationships to the pandemic spread indicators, such as the mortality due to COVID-19. To obtain a convenient visual representation of the context, we create an artificial matrix space (Fig. 10) where the rows correspond to the countries and the columns to the start times of the episodes. To put the countries in a linear order, we apply the Principal Component method to the spatial positions of the country capitals and take the ordering based on the first component, as suggested by Wulms et al. [WBM*21]. With this approach, close spatial positions tend to receive close positions in the linear order. The episodes referring to each country are arranged chronologically in the corresponding row. The vertical line in Fig. 10 approximately marks the time of Christmas. It is impossible to mark any date in this view precisely because the horizontal positions correspond not to individual days but to temporally overlapping episodes of 21 days length.

In this matrix space, we visualise the pandemic-caused mortality rates by shading from white for zero to dark brown for the highest values. The display in Fig. 10 reveals prominent spatio-temporal patterns in the variation of the mortality rates. Thus, we observe high mortality rates in several

Figure 10: *The episodes are arranged in a matrix space with the rows corresponding to the countries and columns to the starting times of the episodes. The background colouring of the matrix represents the distribution of the average daily counts of the deaths due to COVID-19 (Source of the data: [Goo22]).*

countries of Europe (Ireland, Spain, UK, Belgium, Italy, and Sweden) in March and April 2020, relatively low values in the summer of 2020 and an increase of the deaths rates in many countries starting from October 2020.

Now we need to visualise the distribution of the topics in this context space. One possible approach is to use pie charts with sector sizes representing the topic weights for the episodes. Fig. 11 shows a fragment of such a display. Please note that the topics have been assigned distinct colours, which are shown in the caption of the table display in Fig. 7. These colours are used for painting the corresponding sectors of the pies. The purpose of this representation is not to enable accurate perception of individual topic weights or estimation of the weight proportions (pie charts are commonly judged as poorly suited for these tasks [CM84]) but to provide an overall view of the distribution of the topic colours. To gain an overview, it is best to look at the display from a distance without paying much attention to the individual diagrams but instead perceiving the patterns of the overall colour distribution. This is possible due to the associative property of the visual variable 'colour' [Ber83].

Thus, we notice display areas with high amounts of lilac representing topic 6, which is interpreted as strict "stay at home" regime. This topic prevails in the spring of 2020 and re-occurs in some countries in the winter and spring of 2021 when the death rates increase. The blue colour of topic 8, which is also interpreted as strict staying at home, occurs in the same periods as the lilac of topic 6, sometimes as sectors of the same pies. Topic 2 and 5 (red and magenta), both inter-

Figure 11: *A fragment of the matrix display with topic weights represented by pie charts.*

preted as decreased mobility and increased staying at home, often occur close in time to topics 6 and 8. High amounts of cyan representing topic 1 (average to high mobility and reduced staying at home) are observed in the summer of 2020 and also in the summer of 2021. The light green of topic 4 and brown of topic 3 also frequently occur in the same periods as topic 1.

Generally, we observe that topics with similar interpretations tend to have close positions in the matrix space, which reinforces our confidence that the topics are semantically close. This gives us a ground to aggregate semantically close topics, namely, unite the groups of topics $\{1,3,4\}$, $\{2,5\}$, and $\{6,7,8\}$. Technically, the aggregation is done by summing up the weights of the topics of each group for each episode. Fig. 12 demonstrates the appearance of the matrix display with pies after the aggregation. Here, the colours of the topics 1, 2, and 6 are assigned to the aggregates in which these topics are included.

The aggregation simplifies the perception of the patterns of colour distribution across the display. Nevertheless, to see the distributions of the individual (original or aggregated) topics more clearly, it may be beneficial to use a small multiple display as shown in Fig. 13. For each topic, there is a separate matrix where the topic weights for the episodes are represented by proportional sizes of circle symbols.

Again, this view is meant not for estimation of individual values but for perceiving all circles in a matrix at once, in one instance of sight, i.e., as a single image [Ber83]. In so doing, we can see very prominently that the combined topics $1+3+4$ (increased mobility) and $6+7+8$ (increased staying at home) have complementary distribution patterns. The former occurs where and when the mortality rates are low, except for the early period of the pandemic spread (starting from mid-March), when lockdown regimes were introduced even in the countries whose local death rates had not yet significantly increased; see `https://en.wikipedia.org/wiki/COVID-19_pandemic_in_Europe`. The combined topic $6+7+8$, on the opposite, occurs almost everywhere in the period starting from mid-March and reoccurs after the summer of 2020 when the death rates increase in the majority of the countries. For the topics 0 and $2+5$, the small multiples display does not reveal obviously interpretable patterns. To understand the distribution of these topics, it is better to use the pie chart display (Fig. 12). It shows that these topics tend to have intermediate positions between the periods of staying at home and periods of high mobility.

4.1.4. Lessons learnt

This investigation showed us that symbolic encoding of numeric time series, which involves division of the value range into bins, requires attention and, preferably, control from a human analyst. While there exists a sound rationale for dividing values into equal-frequency intervals [LKWL07], the analyst should examine results of automatic division to be able to interpret the codes correctly. In our case study, the middle value interval represented by the symbol c could be wrongly interpreted as values around zero, i.e., close to the pre-pandemic levels. In reality, this interval includes values around the median, which may significantly differ from zero. The meaning of the symbol c is defined by the interval breaks shown in the columns 2 and 3 of Table 2. While this division was suitable for our experimental study, there may be applications requiring involvement of domain knowledge and/or adopted conventions in the discretisation of attribute value ranges. This can be enabled by interaction techniques that allow the effects of different divisions on the resulting symbolic patterns to be observed.

Another important lesson is that semantically close symbolic patterns are treated as completely different and unrelated by a topic modelling algorithm, which leads to generation of multiple topics with similar meanings from a human perspective. This reveals a need in interactive post-processing of topic modelling results, which includes merging of semantically close topics, as we did in Fig. 12. It may also be appropriate to edit some topics to make their meanings clearer by modifying the weights of specific patterns.

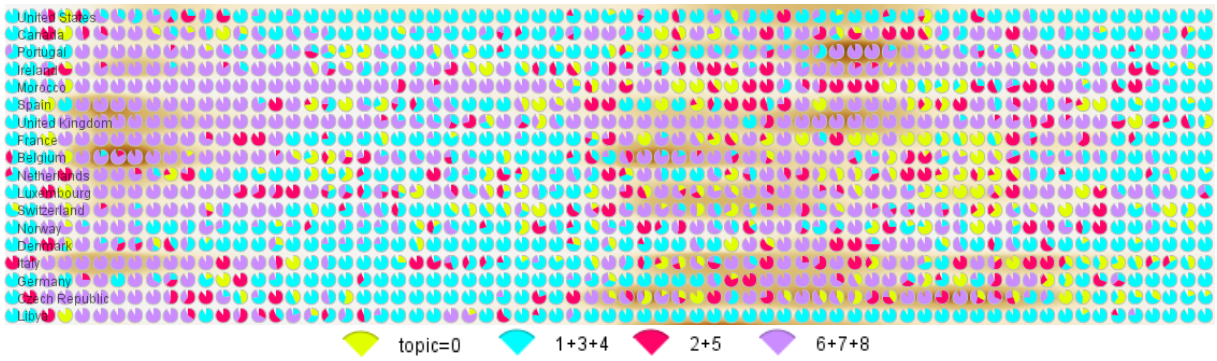


Figure 12: *A fragment of the matrix display with pie charts representing the weights of aggregated topics.*

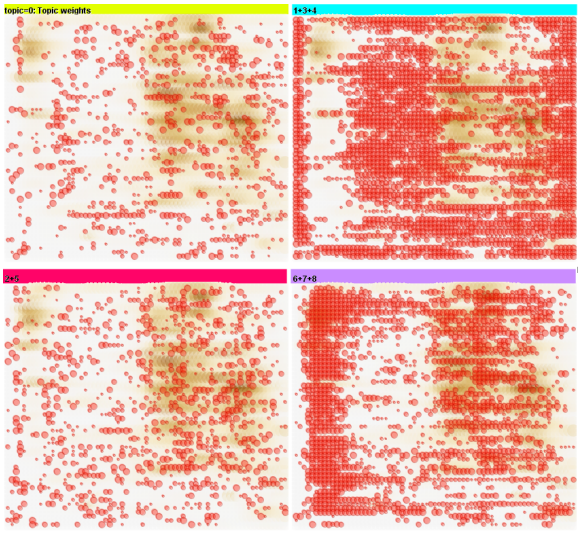


Figure 13: *A small multiple display where each matrix represents the distribution of the weights of one original or aggregated topic by proportional circle sizes.*

In our case, topic 7 could be edited by setting the weight of the pattern *aaaaa* for 'home' to zero (the weights of the remaining patterns should in this case be adjusted to make 1 in total). After such editing, the topic weights for the episodes need to be re-calculated.

4.2. Case study 2: Team behaviours in football

In this case study, we used tracking data from two football games of the German Bundesliga season 2019-2020, such that the same home team played against different guest teams. We used the original data, which included the players' and ball's trajectories, to derive time series of the following attributes that are used by FIFA to indicate team performance during a game [FIF22]:

- pressure of the defending players on the ball [AAB*17];
- pressure of the defending players on the attackers;
- percent of the attacking players in the final third of the pitch;
- depth and width of the home and guest teams on the pitch;
- stretch index of the home and guest teams;
- mean and minimal distance of the attacking players to the opponents' goal;
- minimal X-distance (i.e., distance along the pitch) of the defending players to their own goal.

The time resolution of the data is 25 steps per second, i.e., the time interval between consecutive time steps is 40 milliseconds.

After excluding the time intervals when the ball was out of play from the time series, we extracted episodes of the length (duration) of 10 seconds starting at the moment of ball possession change as well as episodes starting 10 seconds before the ball possession change, i.e., the first and last 10 seconds on one team's ball possession. We skipped the time intervals where the ball possession of one team lasted for less than 9 seconds; however, intervals of very short ball possession (less than 1 second) were treated as parts of longer episodes of ball possession of the opponent team.

In the result, we got 250 episodes of possession start and 249 episodes of possession end, 499 episodes in total. Due to the way of episode extraction, episodes from the two categories may be overlapping or even coinciding in time. Thus, there are 21 duplicated episodes belonging to both categories. Overlapping of episode times and duplication of episodes are acceptable for our analysis, where we want to reveal differences (if any) between team behaviours at the beginning and at the end of one team's ball possession. Particularly, we want to see how teams begin their attacks, how defenders behave in response to that, and what is happening before defenders re-gain the ball. Still, we exclude the second instances of the 21 duplicated episodes from the further consideration to avoid their excessive impact on the results of topic modelling.

4.2.1. Step 1: generating single-attribute patterns

In the episodes, we transform the time series of the attributes specifying team extents (depth, width, and stretch index) and distances to the goals into time series of changes (i.e., differences) with respect to the values at the beginning of the episodes. Then we generate SAX patterns of length 4 using the same alphabet as in the first use case. Table 3 shows the breaks of the attribute value ranges that were applied for encoding the values by the symbols. A sample of the patterns can be seen in Fig. 14. As in the previous case study, we construct strings including the patterns of all 13 attributes preceded by abbreviated attribute names.

4.2.2. Step 2: obtaining multi-attribute patterns

In this case study, we compared the work of two topic modelling algorithms: Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorisation (NMF). We ran each of the methods with the same value of the parameter k (number of topics). Fig. 15 shows the topics constructed by LDA and NMF for $k = 8$. The topics are represented by combinations of patterns having high weights.

It can be seen in Fig. 15 that the topics produced by LDA (left table) are less clear than the topics resulting from NMF (right table). Thus, many cells in the left table contain both patterns of value increase (represented by shades of red) and patterns of value decrease (represented by shades of blue). This complicates interpretation of the topics. We also see that LDA did not give significant weights to the patterns of the attribute 'pressure on the ball', which mismatches our knowledge that exerting pressure on the ball is an important defensive tactics in football. The NMF topics, in contrast, are differentiated in terms of the pressure on the ball: topics 2, 3, and 6 are characterised by low pressure, topic 0 by moderate pressure, topic 4 by high pressure, and topics 1 and 7 by increasing pressure.

What concerns the patterns of the other attributes, we see a variety of attacking and defensive tactics. Thus, attackers may increase or decrease the team's depth and width on the pitch, and defenders tend to behave similarly. There are topics (0, 3, and 6) in which the ball moves away from the goal of the defending team and the attackers do not approach the opponents' goal and do not increase their presence in the final third of the pitch. The other topics represent more active attacks and corresponding defensive behaviours, including increased pressure on the ball and attackers when they approach the goal of the defending team. However, to understand better the behaviours represented by the topics, we need to see them in the context of the pitch in connection to the episodes.

4.2.3. Step 3: understanding the circumstances of the multi-attribute patterns

The data we analyse require a domain-specific representation of the episodes and their topic weights. For this purpose, we use maps with the background representing the football pitch. The data have been transformed so that the goal of the home team is always on the left and the goal of the guest team on the right. We represent the episodes on the maps by vectors (directed lines) connecting the first and last positions of the ball, as can be seen in Fig. 16. We apply colour coding to show which team possesses the ball in the episode and whether it is at the start or at the end of the ball possession; see the legend at the bottom of Fig. 16. The weights of the topics are represented by proportional widths and, simultaneously, opacity levels of the vector lines. The redundant encoding of the weights improves the perception.

The position, orientation, and length of a vector not only show the overall relocation of the ball but also give some hints about the character of the episode: whether it was a swift attack along the pitch, or seeking a possibility for an attack while passing the ball across the pitch, or maintaining the ball possession while staying close to own goal, or offensive activities near the opponents' goal. We want to see whether these types of episodes would be distinguishable in terms of the topic weights.

Fig. 16 contains two sets of small multiple maps. Each map represents the weights of one topic (represented by the background colour of the map caption) by line widths and opacity levels. The eight maps on the left correspond to the LDA-generated topics and the eight maps on the right to the NMF topics. The two sets of maps look very different. On the left, all vectors seem to have the same width. Indeed, for more than 76% of the episodes, the dominant LDA topic has the same weight 0.9375. Also for each of the remaining episodes, there is one LDA topic with a very high weight while the weights of the other LDA topics are close to zero. The maps showing the weights of the LDA topics look very chaotic. We do not see any clear patterns in terms of the types and characteristics of the episodes. Each map shows a disorderly looking mixture of vectors of different origins, lengths, and orientations.

Different from what we see on the left, the brighter and thicker vectors in each map on the right have some features in common. In the map for topic 0, the vectors originate from a common point in the centre of the pitch. Many of them represent ball movements away from the defended goal, and this is consistent with the representation of topic 0 in the topic table in Fig. 15, bottom. In contrast, the map for topic 1 highlights episodes (mostly at the end of ball possession, as signified by the line colours) with great advancements of the ball towards the target, which agrees with the patterns of the change of the ball and attackers distances to the goal shown in the topic table. The table also says that these episodes are characterised by the attackers increasing their presence in the opponents' third of the pitch, the defenders increasing their pressure on the ball and attackers, and both teams increasing their depth and decreasing widths. Similar behavioural patterns are observed in topics 5 and 7, and the

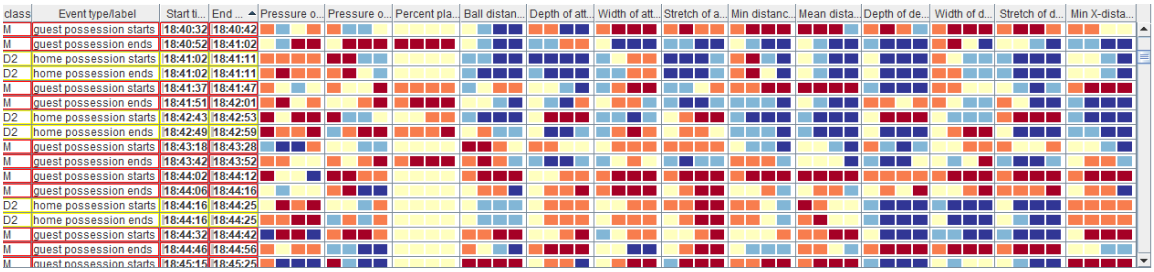


Figure 14: *A fragment of a table showing colour-coded SAX patterns of the football episodes. The patterns of length 4 were generated using the same alphabet as in the first case study.*

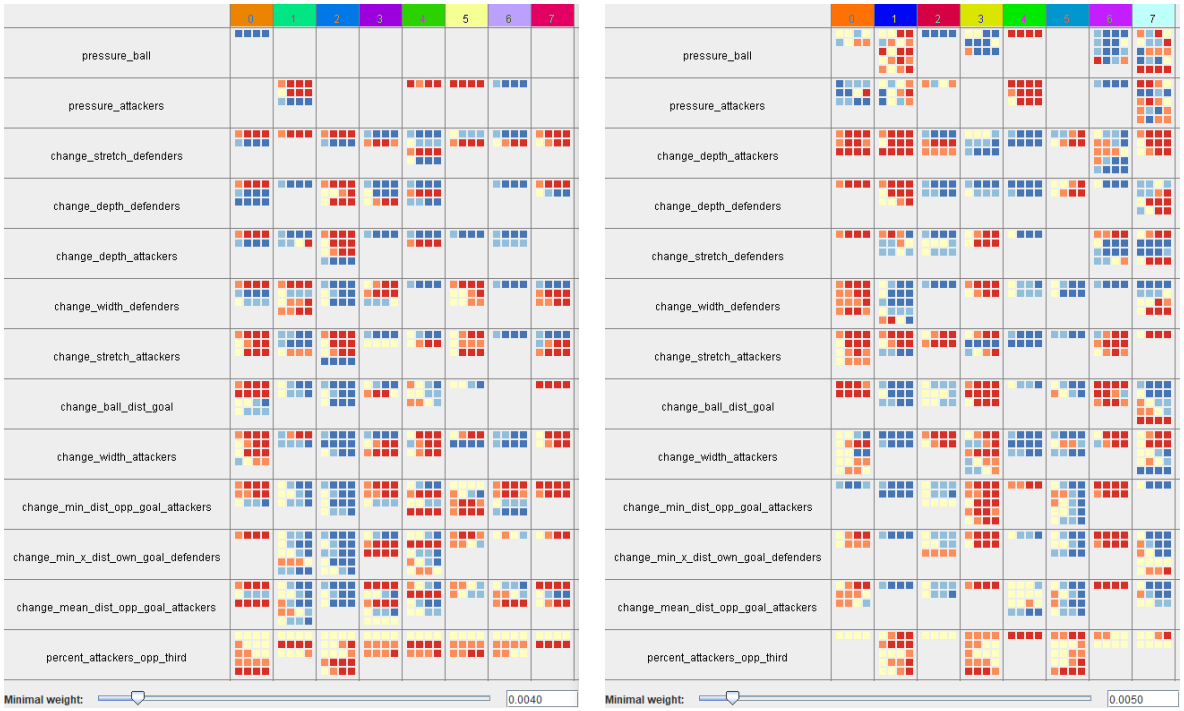


Figure 15: *Topics extracted from the football episodes by means of LDA (left) and NMF (right).*

vectors that are prominent in the corresponding maps also look similar to those in the map for topic 1.

Topic 3 expresses behaviours that are opposite to topics 1, 5, and 7. They are represented on the map by vectors oriented across the pitch and painted in the colours corresponding to the beginnings of ball possession. Topic 2 characterises episodes in which teams' ball possession begins close to their own goals. They slowly move towards the opponents' goal stretching across the pitch while the opponents make their team more compact preparing to defend. Topic 4, in contrast, characterises the behaviours of the teams in dramatic situations when the ball is close to the target. Both teams get more compact, and the defenders exert high pressure on the ball and attackers. Topic 6 reflects somewhat relaxed behaviours when ball possession begins close to the pitch centre, and the possessing teams move the ball closer to their own goal while stretching in width. In response, the defenders decrease their pressure on the ball while following the attackers' retreat and making their team more compact.

The maps in Fig. 16 do not show clearly whether any topics prevail more for the home team or for the guest team. We would also like to compare the team behaviours in the two games for which we have data. We remind that the home team (Borussia Dortmund) was the same in both games. The guest team in the first game was FC Nuremberg and in the second game Bayern Munich. Assuming that the dominant topics of the episodes represent the main features of the teams' behaviours, we create a display of the co-occurrences

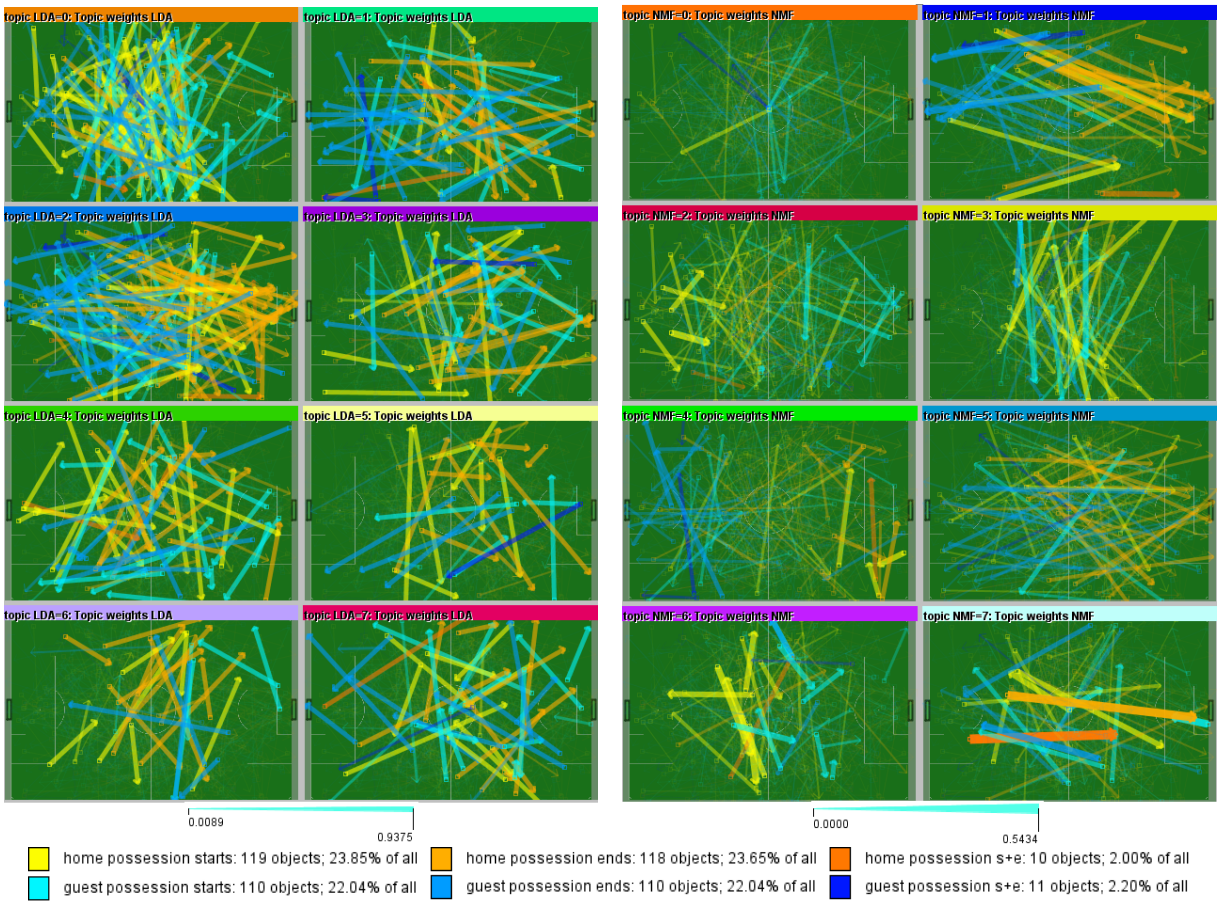


Figure 16: *The weights of the topics for the football episodes are represented on small multiple maps where each map shows the weights of one topic. The background of each map represents the football pitch. The episodes are represented by vectors (directed lines) connecting the initial and final positions of the ball. The line widths and opacity levels are proportional to the topic weights. The colours correspond to the types of the episodes. The images on the left and right show the weights of the LDA and NMF topics, respectively.*

of the dominant topics and the episode types; see Fig. 17. On the top, the display shows the co-occurrences in both games, and the two screenshots below show the co-occurrences in the first and second games. The display consists of eight bar charts (for the eight topics) with horizontal bars oriented from right to left. The upper bar in a bar chart shows in how many episodes in total the corresponding topic was dominant. The following six bars show the frequencies of the dominance of this topic for the six types of episodes.

From the three instances of the co-occurrence display visible in Fig. 17, we learn that topic 0 was rarely dominant in the episodes with the home team's possession. We also observe differences between the two games, especially in the episodes with the guest team's possession. Under the guests' possession, the topics from 3 to 7 were dominant much more frequently in the second game than in the first. This may mean that the guest team of the first game did not vary its attacking behaviour as much as the guest team of the second game. It can also be noticed that topic 4 (green), which we interpreted as fighting close to the target, was very rarely dominant under the guest team's possession in the first game, whereas in the second game it prevailed much more frequently in the episodes with the guest team's possession than under the possession of the home team. Hence, Bayern Munich quite frequently created dangerous episodes at the goal of Borussia Dortmund. Similar whilst not so striking differences exist for topic 5 (light blue) characterised by the ball and the attackers approaching the target. This behaviour was more frequent under the home team's possession in game 1 and under the guest team's possession in game 2.

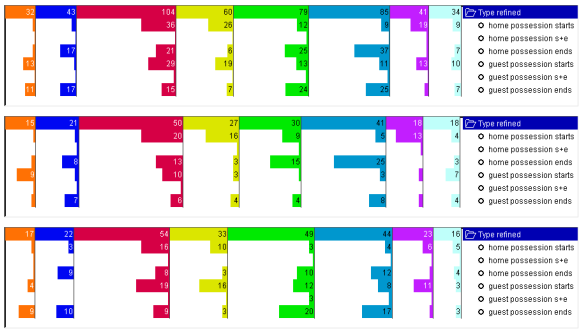


Figure 17: *Distribution of the dominant topics for the episode types in the whole dataset (top), data from game 1 (middle), and data from game 2 (bottom).*

4.2.4. Lessons learnt

This experiment showed us that different methods of topic modelling may produce very different results and that some results may not be very useful. It is questionable whether the success or failure of a given algorithm for a given dataset can be predicted. There have been comparative studies of the efficacy of LDA and NMF in application to short texts, such as tweets. In some studies, the results of the methods were assessed as equally good [AYB20], whereas other researchers found that topics produced by NMF were more in line with human judgment [EY22]. However, whatever results may be obtained for text data, they are not necessarily transferable to non-textual applications of topic modelling. Therefore, if the chosen method does not produce an acceptable result, it may be worth trying another method.

Another useful lesson concerns preparation of the data. As we mentioned in Section 4.2.1, we transformed the original values of some of the attributes into differences from the values at the beginnings of the episodes and used the time series of the differences to generate the SAX patterns. We applied this transformation after an unsuccessful attempt to use the original time series. The resulting topics were trivial, mainly distinguishing the episodes based on the players' distances from the goals and not revealing differences in team behaviours. Therefore, transforming absolute values into changes can be crucial in analysing behaviours. It is worth noting that the attribute values we had in the first case study were already provided as differences from the baseline values, which allowed us to analyse behaviour changes with respect to the pre-pandemic period.

5. Discussion

5.1. Application of the pattern theory

In this work, we aimed to find a way to derive a general understanding of a phenomenon reflected in episode-based data, which consist of multivariate time series encapsulated in episodes. To obtain an overall view of the phenomenon as a whole from elementary data (i.e., attribute values and time references), high abstraction is required. We applied the pattern theory [AAM*21], which posits that abstraction in data analysis is achieved through the discovery of patterns formed by relationships between data items. We developed an approach to analysis that incrementally increases the level of abstraction, beginning with the relationships between elementary data items (i.e., attribute values and time references) that form temporal variation patterns of individual attributes. To achieve the next level of abstraction, these patterns are treated as elements, and relationships between them (specifically, co-occurrence) are considered. We represented single-attribute patterns as tokens and used techniques such as topic modeling to discover patterns of token co-occurrence. However, to gain an overall understanding, the level of abstraction needs to be further increased by discovering patterns in the distribution of these token co-occurrence patterns over time and in relevant contexts.

Using the pattern theory, we devised an abstract analysis workflow that includes three steps of abstraction (Fig. 2). We defined the types of patterns to be discovered in each step and the types of relationships that are involved in these patterns. This abstract workflow is transformed into a concrete work plan by choosing methods that will be used to implement each abstract operation. We chose the SAX pattern method [LKWL07] for the first step of the analysis workflow, topic modelling [VK20] for the second step, and interactive visualisations of topic distributions for the third step. We implemented this work plan for two different datasets reflecting phenomena of distinct nature and scale and found the approach to be effective. However, it is important to note that other implementations of the abstract workflow may be possible, as discussed below.

5.2. Design space in implementing the abstract workflow

Step 0: Data pre-processing. If the original data have the form of continuous time series rather than episodes, they need to be transformed to episodes. This can be done using any of the existing approaches, e.g, one of those mentioned in Section 3.2. We applied the sliding window approach in the first case study and event-based definition of episodes in the second. Domain knowledge may be involved in defining episodes, e.g., to ignore irrelevant weekly fluctuations, as in the first study, or out-of-play times, as in the second study. Besides division into episodes, data preprocessing may include data cleaning, missing value imputation, smoothing, transforming absolute values to relative, aggregation, re-sampling, etc.

Step 1: Deriving single-attribute patterns. In this step, the task is to transform each temporal sequence of attribute values into an object that can be represented both visually, to enable human interpretation, and as a symbolic token or word,

to enable computational processing in the second step. Apart from the SAX method [LKWL07] that we used in our studies, a variety of possibilities exist. One of them is to detect predefined shapes [HÖ2, SAA23] and represent them visually as shapes and symbolically by words 'increase', 'decrease', 'peak', etc. Another possibility is to transform the original values to changes with respect to the previous or initial value apply the SAX method to the transformed data [SAA21]. Domain-specific rules [CC99, AMM*08] or a domain ontology [Sha97] can be employed to assign human-understandable labels to time series, and these labels can represent the patterns in the following analysis. Generally, any approach producing interpretable codes or meaningful labels is suitable for this step.

Step 2: Deriving multi-attribute patterns. Here, the symbolic representations of single-attribute patterns serve as an input to a method capable to detect repeated co-occurrences of the patterns within episodes. Topic modelling methods, namely, LDA [BNJ03] and NMF [LNC*17], proved to be suitable for this purpose. While these methods are the most popular, there are many other topic modelling methods that can be potentially applied. Multiple surveys [KB18, VK20, AEG*23] discuss properties and capabilities of different methods, so that an analyst can make an informed choice. Apart from topic modelling, re-occurring combinations of single-attribute patterns represented by labels can be detected using various algorithms designed for frequent item set mining [HCXY07, LFVV19]. It should be noted, however, that these algorithms tend to produce an excessive number of patterns, which may be very challenging for the following exploration.

It may be interesting to try network analysis, specifically, community detection methods [JYL*18]. The input may be a graph with vertices corresponding to the single-attribute patterns and weighted edges connecting patterns that occurred together, the weights being the counts of the joint occurrences. Network analysis, however, provides a different kind of information than we obtained using topic modelling. It reveals strong pairwise associations, but existence of a community including three or more patterns does not necessarily mean that all these patterns often occur together.

Due to these limitations and inconveniences of the network analysis and item set mining methods, we consider topic modelling to be a better tool for fulfilling the second step of data abstraction.

Step 3: Finding patterns of distribution of multi-attribute patterns. Implementation of this step is data- and domain-specific. We propose to support this step by visualisations designed according to the nature of the data and analysis goals. The key idea is to colour-code the second-level patterns and use these colours to represent the patterns in visual displays. If the analysis goals require considering the distribution of the patterns over time, the task can be supported by variants of a time line display, as we did in our first study. The episodes are positioned along the time axis according to their existence times. Since several patterns may be associated with one episode, the combination of these pattern can be represented by a diagram or glyph consisting of elements painted in the colours of the patterns. This can be considered as a basic design for exploring the temporal distribution of patterns. Our visualisations provide examples of using the second display dimension to represent a relevant aspect of the context in which the episodes occur. In our case, it is spatial location (country), but it is also possible to represent other kinds of context information. When the distribution of the patterns with respect to temporal cycles is of interest, polar coordinates can be used instead of Cartesian.

In our second study, the distribution of the patterns over time was not important for the analysis. We wanted to see how the second-level patterns, i.e., the topics, are related to spatial properties of the episodes, which was the relevant type of contextual information. We used a small multiples display with one panel for each topic. Within the panels, we visualised the relevant properties (namely, the ball possession and displacement vector) of the episodes significantly associated with the corresponding topics. The small multiples is a general design that can be applied to different types of data, while the visualisations within the panels depends on the nature of the episodes and analysis goals. The topic weights for the episodes shown in the panels can be represented by a suitable visual variable; we used line widths.

Hence, the general design recommendations for visually supporting the third step of abstraction include (a) timeline display of episodes, possibly, with an additional dimension representing some aspect of the context; (b) circular display with polar coordinates representing temporal positions of episodes; (c) diagrams or glyphs showing topic composition for the episodes; (d) small multiples display with panels corresponding to topics and application-specific representation of properties of the episodes associated with the topics.

5.3. Technical aspects of computational methods

In the following section, we will discuss the methods we have used in our implementation of the abstract workflow.

5.3.1. SAX encoding

The Symbolic Aggregate approXimation (SAX) method [LKWL07] divides the time series into equal segments and computes a single numeric value for each segment. This requires deciding how many parts to select and which aggregation function to use. The number of parts is selected according to the desired level of detail in representing a time series. In our first use case, we divided the time series into 5 segments, resulting in one aggregated value representing 3 original daily values (we remind that we took episodes consisting of the weekdays of three consecutive weeks, i.e., 15

days in total). In the second use case, we used 4 segments to represent attribute dynamics over 10 seconds.

The SAX method typically uses either the mean or median as the aggregation function, but different application domains may require different approaches. For example, in Schreck et al's work on financial time series analysis [STKF07], the raw data included stock sell transactions with amounts and prices. The transactions for each stock have been aggregated by daily intervals into the average price and total volume. Domain experts may suggest other aggregation methods such as opening and closing prices, minimum and maximum prices for the day, or the difference/ratio between opening and closing prices or between maximum and minimum values. Some of these aggregates were used by Shirato et al. [SAA21] to represent trends over parts of time series data.

After aggregating the values, the next step in the SAX method is to symbolically encode the aggregate values based on their frequency distribution. Two decisions must be made at this stage: the size of the alphabet (i.e., the number of bins to divide the distribution into) and the breaks in the range of aggregate values that determine how the distribution is divided into bins. The number of bins affects the level of detail in the resulting representation. In our case studies, we used 5 bins to represent values around the average, values moderately lower and moderately higher than the average, and values much lower and much higher. The classical SAX method uses equal-frequency breaks, but it may be useful to consider other options based on domain knowledge and the semantics of the attributes, as we discussed in connection with our first case study.

5.3.2. Topic modelling

There are two key questions to consider when using a topic modeling method: which method to choose and how to set its parameters. A detailed review of available methods can be found in [VK20]. It is worth noting that Non-Negative Matrix Factorization (NMF) has been found to be more effective than Latent Dirichlet Allocation (LDA) for short texts [AYB20, EY22], but it is unclear if this applies to non-textual data. In our case studies, LDA worked well in the first study and poorly in the second, even though the "texts" (i.e., symbolic descriptions of the episodes) were longer in the second study. It is worth keeping in mind that LDA is a probabilistic method. It works better with large amounts of "texts" (i.e., large number of episodes), so that term probabilities could be more reliably estimated. The number of episodes was quite low in our second study, which may be the reason of the failure of the LDA method. To determine the target number of topics, we used the approach proposed by Chen et al. [CAA*20]: running the selected method with different parameters, projecting the topics generated in the different runs into a single embedding space using a dimensionality reduction method, and exploring the topic distribution in this space, which is expected to reflect the similarities and differences between the topics. The number of visible distinct clusters in the projection suggests the potentially suitable number of topics. However, the main criterion is interpretability of the topics by a human. As it cannot be formally evaluated, we do not see a feasible way to fully automate the selection of the suitable number of topics.

5.3.3. Visualisation techniques

The abstract analytical workflow (shown in Fig. 2) is designed for a human analyst to gain insights into the behaviour of a phenomenon. All steps in the workflow require human reasoning, which should be supported by appropriate visualisations of relevant information. As the workflow is defined abstractly, it does not specify which visualisations should be used, but it suggests the types of information that need to be visualised: (1) single-attribute variation patterns, (2) multi-attribute combination patterns, and (3) the distribution of multi-attribute combination patterns over a set of episodes.

In our example implementation, we chose to use colour encoding for the SAX representation of the single-attribute patterns because the visual variable "colour" has a strong association capacity [Ber83], allowing multiple coloured stripes drawn close together to be efficiently perceived as a single image. This helps with the effective perception of tables displaying multiple SAX patterns, including tables of episodes (Figs. 6, 8, 14) and tables of topics (Figs. 7, 15). In contrast, the visual variable "shape" is not associative [Ber83], so representing patterns by shapes instead of coloured stripes would require inefficient scanning and memorisation of individual shapes, making it more difficult to interpret the overall display. Another advantage of using colour-coding is the ability to represent SAX patterns summarised by groups of episodes in the form of segmented bars, as shown in Fig. 9.

We also chose to use the visual variable 'colour' to represent topics, with each topic being encoded by a distinct colour. This allows us to use charts with colored segments to represent topic mixtures. Specifically, we used pie charts, which are compact, easily perceived as units rather than conglomerates of distinct elements, and enable the estimation of relative weights of the topics. We used the selective power of the visual variable 'colour' [Ber83] to differentiate the topics and the associative power of this variable to support the perception of topic distribution patterns from displays with multiple pie charts, as shown in Figs. 11 and 12.

The ways to visualise the distribution of topics over episodes depend on the organisation of the set of episodes and the patterns that can be expected based on domain knowledge. In our first case study, the episodes refer to different times over a long period and to different entities (countries). To visualise the distribution, we created a matrix with columns and rows corresponding to the times and

entities and placed pie charts in the cells. We also used background shading of the cells to represent some aspect of the relevant context, enabling the investigation of the distribution in relation to the context. This kind of visualisation would not be useful in the second case study, where team tactics constantly vary during a game and no meaningful temporal patterns of topic distribution can be expected. Instead, it is more appropriate to investigate how the topics are related to spatial properties of the episodes. This motivated the visualisation of the episodes in the space of the football pitch. We created a small multiple display to show the distribution of the weights of each topic over the episodes (Fig. 16). We also found small multiple displays to be useful in the first case study (Fig. 13). As stated in Section 5.1, timeline and small multiples are basic designs that can be used for episodes of various kinds.

A limitation of all displays we used is low scalability regarding the number of topics. However, a large number of topics may also be problematic for human interpretation and analysis; therefore, an analyst should strive to generate the minimal number of topics that are easily distinguished and well understood. Another problem may be a large number of episodes, which are hard to visualise without display clutter. A possible approach to alleviate this problem is aggregation of episodes and increasing the level of detail when the user zooms in or filters the data.

5.3.4. Software implementation

For our studies, we utilised the implementation of the topic modelling methods from the *scikit-learn 1.3.1* [PVG*11] Python library. For the data processing and visualisation, we used our in-house system V-Analytics [AAB*13], which has been developed by the authors over many years. Researchers interested in the latest version of V-Analytics are welcome to contact the authors; however, there are also state-of-the-art libraries available, such as MovingPandas [Gra19], that offer similar processing capabilities. For visualizations, Python libraries like Plotly and Bokeh, as well as the JavaScript library D3 [BOH11], can be utilised. The specific details of our software go beyond the scope of this paper, as our primary focus is on presenting the abstract workflow and providing examples of its implementation.

6. Conclusion

In our work, we applied a theoretical model [AAM*21] to develop an abstract general approach to analysing the type of data in which a set of episodes is characterised by multiple time-variant attributes. This approach involves incrementally increasing the level of data abstraction by merging multiple elements into patterns. We implemented this approach by selecting specific methods for each step and tested the resulting workflow in two case studies. In particular, we evaluated the usefulness of topic modeling methods for deriving multi-attribute combination patterns from patterns of temporal variation of individual attributes. Topic modelling has proved to be useful in two distinct case studies and can therefore be recommended for this kind of tasks.

At a broader level, our work demonstrates the feasibility and value of a theory-based approach for devising data analysis workflows and choosing appropriate methods to implement them. In the future, we plan to continue applying theoretical models proposed for visual analytics, such as theories of data patterns [AAM*21], knowledge generation [SSS*14], model building [ALA*18], and qualitative analysis [KHL21], to different types of data. Our goal is not only to find effective ways to analyse data, but also to identify and demonstrate the prescriptive potential of these primarily descriptive theoretical models.

References

[AA23] ANDRIENKO N., ANDRIENKO G.: It's about time: Analytical time periodization. *Computer Graphics Forum n/a*, n/a (2023). doi:10.1111/cgf.14845. 5

[AAB*13] ANDRIENKO G., ANDRIENKO N., BAK P., KEIM D., WROBEL S.: *Visual analytics of movement*. Springer Science & Business Media, 2013. doi:10.1007/978-3-642-37583-5. 18

[AAB*17] ANDRIENKO G., ANDRIENKO N., BUDZIAK G., DYKES J., FUCHS G., VON LANDESBERGER T., WEBER H.: Visual analysis of pressure in football. *Data Mining and Knowledge Discovery 31*, 6 (2017), 1793–1839. doi:10.1007/s10618-017-0513-2. 11

[AAC*17] ANDRIENKO N., ANDRIENKO G., CAMOSSI E., CLARAMUNT C., CORDERO-GARCIA J. M., FUCHS G., HADZAGIC M., JOUSSELME A.-L., RAY C., SCARLATTI D., VOUROS. G.: Visual exploration of movement and event data with interactive time masks. *Visual Informatics 1*, 1 (2017), 25 – 39. doi:10.1016/j.visinf.2017.01.004. 5

[AAK*21] ANDRIENKO G., ANDRIENKO N., KURESHI I., LEE K., SMITH I., STAYKOVA T.: Automating and utilising equal-distribution data classification. *International Journal of Cartography 7*, 1 (2021), 100–115. doi:10.1080/23729333.2020.1863000. 5

[AAM*21] ANDRIENKO N., ANDRIENKO G., MIKSCH S., SCHUMANN H., WROBEL S.: A theoretical model for pattern discovery in visual analytics. *Visual Informatics 5*, 1 (2021), 23–42. doi:10.1016/j.visinf.2020.12.002. 2, 6, 15, 18

[AEG*23] ABDELRAZEK A., EID Y., GAWISH E., MEDHAT W., HASSAN A.: Topic modeling algorithms and applications: A survey. *Information Systems 112* (2023), 102–131. doi:10.1016/j.is.2022.102131. 1, 6, 16

[ALA*18] ANDRIENKO N., LAMMARSCH T., ANDRIENKO G., FUCHS G., KEIM D., MIKSCH S., RIND A.: Viewing visual analytics as model building. *Computer Graphics Forum 37*, 6 (2018), 275–299. doi:10.1111/cgf.13324. 18

[AMM*08] AIGNER W., MIKSCH S., MÜLLER W., SCHUMANN H., TOMINSKI C.: Visual methods for analyzing time-oriented data. *IEEE Transactions on Visualization and Computer Graphics 14*, 1 (2008), 47–60. doi:10.1109/TVCG.2007.70415. 6, 16

[AMST11] AIGNER W., MIKSCH S., SCHUMANN H., TOMINSKI C.: *Visualization of time-oriented data*. Springer, 2011. doi:10.1007/978-0-85729-079-3. 6

[AYB20] ALBALAWI R., YEAP T. H., BENYOUCEF M.: Using topic modeling methods for short-text data: A comparative analysis. *Frontiers in Artificial Intelligence 3* (2020). doi:10.3389/frai.2020.00042. 6, 15, 17

[BBB*18] BERNARD J., BORS C., BÖGL M., EICHNER C., GSCHWANDTNER T., MIKSCH S., SCHUMANN H., KOHLHAMMER J.: Combining the automated segmentation and visual analysis of multivariate time series. In *EuroVA@ EuroVis* (2018), pp. 49–53. 5

[BBC16] BONDU A., BOULLÉ M., CORNUÉJOLS A.: Symbolic representation of time series: A hierarchical coclustering formalization. In *Advanced Analysis and Learning on Temporal Data* (Cham, 2016), Douzal-Chouakria A., Vilar J. A., Marteau P.-F., (Eds.), Springer International Publishing, pp. 3–16. doi:10.1007/978-3-319-44412-3_1. 5

[BDB*16] BERNARD J., DOBERMANN E., BÖGL M., RÖHLIG M., VÖGELE A., KOHLHAMMER J.: Visual-interactive segmentation of multivariate time series. In *Proceedings of the EuroVis Workshop on Visual Analytics* (Goslar, DEU, 2016), Eurographics Association, p. 31–35. 5

[Ber83] BERTIN J.: *Semiology of Graphics*. University of Wisconsin Press, 1983. 9, 10, 17

[BFF*20] BARANDAS M., FOLGADO D., FERNANDES L., SANTOS S., ABREU M., BOTA P., LIU H., SCHULTZ T., GAMBOA H.: TSFEL: Time series feature extraction library. *SoftwareX 11* (2020), 100456. doi:10.1016/j.softx.2020.100456. 5

[BHR*19] BERNARD J., HUTTER M., REINEMUTH H., PFEIFER H., BORS C., KOHLHAMMER J.: Visual-interactive preprocessing of multivariate time series data. *Computer Graphics Forum 38*, 3 (2019), 401–412. doi:https://doi.org/10.1111/cgf.13698. 4

[BNJ03] BLEI D. M., NG A. Y., JORDAN M. I.: Latent Dirichlet Allocation. *Journal of Machine Learning Research 3*, Jan (2003), 993–1022. 6, 16

[BOH11] BOSTOCK M., OGIEVETSKY V., HEER J.: D3: Data-driven documents. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)* (2011). URL: http://vis.stanford.edu/papers/d3. 18

[BSH*16] BACH B., SHI C., HEULOT N., MADHYASTHA T., GRABOWSKI T., DRAGICEVIC P.: Time curves: Folding time to visualize patterns of temporal evolution in data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (Jan 2016), 559–568. doi:10.1109/TVCG.2015.2467851. 4

[BWK*13] BERNARD J., WILHELM N., KRÜGER B., MAY T., SCHRECK T., KOHLHAMMER J.: Motionexplorer: Exploratory search in human motion capture data based on hierarchical aggregation. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2257–2266. doi:10.1109/TVCG.2013.178. 4, 5

[BWS*12] BERNARD J., WILHELM N., SCHERER M., MAY T., SCHRECK T.: TimeSeriesPaths : Projection-based explorative analysis of multivariate time series data. In *Journal of WSCG* (2012), pp. 97–106. 4

[CAA*20] CHEN S., ANDRIENKO N., ANDRIENKO G., ADILOVA L., BARLET J., KINDERMANN J., NGUYEN P. H., THONNARD O., TURKAY C.: Lda ensembles for interactive exploration and categorization of behaviors. *IEEE Transactions on Visualization and Computer Graphics 26*, 9 (2020), 2775–2792. doi:10.1109/TVCG.2019.2904069. 6, 17

[CC99] COMBI C., CHITTARO L.: Abstraction on clinical data sequences: an object-oriented data model and a query language based on the event calculus. *Artificial Intelligence in Medicine 17*, 3 (1999), 271–301. doi:https://doi.org/10.1016/S0933-3657(99)00022-6. 6, 16

[CM84] CLEVELAND W. S., MCGILL R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), 531–554. doi:10.1080/01621459.1984.10478080. 9

[CS10] CHIA C., SYED Z.: Using adaptive downsampling to compare time series with warping. In *ICDMW 2010, The 10th IEEE International Conference on Data Mining Workshops* (2010), Fan W., Hsu W., Webb G. I., Liu B., Zhang C., Gunopulos D., Wu X., (Eds.), IEEE Computer Society, pp. 1304–1311. doi:10.1109/ICDMW.2010.94. 5

[CSZ*14] CHU D., SHEETS D. A., ZHAO Y., WU Y., YANG J., ZHENG M., CHEN G.: Visualizing hidden themes of taxi movement with semantic transformation. In *2014 IEEE Pacific Visualization Symposium* (March 2014), pp. 137–144. doi:10.1109/PacificVis.2014.50. 6

[CTH16] CHEN T.-H., THOMAS S. W., HASSAN A. E.: A survey on the use of topic models when mining software repositories. *Empirical Software Engineering 21*, 5 (2016), 1843–1919. doi:10.1007/s10664-015-9402-8. 6

[CTMMB22] CHERNIKOV A., TAN C. W., MONTERO-MANSO P., BERGMEIR C.: Frans: Automatic feature extraction for time series forecasting, 09 2022. doi:10.48550/arXiv.2209.07018. 5

[CvW18] CAPPERS B. C., VAN WIJK J. J.: Exploring multivariate event sequences using rules, aggregations, and selections. *IEEE Transactions on Visualization and Computer Graphics 24*, 1 (2018), 532–541. doi:10.1109/TVCG.2017.2745278. 5

[DP73] DOUGLAS D. H., PEUCKER T. K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization 10*, 2 (1973), 112–122. doi:10.1002/9780470669488.ch2. 5

[EY22] EGGER R., YU J.: A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in Sociology 7* (2022). doi:10.3389/fsoc.2022.886498. 6, 15, 17

[FIF22] FIFA: Football intelligence, 2022. [accessed 26=December-2022]. URL: https://www.fifa.com/technical/football-technology/media-releases/fifa-to-introduce-enhanced-football-intelligence-at-fifa-11

[FSS*21] FUJIWARA T., SHILPIKA, SAKAMOTO N., NONAKA J., YAMAMOTO K., MA K.-L.: A visual analytics framework for reviewing multivariate time-series data with dimensionality reduction. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2021), 1601–1611. doi:10.1109/TVCG.2020.3028889. 4

[GCML06] GUO D., CHEN J., MACEACHREN A., LIAO K.: A visualization system for space-time and multivariate patterns (VIS-STAMP). *IEEE Transactions on Visualization and Computer Graphics 12*, 6 (2006), 1461–1474. doi:10.1109/TVCG.2006.84. 4

[GGJ*22] GUO Y., GUO S., JIN Z., KAUL S., GOTZ D., CAO N.: Survey on visual analysis of event sequence data. *IEEE Transactions on Visualization and Computer Graphics 28*, 12 (2022), 5091–5112. doi:10.1109/TVCG.2021.3100413. 4

[GLS*13] García S., Luengo J., Sáez J. A., López V., Herrera F.: A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering 25*, 4 (2013), 734–750. doi:10.1109/TKDE.2012.35. 5

[Goo22] Google: Google COVID-19 open data repository, 2022. [accessed 7-September-2022]. URL: https://health.google.com/covid-19/open-data/. 6, 9

[Gra19] Graser A.: MovingPandas: Efficient structures for movement data in python. *GI_Forum 1* (2019), 54–68. doi:10.1553/giscience2019_01_s54. 18

[GYD*19] Gharghabi S., Yeh C.-C. M., Ding Y., Ding W., Hibbing P., LaMunion S., Kaplan A., Crouter S. E., Keogh E.: Domain agnostic online semantic segmentation for multi-dimensional time series. *Data mining and knowledge discovery 33*, 1 (2019), 96–130. doi:10.1007/s10618-018-0589-3. 5

[HÖ2] Höppner F.: Time series abstraction methods - a survey. In *Informatik Bewegt: Informatik 2002 - 32. Jahrestagung Der Gesellschaft Für Informatik e.v. (GI)* (2002), GI, p. 777–786. 6, 16

[HB03] Harrower M., Brewer C. A.: Colorbrewer.org: An online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (June 2003), 27–37. doi:10.1179/000870403235002042. 7

[HCXY07] Han J., Cheng H., Xin D., Yan X.: Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery 15*, 1 (2007), 55–86. 16

[HWX*10] Hu Y., Wu S., Xia S., Fu J., Chen W.: Motion track: Visualizing variations of human motion data. In *IEEE Pacific Visualization Symposium (PacificVis)* (2010), pp. 153–160. doi:10.1109/PACIFICVIS.2010.5429596. 4

[JC16] Jolliffe I. T., Cadima J.: Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 374*, 2065 (2016), 20150202. doi:10.1098/rsta.2015.0202. 6

[Jen77] Jenks G. F.: Optimal data classification for choropleth maps. *Department of Geographiy, University of Kansas Occasional Paper* (1977). 5

[JME10] Javed W., McDonnel B., Elmqvist N.: Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 927–934. doi:10.1109/TVCG.2010.162. 4

[JYL*18] Javed M. A., Younis M. S., Latif S., Qadir J., Baig A.: Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications 108* (2018), 87–111. 16

[KB18] Kherwa P., Bansal P.: Topic modeling: A comprehensive review. *ICST Transactions on Scalable Information Systems 7* (07 2018), 159623. doi:10.4108/eai.13-7-2018.159623. 16

[KFC*22] Kuo Y.-H., Fujiwara T., Chou C. C., Chen C., Ma K.-L.: A machine-learning-aided visual analysis workflow for investigating air pollution data. *2022 IEEE 15th Pacific Visualization Symposium (PacificVis)* (2022), 91–100. 4

[KHL21] Karer B., Hagen H., Lehmann D. J.: Insight beyond numbers: The impact of qualitative factors on visual data analysis. *IEEE Transactions on Visualization and Computer Graphics 27*, 2 (2021), 1011–1021. doi:10.1109/TVCG.2020.3030376. 18

[LFVV19] Luna J. M., Fournier-Viger P., Ventura S.: Frequent itemset mining: A 25 years review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 9*, 6 (2019), e1329. 16

[LKWL07] Lin J., Keogh E., Wei L., Lonardi S.: Experiencing SAX: a novel symbolic representation of time series. *Data Min. Knowl. Discov. 15*, 2 (Oct. 2007), 107–144. 3, 4, 5, 10, 15, 16

[LNC*17] Luo M., Nie F., Chang X., Yang Y., Hauptmann A., Zheng Q.: Probabilistic non-negative matrix factorization and its robust extensions for topic modeling. In *Thirty-first AAAI conference on artificial intelligence* (2017). 6, 16

[LSK*19] Lubba C. H., Sethi S. S., Knaute P., Schultz S. R., Fulcher B. D., Jones N. S.: Catch22: Canonical time-series characteristics: Selected through highly comparative time-series analysis. *Data Min. Knowl. Discov. 33*, 6 (nov 2019), 1821–1852. doi:10.1007/s10618-019-00647-x. 5

[LTD*16] Liu L., Tang L., Dong W., Yao S., Zhou W.: An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus 5*, 1 (2016), 1–22. doi:10.1186/s40064-016-3252-8. 6

[MLL*13] Monroe M., Lan R., Lee H., Plaisant C., Shneiderman B.: Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2227–2236. doi:10.1109/TVCG.2013.200. 5

[MW22] Merriam-Webster: episode, 2022. [accessed 26=December-2022]. URL: https://www.merriam-webster.com/dictionary/episode. 2

[NHZ16] Niu Y. S., Hao N., Zhang H.: Multiple change-point detection: A selective overview. *Statistical Science 31*, 4 (2016), 611–623. 5

[PVG*11] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830. 18

[SAA21] Shirato G., Andrienko N., Andrienko G.: What are the topics in football? Extracting time-series topics from game episodes. *IEEE VIS 2021 poster* (2021). URL: http://geoanalytics.net/and/papers/vis21poster.pdf. 5, 6, 16, 17

[SAA23] Shirato G., Andrienko N., Andrienko G.: Identifying, exploring, and interpreting time series shapes in multivariate time intervals. *Visual Informatics 7*, 1 (2023), 77–91. doi:10.1016/j.visinf.2023.01.001. 6, 16

[SAAF18] Shurkhovetskyy G., Andrienko N., Andrienko G., Fuchs G.: Data abstraction for visualizing large time series. *Computer Graphics Forum 37*, 1 (2018), 125–144. doi:10.1111/cgf.13237. 5

[Sha97] Shahar Y.: A framework for knowledge-based temporal abstraction. *Artificial Intelligence 90*, 1 (1997), 79–133. doi:10.1016/S0004-3702(96)00025-2. 6, 16

[SLCB07] Sacchi L., Larizza C., Combi C., Bellazzi R.: Data mining with temporal abstractions: Learning rules from time series. *Data Mining and Knowledge Discovery 15*, 2 (oct 2007), 217–247. doi:10.1007/s10618-007-0077-7. 6

[SMKH22] Slocum T. A., McMaster R. B., Kessler F. C., Howard H. H.: *Thematic cartography and geovisualization*. CRC Press, 2022. 5

[SSS*14] SACHA D., STOFFEL A., STOFFEL F., KWON B. C., ELLIS G., KEIM D. A.: Knowledge generation model for visual analytics. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1604–1613. doi:10.1109/TVCG.2014.2346481. 18

[Ste13] STEINARSSON S.: *Downsampling Time Series for Visual Representation*. PhD thesis, University of Iceland, 2013. 5

[STKF07] SCHRECK T., TEKUŠOVÁ T., KOHLHAMMER J., FELLNER D.: Trajectory-based visual analysis of large financial time series data. *SIGKDD Explor. Newsl. 9*, 2 (dec 2007), 30–37. doi:10.1145/1345448.1345454. 5, 17

[VK20] VAYANSKY I., KUMAR S. A.: A review of topic modeling methods. *Information Systems 94* (2020), 101582. doi:10.1016/j.is.2020.101582. 1, 3, 6, 15, 16, 17

[vWvS99] VAN WIJK J. J., VAN SELOW E. R.: Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99)* (Oct. 1999), pp. 4–9. 5, 6

[WBM*21] WULMS J., BUCHMÜLLER J., MEULEMANS W., VERBEEK K., SPECKMANN B.: Stable visual summaries for trajectory collections. In *IEEE 14th Pacific Visualization Symposium (PacificVis)* (2021), pp. 61–70. doi:10.1109/PacificVis52677.2021.00016. 9

[WG11] WARD M. O., GUO Z.: Visual Exploration of Time-Series Data with Shape Space Projections. *Computer Graphics Forum 30*, 3 (2011), 701–710. doi:10.1111/j.1467-8659.2011.01919.x. 5

[WG12] WONGSUPHASAWAT K., GOTZ D.: Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2659–2668. doi:10.1109/TVCG.2012.225. 4

[WGGP*11] WONGSUPHASAWAT K., GUERRA GÓMEZ J. A., PLAISANT C., WANG T. D., TAIEB-MAIMON M., SHNEIDERMAN B.: Lifeflow: Visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2011), CHI '11, Association for Computing Machinery, p. 1747–1756. doi:10.1145/1978942.1979196. 4

[ZZYG21] ZHOU F., ZHOU H., YANG Z., GU L.: If2cnn: Towards non-stationary time series feature extraction by integrating iterative filtering and convolutional neural networks. *Expert Systems with Applications 170* (2021), 114527. doi:10.1016/j.eswa.2020.114527. 5

Appendix A: Appendix

Table 2: *Breaks in the division of the attribute value ranges into bins. Case study 1: COVID-19 mobility trends*

Attribute	**0** *(min)*	**1**	**2**	**3**	**4**	**5** *(max)*
retail & recreation	-100	-40	-22	-10	1	143
grocery & pharmacy	-98	-13	-3	4	16	170
parks	-93	-23	-3	16	52	646
transit stations	-100	-45	-30	-17	-2	96
workplaces	-93	-41	-28	-20	-11	45
residential	-21	1	5	9	14	43

Table 3: *Breaks in the division of the attribute value ranges into bins in the football case study*

Attribute	**0** *(min)*	**1**	**2**	**3**	**4**	**5** *(max)*
Pressure on ball	0.000	1.699	11.550	23.672	40.572	173.684
Pressure on attackers	5.196	117.818	156.291	188.216	233.835	487.682
Percent attackers in opp. third	0.000	0.000	0.000	20.000	47.302	100.000
Change of ball distance to defense goal	-71.930	-20.194	-7.406	-0.630	8.139	48.713
Change of depth of attackers	-29.193	-3.138	-0.600	1.140	4.478	26.445
Change of width of attackers	-36.476	-1.985	0.734	3.283	8.365	33.848
Change of stretch of attackers	-7.953	-0.653	0.147	0.861	2.421	10.193
Change of min distance of attackers to opp. goal	-54.045	-7.787	-1.880	0.408	3.286	22.679
Change of mean distance of attackers to opp. goal	-49.126	-7.222	-2.161	-0.047	1.946	23.352
Change of depth of defenders	-28.273	-4.085	-1.088	1.054	4.694	25.950
Change of width of defenders	-28.794	-5.664	-1.711	0.347	2.893	17.345
Change of stretch of defenders	-9.494	-1.635	-0.476	0.186	1.160	6.928
Change of min X-distance of defenders from own goal	-44.849	-6.321	-1.207	0.854	3.822	25.177

Chapter 6
Discussion

6.1 Summary of contributions and discussion

We introduce a conceptual framework for extracting knowledge from multivariate time series (MVTS) data through progressive temporal abstraction. This approach is structured into three sequential tasks that elevate the level of abstraction: 1) extracting univariate patterns of individual variables, 2) deriving higher-level patterns, and 3) exploring the distribution of these identified behavior patterns across the dataset.

These tasks are applied to two different types of MVTS data: continuous and discretized, the latter being derived from the former. For each task, we assess the suitability and utility of various techniques, providing examples of their application. We then introduce computational algorithms along with visualization techniques. Chapters 3 to 5 address tasks relevant to both forms of MVTS.

Table 6.1 provides a summary of the research findings derived from Chapter 3 to 5.

Research Question	**Chapter 3**	**Chapter 4**	**Chapter 5**	**Overall**
Identifying relevant intervals	Query-based segmentation	Pattern extraction without predefining pattern duration	Query-based segmentation	Segmentation methods with both predefined and flexible pattern extraction techniques
Extracting univariate patterns	Geometry-based shapes such as increasing and peak	Temporal trend such as increasing, constancy, or decreasing	SAX to extract state values	Trend-based patterns using geometric shapes and state-based patterns
Deriving higher-level patterns	Temporal co-occurrence	Temporal relations such as before, overlap, and after	Symbolic representation and topic modelling. Concurrency of univariate patterns	Various types of temporal relationship
Exploring the distribution of identified patterns	Frequency of pairwise co-occurrence and temporal distribution of univariate patterns	Frequency and temporal distribution of temporal relations	Spatio-temporal distributions of topic weights by the time-space matrix	Frequency, composition, and temporal distribution
Visualization	Co-occurrence network to represent frequency	Visual Information-Seeking Mantra and a small multiples display. Timeline views and circular representation	Visual Information-Seeking Mantra and a small multiples display. Mosaic matrices and pie charts	Visual Information-Seeking Mantra and small multiples displays. Network for pairwise relationships.

Table 6.1: Synthesis of answers to research questions from Chapter 3 to 5

Chapter 3: Identifying, exploring, and interpreting time series shapes in multivariate time series

Using the discretized form of MVTS, called episodes, we first present computational algorithms to identify predefined trend patterns, such as up-trend, peak, and constancy. From these identified patterns, we derive higher-level patterns, characterized by their concurrency. Finally, we investigate these patterns based on their occurrence frequency, temporal distribution, and frequent transitions.

Unless otherwise stated, section and figure numbers correspond to those in the related paper for this chapter.

Summary

This research as a whole outlines a general methodology for identifying high-level patterns, irrespective of the method used for basic pattern identification. It should be noted that defining a comprehensive pattern vocabulary is not a primary focus of this research.

In the part of the work presented in Chapter 3, we used geometry-based rules to identify five predefined types of patterns that are easily understandable: namely up-trend, down-trend, constancy, peak, and though. This approach allows for the detection of recognizable and interpretable shapes in time series and the assignment of understandable labels, thus facilitating further analysis of complex patterns. Temporal patterns that can be represented by a straight line, such as constancy, up-trend, and down-trend, were detected by simply considering the difference between the first and last values. Conversely, the remaining patterns, specifically peak and trough, required geometry-based thresholds such as triangle area and vertex order. The geometry-based algorithm also served to downsample time intervals (Chapter 3 Figure 9), aligning with the primary objective of the original study [39].

We treated pairwise co-occurrences of formerly identified patterns as high-level patterns. These co-occurrences facilitate searching for intervals with specific domain meanings, e.g., counter-attacks in football as described in Section 4.4 of the paper.

Various visualization techniques were used to display both predefined and high-level patterns, depending on the data's nature. Color-coded time series highlighted the identified patterns (shown in Figures 5 and 10). These two types of charts represented the temporal distribution of patterns for Google Mobility Data, which encompasses daily mobility data spanning three years. While timelines (Figures 1 and 12) are useful for identifying pattern co-occurrences, circular views (Figures 6 and 13) enable to compare the same periods of time, e.g., identifying annually recurring patterns. It is worth mentioning that the predominant pattern was constancy, colored in gray, which likely influenced the visual pattern identification.

Addressing the research questions

RQ1: Identifying relevant intervals: *How to find relevant intervals in univariate time series, such that the content of each interval can be considered holistically as an interpretable pattern?*

In this research, we use a query-based segmentation that allows analysts to select specific data events, which are distinct occurrences within a dataset that are either manually annotated or detected by prior computational processes, as discussed in Section 2.2. As a result, each interval encompasses at least one such data event, forming an interpretable pattern. It is worth noting that we assume relevant intervals are determined by domain experts or data analysts based on the inherent nature of the observed phenomenon. For instance, in the case of Google Mobility Data, a one-week interval was selected, reflecting the temporal cycle of human activities. For situations like football, intervals had variable duration, primarily determined by the ball possession by one team.

RQ2: Extracting univariate patterns of individual variables: *How to transform sequences of elementary values of individual variables into constructs that can be interpreted by humans as recognisable behavior patterns?*

The paper in Chapter 3 discussed transforming time series data into specific geometric shapes such as up-trend, peak, constancy, trough, and down-trend. An algorithm was proposed to automatically recognize these patterns. This approach directly answers the

research question by offering a method to transform raw data into human-understandable patterns.

RQ3: Deriving higher-level pattern types: *How to help analysts to 1) define higher level concepts as combinations of univariate pattern types linked by particular relationships, and 2) identify instances of these concepts (i.e., composite pattern types) in the data?*

In this work, we focused on one type of temporal relation between univariate patterns, to derive higher-level pattern, i.e., temporal co-occurrence. However, the introduced techniques introduced consider only pairwise co-occurrences, thus offering limited opportunities for exploring multivariate temporal patterns. There remains a challenge of analyzing and integrating these patterns into higher-level composite patterns. To address this gap, subsequent research, as presented in Chapter 5, utilized topic modeling.

RQ4: Exploring the distribution of identified behavior patterns over the dataset: *How to enable finding patterns in the distribution of earlier extracted patterns over the dataset dimensions?*

Visual analytics techniques were introduced to support the exploration of the temporal distribution of different types of patterns and relationships between these patterns. With the use of timeline views, circular representations, and graphical methods, the distribution and relationships of patterns over time were visualized. This allows for better understanding and analysis of the dynamic behavior of phenomena.

RQ5: Visualization: *How to represent computationally derived constructs to humans to enable pattern recognition and interpretation?* Multiple visualization methods, including timeline views, co-occurrence networks, and chord diagrams, were discussed. The visualizations aimed to represent the patterns and their relationships effectively to facilitate human understanding. The visual representation of computationally derived patterns offered users multiple perspectives on the data. However, as with all visual methods, the choice of representation might be subjective and dependent on user preferences and data properties.

Limitations

The selection of tasks T1-T4 was guided by the pattern theory, a theoretical model for pattern discovery as proposed by Andrienko et al. [6]. The underlying rationale was to create a structured approach wherein higher-level patterns are deduced from lower-level patterns. Task T1 was designed to extract these lower-level patterns, while tasks T2-T4 were geared towards identifying various forms of higher-level patterns shaped by their lower-level counterparts.

It's worth noting, however, that our framework doesn't encompass all potential tasks related to time series analysis. The inherent challenge lies in striking a balance between a structured, theory-driven approach and the vast array of possible tasks in time series analysis. Our method leans towards the former, which might not satisfy every conceivable analytical requirement.

Although the patterns used are easily recognizable, some, such as peaks and troughs, can be further decomposed into up-trends and down-trends. This issue was addressed by introducing a new segmentation method in Chapter 4.

We considered pairwise co-occurrences as high-level patterns, meaning only two co-occurring patterns were taken into account. Chapter 5 expanded this by considering concurrency among more than two patterns and their temporal distribution.

The chosen level of abstraction in this study has become a topic of debate. In our exploration, we adjusted values in a time series based on the initial value, resulting in ignoring the value magnitudes. This means changes in high and low states are treated equally. This methodological choice aligns with our aim to detect understandable shapes within time series data. However, it also presents an inherent limitation, as it might not capture the nuances and intricacies associated with magnitude-focused analyses, which other methods might offer.

Therefore, we introduced thresholds in our algorithm to refine the set of pattern types, offering a degree of flexibility in adjusting the level of abstraction. While this provides a potential pathway to satisfy more detailed analytical requirements, we should acknowledge that the high abstraction level might not be always appropriate for more granulated analyses.

Design recommendations

While our work predominantly centers on the development of a robust theoretical and methodological framework, we also recognize the benefits of practical software implementations. As such, future research might strive to strike a balance between these two aspects, ensuring both understanding and real-world applicability.

Regarding visual representations, it's essential to be aware of not only the differing interpretations of color across various application domains but also the accessibility concerns related to color perception. For instance, where red might symbolize loss in a financial context, it could be associated with positive values in another scenario. Additionally, given that many individuals perceive red-green colors differently, we may have to avoid this color combination, even though it was employed in our research.

While analysts can choose between timeline or circular views to understand concurrency or periodicity, a representation that encompasses both would be beneficial. We can use a matrix display as an alternative to the network diagram for representing pairwise co-occurrences. Additionally, the identification of pattern transitions that extend beyond two sequences could aid in detecting recurrent sequences.

Chapter 4: Exploring and visualizing temporal relations in multivariate time series

We first introduce computational algorithms to identify predefined types of patterns in the continuous form of MVTS, such as up-trend, down-trend, and constancy. For higher-level patterns, we compute temporal relations between these identified patterns. Lastly, we investigate relations between patterns from different attributes that frequently occur and demonstrate their temporal distributions.

Summary

In our examination of multivariate time series, our methodology detects patterns directly from continuous data without needing prior discretization. This approach not only avoids restricting patterns to a predefined duration but also does not operate under the assumption of synchronous behaviors among different attributes. Instead, we deeply explore a variety of possible temporal relations between patterns.

We have developed a progressive abstraction process. This process starts with identifying basic patterns in univariate time series. Subsequently, it provides analysts with

the flexibility to define high-level patterns through temporal relations between the basic patterns as sequences of elementary patterns. The ensuing analysis, which explores the distribution of these patterns, is applicable to both elementary and more complex pattern types.

Our geometry-based approach segments continuous time series to delineate predefined types of patterns like increasing, decreasing, and constancy. A distinction from Chapter 3 is our exclusion of peaks and troughs, treating them as composite of increasing and decreasing trends. Analysts can tailor the segmentation threshold based on their dataset characteristics.

Subsequent to the basic pattern identification, we derived high-level patterns by considering temporal relations between these patterns. The relationships are defined using simplified versions of Allen's temporal relations, considering the context of overlap between intervals.

Visualization was directed by the "Visual Information-Seeking Mantra" [37]. We adopted "small multiples" [42] for an overarching view and matrix views to denote the frequency of relations between patterns. Color schemes were selected with consideration for individuals with varied color perception. Finally, the temporal distribution of these relations was visualized using density charts, scales being standardized for easy comparison.

Addressing the research questions

RQ1: Identifying relevant intervals: *How to find relevant intervals in univariate time series, such that the content of each interval can be considered holistically as an interpretable pattern?*

The framework proposes an approach to pattern extraction without specifying the pattern duration in advance. We used the algorithms discussed in Chapter 3 to identify temporal patterns. This research excluded peak and trough patterns, opting instead to segment them into increasing and decreasing trends. Analysts have the flexibility to adjust thresholds to obtain intervals with their desired patterns.

RQ2: Extracting univariate patterns of individual variables: *How to transform sequences of elementary values of individual variables into constructs that can be interpreted by humans as recognisable behavior patterns?*

After segmentation, each interval encompasses one of the univariate patterns: increasing, constancy, or decreasing, which appear to be recognisable. Furthermore, the flexibility in interval length suggests that the derived patterns might be more intuitive and are likely not to constrained to inappropriate classification.

RQ3: Deriving higher-level pattern types: *How to help analysts to 1) define higher level concepts as combinations of univariate pattern types linked by particular relationships, and 2) identify instances of these concepts (i.e., composite pattern types) in the data?*

The framework can explore temporal relations between various pattern instances across different variables by using relations like "before", "overlap", and "after" from Allen's algebra of time intervals. However the specifics of overlaps, including their duration and the number of overlapping intervals, remain to be investigated.

RQ4: Exploring the distribution of identified behavior patterns over the dataset: *How to enable finding patterns in the distribution of earlier extracted patterns over the dataset dimensions?*

To determine the distribution of temporal relations, we calculated the frequency of pairwise relations. Additionally, density charts are employed, drawing on relative times between two intervals.

RQ5: Visualization: *How to represent computationally derived constructs to humans to enable pattern recognition and interpretation?*

Our approach uses Visual Information-Seeking Mantra [37] by providing analysts with an intial broad view of the computationally derived constructs. As analysts go deeper into specific regions of interest, they can seamlessly zoom and retrieve granular details. Furthermore, our visualization allows for the juxtaposition of similar data segments by incorporating the concept of small multiples [42]. This facilitates comparison of pattern recognition and makes contrasts more evident, improving the interactive process.

Limitations

A notable limitation in our approach was the limited scope for exploring the distribution of multivariate patterns, especially their temporal distribution. While the timeline view displays the univariate patterns, it does not show explicitly how these patterns are linked into multivariate patterns by particular temporal relations. One potential solution to this constraint could be the incorporation of query interface. Here, users can specify a combination of univariate patterns and relations between them. In return, the framework would highlight occurrences of these complex patterns along the timeline.

Another challenge arises from the dependency on threshold values, both for the segmentation of time series and the determination of temporal neighbors. Inappropriate thresholds could misrepresent the data, resulting in either overlooked patterns or excessive segmentation. Furthermore, the decision to simplify some of Allen's temporal relations, while practical, could lead to potential losses in specificity or data context relevance, as discussed by criticism in the introductory section [24].

The current evaluation, primarily anchored in usage scenarios, could benefit from practical assessments in real-world contexts. This approach would potentially yield a more nuanced understanding of the framework's practicality and effectiveness. While our dissertation emphasizes the conceptual framework, the software tools we developed are preliminary and serve primarily as a proof of concept. As such, they might not be ideally suited for exhaustive real-world evaluations. A holistic assessment would be better suited to tools specifically designed upon the principles of our proposed framework.

Design recommendations

For future implementations of this framework, an iterative pattern recognition process would be beneficial. Such a progress would allow for the continuous refinement of identified patterns. Specifically, this refinement involves defining new pattern types as combinations of earlier detected ones, linking them through specific relations. This could be done through immediate visual feedback enabled by an enhanced user interface, especially for locating complex pattern occurrences in data distribution.

Chapter 5: Episodes and topics in multivariate temporal data

In the earlier chapter, we highlighted a limitation: considering only pairwise relations between univariate patterns. In this study, we address this limitation by exploring the potential of topic modelling to detect co-occurrences involving more than two patterns.

Using the discretized version of MVTS, we first use computational algorithms to compress and encode the progression of values within each episode. These encoded values, which signify single-attribute patterns, are incorporated into topic modeling techniques. Finally, we investigate temporal variations in topic compositions, specifically examining how the distribution of topics changes over time.

Summary

We segment continuous time series into episodes using either a query-based method or a sliding time window. In the context of sliding windows, we obtain intervals by shifting a fixed-length window, allowing for overlap; these are called overlapping sliding windows. The use of overlapping sliding windows results in a higher number of time intervals than non-overlapping windows. This approach is adopted to reduce the risk of breaking essential patterns and potentially overlooking critical temporal behaviors.

For each univariate attribute within an episode, we represent value variations through a combination of symbols. Each symbol represents a state of values, encoded by Symbolic Aggregate approXimation (SAX) [22], and is considered a word. We propose a visual representation of SAX patterns, enhancing their interpretation by humans.

Each episode comprises words that represent variations in value states, and is treated as a text. Such a text encapsulates a high-level pattern consisting of multiple variables.

We use Natural Language Processing techniques to explore the co-occurrence of these symbolized sequences. To identify recurrent co-occurring episodes, we applied topic modelling techniques, such as Latent Dirichlet Allocation (LDA) [8] and Non-negative Matrix Factorization (NMF) [23]. These models seek to identify re-occurring combinations of words (i.e., SAX codes), which are considered as topics. While we used pie charts to examine the composition of topics (i.e., topic weights) within each time frame, it should be noted that pie charts may not always be suitable for visualising multivariate patterns, especially with linear spatial objects, as seen in our second case study. Instead, a small multiples display can better represent specific pattern distributions. This approach allows for easier comparison of topic distributions rather than precise measurement of individual topic weights. Additionally, merging similar topics can simplify the interpretation of color distributions.

Addressing the research questions

RQ1: Identifying relevant intervals: *How to find relevant intervals in univariate time series, such that the content of each interval can be considered holistically as an interpretable pattern?*

We used a top-down approach for segmentation, mirroring the method presented in Chapter 3. While segmenting based on temporal dimensions, as shown in the COVID-19 use case, may seem intuitive, the intervals may not always offer interpretable patterns. Nevertheless, our approach leans on subsequent processing via topic modelling to extract recurring and thereby significant patterns deserving interpretation, ignoring occasional ones.

RQ2: Extracting univariate patterns of individual variables: *How to transform sequences of elementary values of individual variables into constructs that can be interpreted by humans as recognisable behavior patterns?*

The Symbolic Aggregate Approximation (SAX) technique was used to transform sequences of elementary values into state values. By selecting an optimal number of states,

SAX simplifies data by reducing dimensions and filtering out noise, thus making pattern more interpretable.

RQ3: Deriving higher-level pattern types: *How to help analysts to 1) define higher level concepts as combinations of univariate pattern types linked by particular relationships, and 2) identify instances of these concepts (i.e., composite pattern types) in the data?*

In this work, the analyst defines higher level patterns by interpreting the topics extracted from the data. We captured simultaneous occurrences of various patterns across attributes. The behavior of each attribute within an episode is symbolized by a set of symbols, treated as a word. Consequently, the combined variations of multiple attributes are denoted by a combination of these words, interpreted as a text. Such a symbolic representation, combined with visual presentations like mosaic matrices, facilitates further exploration through topic modelling.

RQ4: Exploring the distribution of identified behavior patterns over the dataset: *How to enable finding patterns in the distribution of earlier extracted patterns over the dataset dimensions?*

Topic modelling facilitates the identification of homogeneous groups within the dataset. By representing behavioral patterns as topics, we were able to identify their distribution across the dataset dimensions. It is important to mention, however, that the high number of topics may pose scalability challenges.

RQ5: Visualization: *How to represent computationally derived constructs to humans to enable pattern recognition and interpretation?*

For pattern interpretation and the mental construction of higher level patterns, mosaic matrices are employed.

When it comes to the exploration of pattern distribution, the approach varies based on data characteristics. In a spatio-temporal view, pie charts represent the occurrences of topics. Alternatively, in the small multiples view, each component displays the distribution of an individual topic. The selection between these visual tools often depend on specific analytical requirements.

Limitations

The abstract nature of the workflow has been designed to encompass general approaches to data visualizations. However, potential scalability issues, highlighted at the end of subsection 5.3.3, indicate that as data dimensions increase, the current visualization techniques might face challenges in preserving clarity and interpretability.

The probabilistic nature of topic modeling methods, such as LDA, means that outcomes might vary across datasets or configurations. While topic modelling serves to abstract elementary values, its inherent probabilistic characteristics may produce inconsistent results, particularly when data volume is limited, as indicated in our second study. Furthermore, the decision between adopting LDA or NMF is not straightforward. The criteria for selecting one model over the other remains ambiguous and warrants further exploration to ensure consistent and reliable results in varied scenarios.

Lessons learnt

Our aim was to design a generalized approach for visual analytics. However, challenges arose in balancing between creating a conceptual framework and adjusting to domain-specific knowledge. As discussed in subsections 5.2 and 5.3, while a generalized framework

offers broad guidelines, data representation often requires adaptability to the unique attributes of datasets and analytical goals.

Our experiences in visualization design emphasized the significance of context. We realized that generic visualization approaches might not resonate across all scenarios, which highlights the necessity of tailoring to the specific needs and constraints of individual datasets.

Design recommendations

Users can gain more knowledge by adjusting the discretization of value intervals in SAX encoding. Additionally, using a diverging color scale enhances the interpretability of these SAX codes.

In cases with limited data, NMF often offers more consistent topic modelling outcomes than LDA. For episodes that can be represented by points in a display space, diagrams, such as pie charts, are advantageous because they provide a clear visualization of proportions. However, pie charts are typically not conducive to precise analysis of proportions. To alleviate this limitation, small multiples can be used, which support comparative analysis. Moreover, small multiples display various topic distributions across diverse data types.

6.2 Conclusions and future work

This dissertation introduces a general framework and methods for temporal abstraction in multivariate time series data. The focus of the research is on a concept-building approach to analysis of multivariate temporal data, which involved identifying basic patterns of individual attributes and analyzing the relations between the basic patterns to derive higher-level abstractions. Our framework involves methods to extract basic patterns of individual variables, define relevant intervals containing basic patterns, join basic univariate patterns into patterns of joint behavior, and identify patterns of occurrences of behavior patterns. Computational methods and visual techniques are suggested to aid domain experts in understanding and interpreting multivariate time series data, thereby extracting meaningful insights from the data. This research advances the field of temporal data analysis by providing a conceptual and methodological framework that combines computational and visual techniques to facilitate the abstraction of multivariate time series data progressively and aid domain specialists in comprehending them.

The approach to characterizing complex phenomena using progressive abstraction has proven useful, particularly in enhancing visualization methodologies and the theoretical foundations of interaction techniques. Progressive abstraction transforms elementary data points into univariate patterns, combines these patterns to form composites, and maps their distributions. The workflow of progressive abstraction (i.e., from elementary data points, via behavior patterns, to their distributions) reverses the traditional visualization mantra [37] of "overview first, zoom and filter, then details-on-demand". It underscores the importance of iterative data exploration, where each step is seamlessly connected through computational algorithms. This iterative process is important when adjusting pattern definitions and parameters for patterns. Interaction techniques can filter elementary data points to identify patterns. The filtered values can then highlight representative data or create new distributions, starting another iteration of the visualization mantra. By bridging different abstraction levels, these interaction techniques,

supported by computational algorithms, facilitate iterative analytical processes involving human analysts i.e., the human-in-the-loop approach.

Looking forward, the study opens up several promising directions for future research, beginning with the adaptation and application of our framework across different domains, as summarized in Table 6.2. This step is crucial for assessing the efficacy of our methods in diverse setting and identifying domain-specific modifications that may be required. As the framework is implemented in various contexts, there emerges a chance to refine and improve types of temporal patterns and temporal relations, making the analytical process more intuitive and efficient. Furthermore, the development of advanced search queries and visualization techniques, such as highlighting, will enable a more intuitive exploration of temporal patterns and relations. These enhancements aim to make the framework not only more robust but also more accessible for domain experts. An essential component of this future work would be establishing a feedback loop with experts. This would allow for the integration of domain specific knowledge, particularly when setting thresholds and defining patterns and relations relevant to different domains. Enabling the framework to incorporate domain knowledge would ensure that these methodologies remain relevant and effective across various applications.

	Computational	**Visual Analytics**
Focus	**Deriving artifacts from data**, such as patterns, relationships, structures, and the aggregation of characteristics.	**Supporting analytical workflows** that integrate human cognitive processes with computational artifact derivation.
Our work	**A conceptual framework** for progressive temporal abstraction and techniques required at each step in the framework.	**Visual techniques** for progressive abstraction and interpretation of multivariate time series data.
Future	**Applying the framework across various domains** to refine types of temporal patterns and relations. Enhancing the analytical process for domain-specific needs or hypothesis.	**Developing advanced search queries and visualization techniques** for intuitive exploration. Establishing a feedback loop with experts to integrate domain-specific knowledge and refine the framework.

Table 6.2: Our path in MVTS research: summary of focus, contributions, and future directions in computational and visual analytics methods

In conclusion, this proof of concept study in progressive temporal abstraction will allow to refine our framework further and thereby establishing a stronger presence in the field of temporal data analysis.

Bibliography

[1] R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. In *Foundations of Data Organization and Algorithms*, pages 69–84. Springer Berlin Heidelberg, 1993.

[2] J. F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, Nov. 1983.

[3] G. Andrienko, N. Andrienko, P. Bak, D. Keim, and S. Wrobel. *Visual Analytics of Movement*. Springer Berlin Heidelberg, 2013.

[4] N. Andrienko, G. Andrienko, E. Camossi, C. Claramunt, J. M. Cordero Garcia, G. Fuchs, M. Hadzagic, A. L. Jousselme, C. Ray, D. Scarlatti, and G. Vouros. Visual exploration of movement and event data with interactive time masks. *Visual Informatics*, 1(1):25–39, 2017.

[5] N. Andrienko, G. Andrienko, G. Fuchs, A. Slingsby, C. Turkay, and S. Wrobel. *Visual Analytics for Data Scientists*. Springer Nature, 2020.

[6] N. Andrienko, G. Andrienko, S. Miksch, H. Schumann, and S. Wrobel. A theoretical model for pattern discovery in visual analytics. *Visual informatics*, 5(1):23–42, Mar. 2021.

[7] N. Andrienko, G. Andrienko, and G. Shirato. Episodes and topics in multivariate temporal data. *Computer graphics forum: journal of the European Association for Computer Graphics*, 42(6), Sept. 2023.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 2003.

[9] D. Brughardt, A. Dunkel, E. Hautahl, G. Shirato, N. Andrienko, G. Andrienko, M. Hartmann, and R. Purves. *Extraction and visually driven analysis of VGI for understanding people's behavior in relation to multi-faceted context*, pages 213–234. Springer, 2023.

[10] K.-P. Chan and A. W.-C. Fu. Efficient time series matching by wavelets. In *Proceedings 15th International Conference on Data Engineering (Cat. No.99CB36337)*, pages 126–133, Mar. 1999.

[11] G. Das, K.-I. Lin, H. Mannila, G. Renganathan, and P. Smyth. Rule discovery from time series. *KDD*, 1998.

[12] L. Etienne, T. Devogele, M. Buchin, and G. McArdle. Trajectory box plot: a new pattern to summarize movements. *International journal of geographical information science: IJGIS*, 30(5):835–853, May 2016.

[13] T.-C. Fu. A review on time series data mining. *Engineering applications of artificial intelligence*, 24(1):164–181, Feb. 2011.

[14] S. K. Gadia. A homogeneous relational model and query languages for temporal databases. *ACM Trans. Database Syst.*, 13(4):418–448, Oct. 1988.

[15] S. Gao. Spatio-Temporal analytics for exploring human mobility patterns and urban dynamics in the mobile age. *Spatial cognition and computation*, 15(2):86–114, Apr. 2015.

[16] D. Herr, F. Beck, and T. Ertl. Visual analytics for decomposing temporal event series of production lines. In *2018 22nd International Conference Information Visualisation (IV)*. IEEE, July 2018.

[17] S. Hirano and S. Tsumoto. Mining frequent temporal patterns from medical data based on fuzzy ranged relations. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2654–2658, Dec. 2019.

[18] I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 374(2065):20150202, Apr. 2016.

[19] E. Keogh, S. Chu, D. Hart, and M. Pazzani. Segmenting time series: a survey and novel approach. In *Data Mining in Time Series Databases*, volume 57 of *Series in Machine Perception and Artificial Intelligence*, pages 1–21. WORLD SCIENTIFIC, June 2004.

[20] M. Last, Y. Klein, and A. Kandel. Knowledge discovery in time series databases. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics: a publication of the IEEE Systems, Man, and Cybernetics Society*, 31(1):160–169, 2001.

[21] T.-Y. Lee and H.-W. Shen. Visualization and exploration of temporal trend relationships in multivariate time-varying data. *IEEE transactions on visualization and computer graphics*, 15(6):1359–1366, 2009.

[22] J. Lin, E. Keogh, L. Wei, and S. Lonardi. Experiencing SAX: a novel symbolic representation of time series. *Data Min. Knowl. Discov.*, 15(2):107–144, Oct. 2007.

[23] M. Luo, F. Nie, X. Chang, Y. Yang, A. Hauptmann, and Q. Zheng. Probabilistic Non-Negative matrix factorization and its robust extensions for topic modeling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), Feb. 2017.

[24] F. Mörchen. A better tool than allen's relations for expressing temporal knowledge in interval data. In *The Twelveth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

[25] R. Moskovitch and Y. Shahar. Fast time intervals mining using the transitivity of temporal relations. *Knowl. Inf. Syst.*, 42(1):21–48, 2013.

[26] R. Moskovitch and Y. Shahar. Classification of multivariate time series via temporal abstraction and time intervals mining. *Knowl. Inf. Syst.*, Sept. 2014.

[27] T. Oates. Identifying distinctive subsequences in multivariate time series by clustering. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '99, pages 322–326, New York, NY, USA, Aug. 1999. Association for Computing Machinery.

[28] Y. Qiang, M. Delafontaine, M. Versichele, P. De Maeyer, and N. Van de Weghe. Interactive analysis of time intervals in a two-dimensional space. *Information visualization*, 11(4):255–272, Oct. 2012.

[29] G. Ruan, H. Zhang, and B. Plale. Parallel and quantitative sequential pattern mining for large-scale interval-based temporal data. In *2014 IEEE International Conference on Big Data (Big Data)*, pages 32–39, Oct. 2014.

[30] L. Sacchi, C. Larizza, C. Combi, and R. Bellazzi. Data mining with temporal abstractions: learning rules from time series. *Data Min. Knowl. Discov.*, 15(2):217–247, Oct. 2007.

[31] Services, Wiley Author. CRediT. `https://authorservices.wiley.com/author-resources/Journal-Authors/open-access/credit.html`. Accessed: 2024-6-2.

[32] Y. Shahar. A framework for knowledge-based temporal abstraction. *Artif. Intell.*, 90(1):79–133, Feb. 1997.

[33] L. Shao, D. Sacha, B. Neldner, M. Stein, and T. Schreck. Visual-interactive search for soccer trajectories to identify interesting game situations. *IS and T International Symposium on Electronic Imaging Science and Technology*, 0(February), 2016.

[34] G. Shirato, N. Andrienko, and G. Andrienko. What are the topics in football? extracting time-series topics from game episodes. In *2021 IEEE Visualization Conference*, 2021.

[35] G. Shirato, N. Andrienko, and G. Andrienko. Exploring and visualizing temporal relations in multivariate time series. *Visual Informatics*, 7(4):57–72, Dec. 2023.

[36] G. Shirato, N. Andrienko, and G. Andrienko. Identifying, exploring, and interpreting time series shapes in multivariate time intervals. *Visual informatics*, 7(1):77–91, Mar. 2023.

[37] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Computer Society Press, 1996.

[38] J. Sklansky and V. Gonzalez. Fast polygonal approximation of digitized curves. *Pattern recognition*, 12(5):327–331, Jan. 1980.

[39] S. Steinarsson. *Downsampling Time Series for Visual Representation*. PhD thesis, University of Iceland, 2013.

[40] M. Su, W. Zhao, Y. Zhu, D. Zha, Y. Zhang, and P. Xu. Anomaly detection of vectorized time series on aircraft battery data. *Expert systems with applications*, 227:120219, Oct. 2023.

[41] K. Takabayashi, T. B. Ho, H. Yokoi, T. D. Nguyen, S. Kawasaki, S. Q. Le, T. Suzuki, and O. Yokosuka. Temporal abstraction and data mining with visualization of laboratory data. *Studies in health technology and informatics*, 129(Pt 2):1304–1308, 2007.

[42] E. Tufte. Envisioning information. *Bulletin of science, technology & society*, 13(1):38–38, 1990.

[43] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. `https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf`, 2008. Accessed: 2023-3-6.

[44] J. J. Van Wijk and E. R. Van Selow. Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis'99)*, pages 4–9, Oct. 1999.

[45] E. Vieth. Fitting piecewise linear regression functions to biological responses. *Journal of applied physiology*, 67(1):390–396, July 1989.

[46] R. Villafane, K. A. Hua, D. Tran, and B. Maulik. Knowledge discovery from series of interval events. *Journal of intelligent information systems*, 15(1):71–89, July 2000.

[47] H. Wang, H. Huang, X. Ni, and W. Zeng. Revealing Spatial-Temporal characteristics and patterns of urban travel: A Large-Scale analysis and visualization study with taxi GPS data. *ISPRS International Journal of Geo-Information*, 8(6):257, May 2019.

[48] K. Wongsuphasawat and D. Gotz. Outflow : Visualizing patient flow by symptoms and outcome. *IEEE VisWeek Workshop on Visual Analytics in Healthcare*, pages 25–28, 2011.

[49] L. Ye and E. Keogh. Time series shapelets: a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 947–956, New York, NY, USA, June 2009. Association for Computing Machinery.