Tampere University

CHIEN LU

Shallow Representations, Profound Discoveries

A methodological study of game culture in social media

Tampere University Dissertations 851

Tampere University Dissertations 851

CHIEN LU

Shallow Representations, Profound Discoveries

A methodological study of game culture in social media

ACADEMIC DISSERTATION
To be presented, with the permission of
the Faculty of Information Technology and Communication Sciences
of Tampere University,
for public discussion in the lecture room K103
of the Linna Building, Kalevantie 5, Tampere,
on 6th of September 2023, at 13 o'clock.

ACADEMIC DISSERTATION
Tampere University, Faculty of Information Technology and Communication Sciences
Finland

Responsible supervisor and Custos
Professor
Jaakko Peltonen
Tampere University
Finland

Supervisor
Senior Research Fellow
Timo Nummenmaa
Tampere University
Finland

Pre-examiners
Associate Professor
Yun-Gyung Cheong
Sungkyunkwan University
South Korea

Assistant Professor
Debora Nozza
Bocconi University
Italy

Opponent
Associate Professor
Arto Klami
University of Helsinki
Finland

Cover design: Roihu Inc.

ISBN 978-952-03-3030-9 (print)
ISBN 978-952-03-3031-6 (pdf)
ISSN 2489-9860 (print)
ISSN 2490-0028 (pdf)
http://urn.fi/URN:ISBN:978-952-03-3031-6

Carbon dioxide emissions from printing Tampere University dissertations have been compensated.

PunaMusta Oy – Yliopistopaino
Joensuu 2023

ABSTRACT

This thesis explores the potential of representation learning techniques in game studies, highlighting their effectiveness and addressing challenges in data analysis. The primary focus of this thesis is shallow representation learning, which utilizes simpler model architectures but is able to yield effective modeling results. This thesis investigates the following research objectives: disentangling the dependencies of data, modeling temporal dynamics, learning multiple representations, and learning from heterogeneous data. The contributions of this thesis are made from two perspectives: empirical analysis and methodology development, to address these objectives. Chapters 1 and 2 provide a thorough introduction, motivation, and necessary background information for the thesis, framing the research and setting the stage for subsequent publications. Chapters 3 to 5 summarize the contribution of the 6 publications, each of which contributes to demonstrating the effectiveness of representation learning techniques in addressing various analytical challenges.

In Chapter 1 and 2, the research objects and questions are also motivated and described. In particular, Introduction to the primary application field game studies is provided and the connections of data analysis and game culture is highlighted. Basic notion of representation learning, and canonical techniques such as probabilistic principal component analysis, topic modeling, and embedding models are described. Analytical challenges and data types are also described to motivate the research of this thesis.

Chapter 3 presents two empirical analyses conducted in Publication I and II that present empirical data analysis on player typologies and temporal dynamics of player perceptions. The first empirical analysis takes the advantage of a factor model to offer a flexible player typology analysis. Results and analytical framework are particularly useful for personalized gamification. The Second empirical analysis uses topic modeling to analyze the temporal dynamic of player perceptions of the game *No Man's Sky* in relation to game changes. The results reflect a variety of player perceptions

including general gaming activities, game mechanic. Moreover, a set of underlying topics that are directly related to game updates and changes are extracted and the temporal dynamics of them have reflected that players responds differently to different updates and changes.

Chapter 4 presents two method developments that are related to factor models. The first method, DNBGFA, developed in Publication III, is a matrix factorization model for modeling the temporal dynamics of non-negative matrices from multiple sources. The second mothod, CFTM, developed in Publication IV introduces a factor model to a topic model to handle sophisticated document-level covariates The develeopd methods in Chapter 4 are also demonstrated for analyzing text data. Chapter 5 summarizes Publication V and Publication VI that develop embedding models. Publication V introduces Bayesian non-parametric to a graph embedding model to learn multiple representations for nodes. Publication VI utilizes a Gaussian copula model to deal with heterogeneous data in representation learning. The develeopd methods in Chapter 5 are also demonstrated for data analysis tasks in the context of online communities.

Lastly, Chapter 6 renders discussions and conclusions. Contributions of this thesis are highlighted, limitations, ongoing challenges, and potential future research directions are discussed.

CONTENTS

List of Figures

List of Tables

ABBREVIATIONS

CFTM	Cross-factor topic model
DNBGFA	Probabilistic dynamic non-negative Bayesian group factor
EFA	Exploratory factor analysis
EFE	Exponential family embedding
ELBO	Evidence lower bound
GCE	Gaussian copula embeddings
GFA	Bayesian group factor analysis
GPLVM	Gaussian process latent variable mode
HCI	Human computer interaction
MUDs	Multi-user dungeons
NLP	Natural language processing
NMF	Non-negative matrix factorization
PA	parallel analysis
PPCA	Probabilistic principal component analysis
RBF	Radial basis function
STM	Structrual topic model
TF-IDF	Term frequency-inverse document frequency
VAE	Variational autoencoder

ORIGINAL PUBLICATIONS

Publication I — Xiaozhou Li, Chien Lu, Jaakko Peltonen, and Zheying Zhang. "A statistical analysis of Steam user profiles towards personalized gamification". In: *Proceedings of the 3rd International GamiFIN Conference, Levi, Finland, April 8-10, 2019*. Ed. by Jonna Koivisto and Juho Hamari. CEUR-WS.org, 2019, pp. 217–228.

Publication II — Chien Lu, Xiaozhou Li, Timo Nummenmaa, Zheying Zhang, and Jaakko Peltonen. "Patches and Player Community Perceptions: Analysis of No Man's Sky Steam Reviews". In: *DiGRA '20 - Proceedings of the 2020 DiGRA International Conference*. Ed. by Dale Leorke. 2020.

Publication III — Chien Lu, Jaakko Peltonen, Jyrki Nummenmaa, and Kalervo Järvelin. "Probabilistic Dynamic Non-negative Group Factor Model for Multi-source Text Mining". In: *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*. Ed. by Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux. ACM, 2020, pp. 1035–1043.

Publication IV — Chien Lu, Jaakko Peltonen, Timo Nummenmaa, Jyrki Nummenmaa, and Kalervo Järvelin. "Cross-structural Factor-topic Model: Document Analysis with Sophisticated Covariates". In: *Asian Conference on Machine Learning, ACML 2021, 17-19 November 2021, Virtual Event*. Ed. by Vineeth N. Balasubramanian and Ivor W. Tsang. PMLR, 2021, pp. 1129–1144.

Publication V Chien Lu, Jaakko Peltonen, Timo Nummenmaa, and Jyrki Nummenmaa. "Nonparametric Exponential Family Graph Embeddings for Multiple Representation Learning". In: *Uncertainty in Artificial Intelligence, 1-5 August 2022, Eindhoven, The Netherlands*. Ed. by James Cussens and Kun Zhang. PMLR, 2022, pp. 1275–1285.

Publication VI Chien Lu and Jaakko Peltonen. "Gaussian Copula Embeddings". In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022.

Author's contribution

For Publications I and II, the author had a lead responsibility for statistical data analysis and model training, where Xiaozhou Li was in charge of data collection, curation, and model interpretation. The author also assisted in result interpretation and content writing in Publication I. For Publication II, the author took a notable role in coordinating the result interpretation and content writing. For Publications III - VI, the author was the primary contributor to the process of conceptualization, implementation, evaluation, and paper writing. Of the remaining authors, Jaakko Peltonen assisted in result interpretation and writing of publications I-VI, and conceptualization and experiment design in Publications III-VI. Timo Nummenmaa assisted in result interpretation and writing of Publications II, IV, and V. Jyrki Nummenmaa and Kalervo Järvelin advised and commented on Publications III-V. Zheying Zhang assisted in result interpretation and writing of Publications I-II.

1 INTRODUCTION

This chapter offers an general outline of this thesis. It introduces the significance of the primary application domain, the explored branch of methods, and the specific type of data under investigation. The research objective and the structure of this thesis are also presented.

1.1 Games, Play, and Data Analysis

> "*Play is older than culture, for culture, however inadequately defined, always presupposes human society, and animals have not waited for man to teach them their playing.*" [71, p. 1]

As stated in Johan Huizinga's seminal work "Homo Ludens", games and play are, inarguably, not just leisure activities, but a fundamental aspect of human culture. In other words, games and play not only offer entertainment but have also participated in the process of the formation of culture [85]. Games and play also create communities and a sense of belonging [162], shaping societal norms and expectations [57]. Understanding such cultural implications of games and play is a vital task for modern society.

To understand games and play, data analysis offers its potentials as a tool for exploring various underlying patterns, especially in a complex setting. The vast amounts of data generated by non-digital and digital gaming and online communities related to games and play provide an access to understanding players' experiences and behaviors. Online data sources, especially those from social media can supplement traditional data collection methods in game studies. Conventional methods like surveys and interviews can be costly and often result in smaller and curated datasets. Such conventional data collection processes requires significant investment in time and effort, from designing questionnaires to carefully executing the process. By taking advantage of such data, patterns and trends can be identified and that can be used

to reveal how players engage with games and how games have made an impact on players. This thesis is devoted to exploring such potential of data analysis with a focus on representation learning to provide insights through analyzing the data from the complex and diverse world of game culture.

1.2 Shallow, Simple yet Effective

This thesis delves into the area of data analysisknown as "*shallow representation learning*". The term "*shallow*" is here used to describe computational models with simpler structures in contrast to "*deep*" learning models which have multiple layers and complex structures [107]. The umbrella term, representation learning, is a set techniques that aims to uncover latent, unobserved, and abstract features from data [10]. It has gained popularity in the field of machine learning, particularly with the rise of deep learning models. On the other hand, research on shallow representation learning has received less attention compared to deep learning and requires necessary exploration.

This thesis focuses on shallow representation learning, which can offer several advantages despite the simpler structure. For example, by incorporating human insights about the analyzed data into the model design, shallowly learned representations can yield competitive results without the need for deep or complex architectures. Additionally, these representations can be more transferable, less task-dependent, and more interpretable. Moreover, shallow models can offer a more data-efficient solution and are potentially more suitable for low-resource settings. This thesis develops various models with simple structures to demonstrate these benefits. The developed models are inspired by the needs of empirical analysis tasks and carried out by solving challenges by introducing various modeling insights, e.g., multiple representations, and heterogeneous data settings. The methods are also proven useful in various data analysis and prediction tasks.

The focal application area of these representation learning methods is games, specifically player-generated data on social media related to games and play. Such data is often complex, large in volume, and contains noise. Representation learning can be used to distill essential and valuable information in such settings. The learned representation can be used to summarize the overall theme of the collected data or describe the dependencies between different data variables.

1.3 Player-generated Data on Social Media

Playing has been a universal human activity throughout history. Moreover, as one of the mainstream forms of entertainment, players' engagement is usually not limited to game-play itself but also diffuses to a broader range of activities that are outside of game-play.

The emergence of internet and social media have facilitated such activities, especially in the formation of online communities [49]. On different online platforms, players interact with other users. Such activities usually leave behind a variety of digital footprints, therefore, social media has become a unique venue for the convergence of various types of player-generated data. These data are often noisy and unstructured, yet they contain a wealth of valuable information for researchers looking to understand players and games.

This thesis aims to explore the potential of using representation learning as a tool for analyzing player-generated data on social media. Representation learning can be used to extract simplified abstractions while preserving important patterns, variations and inter-relationships within the data. For example, without reading through the collected game reviews using human efforts, representation learning techniques such as topic modeling can be used to inspect the themes of the discussions and at the same time, model the features such as inter-correlations and temporal dynamics of those extracted themes.

1.4 Research Objectives, Questions and Contributions

In summary, the research objectives of this thesis lie at the intersection of representation learning and game-related data from social media. This thesis aims to explore the potential of using representation learning as a tool for analyzing data to better understand players and their gaming experiences. The developed methods can be applied to a wide range of data, including player profiles, game reviews, streamer networks, and esports match performances. The objectives can be summarized as follows:

- **Disentangle the Dependencies of Data:** This objective focuses on distilling the complexity of the observed data, especially in the situation where the observed variables are intertwining with each other. For example, the player pro-

files often contain related variables (e.g., achievements and completions) and they show strong inter-dependency. Leveraging the prior knowledge that the observed dependency of variables is governed by a specific latent mechanism can help deduct information that is redundant and extract essential underlying characteristics to describe the data. For example, the activities such as getting achievements and completions are an outcome of a specific player personality that can not be directly observed.

- **Modeling Temporal Dynamics**: This objective focuses on the underlying changes over time. For example, the players' perceptions can alter over time due to many reasons such as the updates or new versions of games.
- **Cross-structure Learning**: It is common that the analyzed data set contains multimodality, as data can be collected in different formats. For example, the text review written by a player is linked to the player's profile which contains mainly numerical values.
- **Multiple Representation Learning**: Just like a word can express different meanings in different contexts, a player can behave differently or show a different personality in different surroundings. Games and play can be diverse and vibrant, for example, each player can carry different traits and personalities and those differences can further affect the player's behavior. The multiple representations learned should be able to uncover such diversity. On the other hand, the learned representation is expected to enhance the performance in various prediction tasks compared to single representation solutions for they can capture such nuance. However, allowing such freedom can lead to risks of introducing noise.
- **Learning from Heterogeneous Data**: A common situation of social media data is that the data are often mixed. For example, each player can be gauged by different performance measurements (win/loss, kills, deaths, and so on) and each is in a different data type. This yields challenges when trying to learn representations for each player. The learned representations are required to distill the necessary information of the player, and at the same time, the interrelations between different measurements.

To address the objectives, this thesis focuses on the following research questions:

- *RQ.1*. How to distill the crucial information from the dependencies and uncover the perpendicular, or uncorrelated dimensions that reveal the underlying structure of the data?
- *RQ.2*. How to leverage representation learning techniques to model and understand such evolution of data over time?
- *RQ.3*. How do different underlying structures interact with each other? How to model and interpret such interactions?
- *RQ.4*. What is the appropriate approach to introduce diversity to representation learning? How to take advantage of the learned representations?
- *RQ.5*. How to effectively analyze and integrate heterogeneous data from various sources to derive meaningful insights? How to interpret and visualize the modeling results?

The thesis employs two perspectives: empirical analysis and methodology development, to address the research objectives and questions. The empirical analysis in Chapter 3 demonstrates how existing representation learning techniques, specifically, factor models and topic models, can be used data analysis when it comes to understanding games and play. It covers game studies issues including player typologies and player perceptions and lays the foundation for the methodologies by illustrating the promising potential of representation learning.

Chapter 4 therefore develops methodologies related to factor models and topic models. A probabilistic dynamic non-negative Bayesian group factor (DNBGFA) model is proposed to analyze text data collected over time. Additionally, the Cross-factor Topic Model (CFTM) is further developed to leverage a factor structure and handle sophisticated covariates when analyzing documents with document-level high-dimensional data.

Chapter 5 delves into the application of another facet of representation learning: embedding models. In contrast to factor and topic models, where learned representations are explicitly align with features (column names) in the training data, embedding models exhibit an implicit nature, disassociated from predefined features. This implicit characteristic offers flexibility but may bring challenges for interpretation. The developed embedding models in Chapter 5 try to overcome such challenges while harnessing the flexibility. The non-parametric graph embedding model developed in the chapter introduces the concept of multiple representations, bolstering

learning flexibility, and the number of learned representations of each node can be further interpreted based on the activities and connections with other nodes. The Gaussian copula embedding model addresses heterogenous data scenarios, wherein the embedding vectors learn associations with specific variables, taking into account their heterogeneity and dependencies, and offer enhanced potential for interpretation.

The contribution of this thesis is contained in 6 publications. Publications I and II are empirical studies that apply representation learning to game data analysis tasks. The investigated issues include discovering underlying player factors and the evolution of game reviews over time. The conducted research has shown the potential of applying such methods in game studies. Inspired by the empirical works, a series of representation learning models have been developed. The scope is further narrowed down to two focused branches of representation learning methods: the factor models and the embedding models. Under each branch, two novel methods have been developed.

- **Factor models**. Two novel methods have been developed. The first developed method (Publication III) corresponds to the first listed objective **Modeling Temporal Dynamics**. A non-negative matrix factorization model is developed to model the underlying temporal dynamics of multiple text sources. The main idea is to introduce a modified Gaussian process latent variable to a non-negative matrix decomposition problem.

  The second developed method (Publication IV, inspired by Publication I) tackles the challenge **Cross-structure Learning**. The developed method uses a factor model to extract the underlying structure from the numerical covariates and integrate the factors into a topic model.

- **Embedding models**. This branch develops two novel methods. The first method (Publication V) focuses on unsupervised graph embedding, with respect to **Multiple Representation Learning**. Bayesian non-parametric methods are employed to derive a multiple representation learning framework.

  The second work (Publication VI) focuses on a general embedding vector learning problem. The method focuses on the challenge of **Learning from Heterogeneous Data** and proposes an embedding framework based on a Gaussian Copula model. The developed method has been shown effective in mod-

eling various data sets about games and play.

1.5 Thesis Structure

This thesis is organized as follows. Chapter 2 provides the fundamental notions for this thesis. A substantial overview of representation learning is concentrating on branches of methods focused on this thesis. It further provides an introduction to game-related social media data. The characteristics, challenges of such data, and potential values can be yielded by harnessing such data as analytical resources are described. This chapter also attempts to draw upon the literature in game culture studies to better position this thesis with a facilitated theoretical background.

Chapter 3 describes the empirical works that apply representation learning in game studies. Chapter 4 introduces the developed method related to **Factor models** and in Chapter 5 the representation learning methods related to **Embedding models** are described. Note that, Chapters 4 and 5 will focus on the overview of models and the key notions supporting such methodology development. The details of the algorithm and experiments that can be found in the publications and will not be repeated. Finally, Chapter 6 concludes the thesis, discusses the limitations, and outlines future opportunities.

2 FUNDAMENTALS

This chapter lays the foundations of the background knowledge for this thesis. An overall introduction of the application area, game studies, is first provided in section 2.1. Section 2.2 describes the basic notions of representation learning, and the reasons for choosing a "shallow" perspective. Section 2.3 covers the characteristics of social data for game studies, the challenges of analyzing such data. The methodologies developed in thesis are devoted to delivering solutions to those challenges.

2.1 Game Studies, Data Analysis, and Social Media

This section examines the specific area of focus for this thesis, beginning with an introduction to research on games with a focus on the field of game studies, as the most established discipline when it comes to games and play. Key issues that are concerned in this thesis will be highlighted. Another attempt of this section is to delve deeper into the theoretical foundations and conceptual resources especially the intersection of data analysis and game studies. The data-intensive nature of game-play are highlighted, emphasizing the potential of data analysis to enhance our understanding of games. The focus is further narrowed down to social media data, discussing the significance of the relationship between social media and games. Finally, the analytical challenges are outlined to further motivate this thesis.

2.1.1 Research Games and Game Studies - An Overview

As mentioned, games have been a substantial cultural form in modern human society. As a results, it has also attracted scholarly attention and academic investigation. For example, in the field human-computer interaction (HCI), games are intensively studied as a "computer-technology system" and players are seen as the "users" of the system [7, 19]. For educational and marketing purposes, games and "playful elements" are used in "non-game purposes" to facilitate the business development or

learning process [73, 83].

In psychology, games have also been investigated as part of internet culture under the context of addiction or abuse of substances [92, 60].

As games have become a primary cultural form [22], increasing scholarly attention has been attracted and devoted to understanding the cultural implications of games and play. Game studies or game culture studies that "emerged in the late 1990s and early 2000s" [113], is a focused field on games culture and an interdisciplinary field dedicated to understanding the unique properties and potential of games as a medium. This field emphasizes the importance of studying games on their own terms, and encourages the development of specific methods and frameworks for studying games.

Due to the dynamic and rapidly developing nature of the field, it can be challenging to find a clear and static definition for game studies. As the field continues to evolve, new encountered research areas and perspectives are constantly emerging into play, making it difficult to encapsulate all aspects of the field in a single definition. One possible definition of game studies can be

> "*study and learning with games and related phenomena as its subject matter*" [109, p. 6].

Studying game cultures involves examining the game, the players, and the contexts in which the game is played in order to understand the meaning and cultural expression of the game. The term "culture" is defined as a "system of meaning" [109]. One important characteristic that differentiates game studies from other fields is that, despite that games have been intensively researched and studied in various fields and academic settings, the mentioned approaches of how games are studied have revealed that the emphasis and visibility of games and their roles in cultural formation are usually not the primary focus. Game studies approaches games as its primary subject matter, and follows the "games in culture" paradigm [109], tries to understand games "in its own right" [22]. In contrast, other fields may study games, in general, in relation to other research objectives or as a particular form or case when being investigated. The importance of taking games as a primary focus as underscored in the prominent journal Game Studies: the International Journal of Computer Game Research, which states that the research articles in the venue should:

> "*attempt to shed new light on games, rather than simply use games as metaphor*

or illustration of some other theory or phenomenon"

The cultural significance of games and the need to understand their unique properties and potential as a medium have been highlighted in pioneer works. It has been stated that:

> "*[games are] an extremely valuable context for the study of cognition as inter(action) in the social and material world. They provide a representational trace of both individual and collective activity and how it changes over time, enabling the researcher to unpack the bidirectional influence of self and society* " [157, p. 1].

The quote further highlights the importance of studying games in order to understand how they shape and are shaped by individuals and society. It investigates games to understand their meanings, perspectives, and contexts.

As a young field, game studies is a rapidly growing field with many opportunities for research and exploration in the continuously evolving world of games. It is worth noting that, although the emphasis is unique, it is still closely related to other research fields as games have been a common studied object in various academic efforts.

This approach highlights the unique properties and potential of games as a medium and encourages the development of specific methods and frameworks for studying games. It emphasizes that game studies as a field of study should focus on understanding games and related phenomena, as opposed to using games as a way to understand other subjects.

Perhaps one noteworthy development phase of the "independence" of game studies was the "Ludology vs. narratology debates". During the debate, the narratologists have argued that games can be interpreted as a form of text, similar to books, films and other media, methods from literacy studies can be accordingly applied to analyze games [1]. To the contrary, the school of ludology, represented by [76], asserted that, instead of being considered as a form of text, a game should be seen as a set of rules and norms and they alone express meanings; furthermore, even if games can comprise storytelling, there are significant differences between games and other media such as the linearity of time [77]. The concern of the "colonisation from other fields" was also raised [44]. Although this debate has not reached a final consensus and even has been later described as a result of "a series of misunderstandings and

misconceptions" [50], it nevertheless exemplifies how game studies has struggled to distinguish itself as an independent academic discipline.

On the other side of the above-mentioned struggles, game studies, as a dynamic and ever-evolving field, also highly stresses the values of interdisciplinarity, inclusivity, and conversation to fields [37]. It has benefited from various theories, research methods, practices, and perspectives brought from various disciplines, as stated:

> "*The scientific and scholarly study of games, play, and related phenomena must be able to address the complex and multidimensional character of games, for which familiarity with multiple fields of inquiry is a clear benefit.*" [111].

The growing amount of recognition and exchange of ideas between different fields of study and research methods are likely to lead to a disruptive shift that will close some of the divide that has been observed between the social sciences, humanities, and the study of technology in relation to games [176]. Researchers in game studies carry a diversity of backgrounds [112].

This characteristic has resulted in that games studies often contains a combination of various approaches and does not often embrace a clear or exclusive set of methodologies [93] so that, techniques are often "borrowed" from other disciplinaries to study games [152]. For example, the notion of "user experience" from HCI can be seen as the precursor of "player experience" when researching games and players [175]; the notion of "player typologies" is closely related to the concept of segmentation marketing theory and literature [64] as a process to differentiate and identify customer groups [90]. Methods such as psychophysiological measurements [86] and modern data analytical techniques such as using natural language processing (NLP) to, for example, analyze game reviews, have also been highlighted [188, 111].

Games and game culture are constantly evolving with the development of technology. The emergence and rise of social media, in particular, has had a significant impact on the field of game studies. It has created a space for game-related cultural activities to take place on platforms such as Reddit [108] and Twitch [54], providing opportunities for researchers to study and understand the complexities of games and play within this context. However, it also poses methodological challenges for researchers as they navigate and utilize the vast amount of complex data available on these platforms.

As the primary application field of this thesis, game studies encompasses various disciplines but emphasizes the uniqueness of games, and examines and analyzes games

as a primary focus. On the other hand, it is an interdisciplinary and dynamic field that shares research methods with various related fields.

2.1.2 Data-intensive Game Culture

The connection between data analysis and game studies can be viewed as the intersection of two fields. Digital games generate vast amounts of data from the user inputs/ instructions (e.g., navigating with the joystick, pulling the trigger on the game-pad, and moving characters with mouse movements) that control the game mechanics. The data can help us understand the pattern game-play and decision-making of players. Additionally, data analysis techniques can also be applied to other areas within the gaming field such as player engagement and understanding how players interact with the games. Therefore, the connection between data analysis and game studies is crucial for understanding and advancing the field of gaming. This data-intensive characteristic of digital games and players was highlighted in one of the pioneer works in game studies:

> "*[Game] Software is data: the data instructions to the hardware of the machine, which is turn executes those instructions on the physical level by moving bits of information from one place to another, performing logical operations on other data, triggering physical devices, and so on.*" [53, p. 2].

> "*The player, or operator, is an individual agent who communicates with the software and hardware of the machine, sending codified messages via input devices and receiving codified messages via output devices.*" [53, p. 2].

As pointed out, playing digital games can be seen as a rapid and intense cycle of inputting data through inputs like points, clicks, voice and even physical motions, and receiving output data from the system in response. Thus, gaming can be interpreted as a process of data exchange. It is important to note that this process of data exchange and the data-intensive nature of digital games is different from other forms of interactive media or activities, such as film or literature, where the data is primarily consumed rather than actively exchanged and generated by the audience.

Apart from data exchange, the data-intensive gaming also entails the chracteristics of players in terms of as a data-provider: facilitated by the necessary software and hardware, large amount of data with high levels of granularity can be easily

recorded, preserved, and processed. It has been pointed out by scholars in game studies that such player-generated data carry unique characteristics compared to the user-generated data in other domains, and result in broad implications. It has been pointed out that data analysis has played an important role in shaping game industries and markets [173, 177]

Recently, the notion of "quantified play" has been proposed [41], which can be summarized into three main characteristics:

- Voluntary: Players usually spontaneously choose to share or let their data be used, and this voluntary participation can be seen as a form of self-surveillance.
- Mundane: Beyond sophisticated and exclusively tools used by professional professionals such as esports players. Quantified play covers a larger range of "ordinary" players and data about routines and rhythms of everyday gameplay.
- Habitual: Data "grammertize and materialize" players' habits, behaviours and experiences into a form of digital records such as numbers and staticstics.

At the same time, the notion of data analysis has also had an impact on contemporary gaming culture, as activities beyond gameplay and various derivatives of games are being "data-fied." Players knowingly leave digital footprints, or generate data, outside of gameplay. For instance, the telemetry infographics, which graphically summarize the gameplay data of the total population, disseminated by game companies, have contributed to the normalization of player surveillance [161].

Player dossiers, in most cases, are third-party data collection and presentation systems that serve as a reward beyond the game's own rewards. They allow players to explore their past gameplay and play an important role in social network and player community formation by allowing players to gain social capital by creating weak ties between other players and facilitating information transfer between otherwise obscure gaming sub-groups. [114].

Metagaming systems [27], such as Steam [174], provide a platform for players to connect with one another and build social connections through various features. These systems not only track personal game-play performance but also have a focus on social functionality. One example of this is online discussion forums, where players can form communities and discuss games and play styles. Performance tracking systems also play a role in building connections, as players are able to view and compare their own performance to others. Furthermore, features such as the ability to

rate other players' profiles and trade virtual items foster social interaction and community building within the game. Displaying players' historical data is relevant to social connections, as it has been suggested that players on Steam

> "*tend to befriend those who are similar in terms of popularity, playtime, money spent, and games owned.*" [126, p. 2].

Overall, the data-intensive nature of gaming culture is a result of the advancement in technology that allows for the efficient collection, processing, and storage of data generated during both gameplay and activities outside of it. This data and the process of its generation have played a significant role in shaping player experiences and cultures. Social media also plays a role in this interplay, further emphasizing the data-intensive characteristics of gaming culture.

2.1.3 Social Media Data and Games

Social media has become a prominent venue for studying and understanding game cultures. These platforms not only provide a virtual space for players to interact with each other and form online communities, but they also offer a wealth of data that goes beyond the quantitative metrics of gameplay. Online interactions on social media can take many forms, such as sharing playful experiences, obtaining advises of game-play and purchase, showcasing achievements, and sharing fan-art. These interactions have played an important role in shaping and enriching the game culture.

When comparing to traditional gameplay data, social media data sets possess unique characteristics. They are often large in volume and have complex structures. They also often provide a more qualitative view of player experiences than gameplay data, with game reviews and discussions being in textual format, and player connections and communities being encoded in a graph structure. This type of data can offer valuable insights into various aspects of game studies, from understanding cheating behaviors on Reddit [164], to investigating the masculinist gamer identity on Twitter [39], from studying community crises on Steam [75] to understanding media enjoyment on Twitch [179].

Moreover, the characteristics of social media data share common features with gameplay data, such as the emphasis on performance statistics and self-tracking capabilities [41]. For example, on Reddit, users are quantified and tracked by the number of Karma and awards they have received and given, and on Steam, players

are quantified and tracked through their player profile, which summarizes their history on the platform. These player profiles consist of statistics such as playtime and achievements, which facilitate the self-tracking capabilities.

Players leave their digital footprints on various social media platforms, such as public player profiles on Steam and writing game reviews. This allows other players to see their accomplishments, such as the number of badges and trophies they have collected, and to learn more about their gameplay habits and opinions. Additionally, it allows players to connect with others who have similar interests and share their experiences with a wider audience.

Social media data are primarily about the ordinary players and their everyday activities related to game-play. This provides a more comprehensive view of how players interact with games and how games fit into their daily lives. The data generated from these interactions are materialized and preserved in various forms, such as player profiles, interactions, and connectivity, game reviews, comments, and discussions, which allow researchers to use this information to inform game development and improve the player experiences.

These characteristics are amplified and enriched in social media due to the chances and motivations for self-expression [127]. Players voluntarily make their Steam profiles public to "show off" to potential profile visitors. Social media data also reveal how a game can affect the habits and behaviours of players even outside of play. For example, by studying the online discussions of the game Nintendo Switch Ringfit Adventure, it was found that players not only discussed their playful experiences but also shared experiences of their daily lives when they were not playing, such as sharing recipes for making the in-game recovery items "smoothies" in real life [104]. This demonstrates that solely "quantifying" players' behaviors with numbers are statistics is not enough. Other data types such as text contain richer contents to offer a more comprehensive understanding of players.

Overall, the digital footprints left by players on social media, such as player profiles, interactions, and connectivity, game reviews, comments, and discussions, provide valuable insights into game culture, industry, and players' behavior. They are encoded and stored in different data formats and highlight the importance and value of social media data in game studies. With social media data, researchers can gain a more comprehensive understanding of players' habits, behaviors, and experiences, both within and outside of gameplay, and use this information to inform game de-

velopment and improve the player experience.

2.1.4 Focused Issues in Game Studies

This thesis mainly focuses on the following empirical issues in game studies, Player Typology and Player Experiences, and online gaming communities.

Online gaming Communities are virtual spaces, often created on social media, where individuals from different parts of the world come together to participate in online games and engage in social interaction. These communities are often based around specific games or genres, and they provide players with opportunities to form social connections with others who share similar interests and hobbies. Players may engage in various forms of social interaction, including chatting, trading items, and collaborating on game strategies.

For example, in different gaming subreddits, players in the online community discuss and share there experiences of gaming and play. Online gaming communities have been researched in terms of communications [12], nostalgia [11], and cheating behaviors [14]. In terms of data analysis, the interactions between different online communities such as common users [171] and webpage hyperlink networks [91] between subreddits can offer analytical insights such as political polarization and conflicts between online communities.

Overall, online gaming communities are complex social phenomena that continue to evolve and impact individuals and society in various ways. As the popularity of online gaming communities continue to grow, it is important for researchers and practitioners to stay abreast of the latest developments and understand the implications of these virtual communities on players' lives.

Player Typologies have been a prominent studied topic in game studies [9, 141, 51] and also other disciplines HCI [25, 103]. This notion has evolved from its root in marketing literature [64], where the purpose is looking for appropriate segmentation of customers for targeted marketing strategies. When it comes to game-play, player typologies can be used to refer to the different ways that individuals approach and engage with video games. These types are determined by a player's motivations, preferences, and behaviors. They can be used to classify players based on their unique play-related personalities or motivational structures specific to video games. Player typologies have also been used to examine the relations between game preferences and game culture [88], the aesthetic meaning and effect of game-play to players [46].

The first attempt to create a player typology was by Bartle [8] with four different observed types of players Achiever, Explorer, Socializer, and Killer, under the the context of multi-user dungeons (MUDs). Later, intensive academic efforts have been made to, e.g., synthesize established typologies [64], expanding typologies by adding new types [78], empirically verify the existing frameworks [51], or validate the usability of, for example, using player types to predict player experiences [25].

Typical strategies forming player typologies inherit the notion of segmentation, and they have been criticized as being overly dichotomous and simple, since players can have simultaneous motivations [78, 64]. A recent focus of the research is to use data analytical methods to extract underlying dimensions from data [184, 185, 166, 51], where most of the proposed methods have used survey questionnaires to players.

Player Experiences, as a research issue, has also attracted increasing research attention, the notion primarily focuses on

> "*investigating emotional, social, and cognitive components of the experience emerging from the interaction between players and a gaming system.*" [122]

Rooted in HCI, player experiences are a critical aspect of game design and are often studied when evaluating or improving the overall game-play experience [128]. The concept of player experience encompasses various elements such as flow and engagement, which are fundamental to the overall enjoyment and satisfaction of the player. Flow refers to the optimal state of immersion in which a player fully engages in the game, and their actions and challenges are well-matched [31]. Engagement refers to the player's emotional connection and investment in the game, which leads to a deeper level of immersion and enjoyment [20]. By understanding and improving player experience, game developers can create more enjoyable and satisfying game-play experiences for players.

Apart from game design, player experiences have been a key area of research as they provide insights into the meanings of games to players and how they interact within the broader framework of culture and society [110]. This research has been used to understand the ways in which players engage with and experience games, as well as how games can influence their perspectives and behaviors. For example, studies have examined the effects of "power-up" mechanics on meanings of the game [94] to its players in different moments of the game-play, the social meanings of games to players [55], and how players engage with political ideologies [56].

Measuring player experiences [123, 158] is a key challenge in research related to player experiences as they are complex and multidimensional. Recent studies have focused on the potential of using data from social media to investigate player experiences, as it can provide a large and diverse sample of players and offer insight into their complex thoughts, feelings, and behaviors [26, 136].

2.2 Representation Learning

Representation learning is a specialized area of study within machine learning that deals with discovering useful representations of data, instead of relying on human-designed or hand-picked features. The aim of representation learning is to automatically identify useful representations of data that can be used for various downstream tasks, such as classification or clustering. The learned representations usually take the form of a vector or a set of vectors, which are designed to capture the most important features of the data.

In machine learning, the term "representations" refers to a set of latent features that are not directly observed but learned from the observed data through a machine learning algorithm. Research has shown that a set of appropriately learned representations can be effective and complement the observed data in various machine learning tasks [10]. In other words, representation learning is an area of research that examines how to "*re-present*" original data to meet desired requirements through well-designed learning algorithms.

2.2.1 Beyond Feature Engineering

From the perspective of a machine learning pipeline, a task such as classification or prediction can be broken down into two main phases: the feature generation phase and the prediction model phase. The first phase is a process that focuses on selecting and extracting suitable features for use in the prediction model. Since the performance of the prediction model can be heavily dependent on the quality of the input features, this phase is often referred to as the "feature engineering" phase and is considered to be a crucial process in the overall pipeline.

Traditionally, the process of feature engineering involves a high degree of human and expert participation. Features are selected and transformed manually, and human knowledge and prior information can provide benefits. However, this process can

be extremely labor-intensive in practice. Apart from the above-mentioned, conventional feature engineering, this dimension of research is set to extract abstract, high-level, lower-dimensional representations from raw data, which can reflect essential features and capture the desired information from the original data. Representation learning has become an important field of research in machine learning.

Moreover, representation learning is an advanced approach to understanding and utilizing data, going beyond traditional feature engineering techniques. The learned representations are typically in the form of a vector or a set of vectors, which are designed to capture the most important features of the data. This approach is also useful for addressomg the problem of the curse of dimensionality, which arises when a dataset has more features than samples, leading to sparse data spaces and degraded analytical results and model performance.

Representation learning has been applied to various tasks and application areas beyond training prediction models. It has been used for data visualization and compression on large datasets [137], modeling interactions between proteins in biology [38], measuring polarization of online communities [171], mitigating biases from data [2], and enhancing privacy protection [172]. These examples demonstrate that representation learning is a powerful tool in data analysis and machine learning that can provide new insights and understanding.

2.2.2 Probabilistic Model vs. Neural Network Perspectives

According to the categorization proposed by [10], the body of literature of representation learning can be, in the perspective of understanding, categorized into two branches of research, the probabilistic model approach and neural network approaches.

The key distinction between the two branches is that the probabilistic modeling approach conceptualizes the learned representations as random variables sampled from some probability distributions whereas in neural networks approaches the representations are understood as computational nodes. In particular, the probabilistic model perspective is elucidated as a process to

> "*recover a parsimonious set of latent random variables that describe a distribution over the observed data*". [10]

Formally, let $\mathbf{Z} = \{\mathbf{z}_n\}$ denote the interested latent representation of the n-th data

point and let the $\mathbf{X} = \{\mathbf{x}_n\}$ denote the n-th observed data point. Each data point is D-dimentional, $\mathbf{x}_n \in \mathbb{R}^D$, the latent representation is K-dimensional, $\mathbf{z}_n \in \mathbb{R}^K$, and $D > K$. The observed data is generated as

$$\mathbf{X} \sim P(\mathbf{X}|\mathbf{Z}) \tag{2.1}$$

where the data are sampled from the probability distribution $P(\mathbf{X}|\mathbf{Z})$ and $\mathbf{Z}$ plays the role of governing the generation of the observed data. Using $P(\mathbf{Z})$ to denote the probability distribution describing the prior information of $\mathbf{Z}$, the joint distribution of the data likelihood and the latent representation is constructed as

$$P(\mathbf{X}, \mathbf{Z}) = P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}). \tag{2.2}$$

where $P(\mathbf{X}|\mathbf{Z})$ is the observed probability distribution of data (likelihood) and $P(\mathbf{Z})$ is the prior distribution of $\mathbf{Z}$.

The representation learning is therefore a process of searching for the posterior distribution of $\mathbf{Z}$ given the observed data $\mathbf{X}$:

$$P(\mathbf{Z}|\mathbf{X}) \propto P(\mathbf{X}, \mathbf{Z}) = P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}). \tag{2.3}$$

Note that the equation 2.3 is derive using Bayes' theorem, therefore, the process of obtaining $P(\mathbf{Z}|\mathbf{X})$ is also known as Bayesian Inference.

On the other hand, neural network models are a family of machine learning method that utilize complex, multi-layered architectures. For simplicity, a neural network model can be denoted as function $f(\cdot)$ such that

$$\mathbf{y}_n = f(\mathbf{x}_n, \theta) = h(g(\mathbf{x}_n, \theta_1), \theta_2) \tag{2.4}$$

where x is the input data, y is the outcome, and $\theta = \{\theta_g, \theta_h\}$ are model parameters. The final layers in these models, denoted as $h(\cdot)$ with parameters θ_h, known as "predictive layers", are responsible for making predictions based on the input data and the outputs generated by the preceding "representation learning layers" or "feature extractors" or "encoders", denoted as $g(\cdot)$ with parameters θ_g. Therefore, the equation 2.4 can be re-written as the composition of the predictive component

$$\mathbf{y}_n = h(\mathbf{z}_n, \theta_h) \tag{2.5}$$

and the representation learning component

$$\mathbf{z}_n = g(\mathbf{x}_n, \theta_g). \tag{2.6}$$

The equation 2.6 denotes the representation learning component, or layers that are responsible for learning and extracting meaningful representations of the input data that can be used to make accurate predictions in equation 2.5. Representation learning layers are often tasked with processing the input data and generating the outputs that are fed into the predictive layers. These models are often task-specific, meaning they are trained for a specific task such as translation or image classification. Due to their ability to achieve high performance, deep learning has gained significant attention in academia, resulting in a vast amount of literature in recent years.

2.2.3 Shallow vs. Deep Representation Learning

Despite the growing popularity of deep learning models in representation learning, another research direction that has not received as much attention is shallow representation learning, referring to models with a single-layer or few layers structure. It is often considered "cannot mine the deep information hidden in the data" [190]. However, these simpler models can offer unique advantages such as interpretability and computational efficiency. Additionally, they can still be effective in uncovering important features and patterns in data, as well as in unsupervised settings. This thesis seeks to explore the opportunities and potential of shallow representation learning techniques, and to demonstrate their value in solving real-world problems.

Overall, the shallow representation learning approach can promise the following merits:

- Simplicity: Shallow models are simpler in terms of their architecture and the number of parameters they have, which makes them easier to train and understand.
- Interpretability: Shallow models are more interpretable than deep models thanks to the simplicity. Since the learned representations are out of simpler transformation of the input features, it is easier to understand how the model is making its predictions.
- Efficiency: Shallow models are computationally less expensive and require less

data to train than deep models. They also tend to be faster to train and make predictions.

- Generalization: Shallow models can generalize better to new data, as they can risk less of overfitting to the training data like deep models might.
- Fewer overfitting issues: Shallow models have fewer parameters compared to deep models, so they are less prone to overfitting and thus generalize better.
- Handling small datasets: Shallow models are less prone to overfitting and can handle small datasets better than deep models.
- Transferability: Shallow models are able to learn simpler and more general representations of the data, which makes them more transferable to new tasks and new domains, especially when trained using self-supervised, unsupervised, or contrastive representation learning techniques.

These merits further motivate the taken research direction in this thesis, that is, to contrast with the idea of "deep" learning models this thesis investigates shallow methods.

2.2.4 Factor Models

Factor models are a canonical representation learning technique. In factor models, the high-dimensional data is factorized into a lower-dimensional representation, which consists of a set of latent factors or features that capture its underlying structure, important patterns and relationships in the data.

One of the key advantages of factor models is their ability to decompose high-dimensional data structures lower dimensional latent features that are mutually independent uncorrelated. It can not only reduces the noise of the original data, but also help to identify important patterns and relationships that may not be immediately apparent. This can provide valuable insights into the underlying structure of the data, improve accuracy, and aid in decision-making.

For text data, factor models such as matrix factorization [133, 118] can provide improved accuracy and interpretability of the underlying patterns and relationships in the data. By extracting low-dimensional features or latent factors that represent important semantic patterns, factor models can improve the accuracy of text analysis tasks such as document classification, recommendation systems, and search engines.

These models can also provide interpretable factors that can be used to gain insight into the underlying semantic structure of the data, helping to identify important topics, themes, and relationships between documents.

For social network data, factor models can provide improved scalability and flexibility for analyzing and modeling complex network structures [138, 102]. By decomposing and factorizing the adjacency matrix into smaller matrices or tensors, factor models can identify important community structures, predict user behavior, and recommend relevant items to users based on their social connections [74, 180]. These models can also be customized to different types of social network data, such as directed or weighted networks [62, 183].

The probabilistic principal component analysis (PPCA) [165] is one of the most influential canonical factor models and can be as the pioneer and iconic representation learning technique. In PPCA, the n-th observed data point $\mathbf{x}_n \in \mathbb{R}^D$ is sampled conditionally to a latent representation, from a multivariate normal distribution

$$\mathbf{x}_n|\mathbf{z}_n \sim N(\mathbf{W}^\top \mathbf{z}_n, \sigma^2 \mathbf{I}) \tag{2.7}$$

where its mean vector is constructed by the product of the latent representation $\mathbf{z}_n \in \mathbb{R}^K$ and a weighting matrix $\mathbf{W} \in \mathbb{R}^{K \times D}$. The hyper-parameter σ controls the noise in the data generation process.

Let the prior distribution of $\mathbf{z}_n$ is a standard multivariate normal distribution. The above sampling process in equation 2.7 can be re-parameterized and written as

$$\mathbf{x}_n = \mathbf{W}^\top \mathbf{z}_n + \varepsilon_n, \tag{2.8}$$

$$\varepsilon_n \sim N(0, \sigma^2 \mathbf{I}). \tag{2.9}$$

where the z_n is a lower dimensional representation of the observed data point x_n. That is, each latent vector $\mathbf{z}_n$ is generated from a non-informative, multivariate normal distribution with zero mean and an identity matrix $\mathbf{I}$ as the covariance matrix

$$\mathbf{z}_n \sim N(0, \mathbf{I}). \tag{2.10}$$

The K-dimensional vector $\mathbf{z}_n$ is a lower-dimensional representation of the D-dimensional data point $\mathbf{x}_n$ and with the observed data and the prior distribution, the posterior distribution of $\mathbf{z}_n$ can be analytically derived as

$$\mathbf{z}_n|\mathbf{x}_n \sim N(\mathbf{M}^{-1}\mathbf{W}^\top\mathbf{x}_n, \sigma^2\mathbf{M}^{-1}) \tag{2.11}$$

which is a normal distribution with the mean vector $\mathbf{M}^{-1}\mathbf{W}^\top\mathbf{x}_n$ and covariave matrix $\sigma^2\mathbf{M}^{-1}$, where $\mathbf{M} \in \mathbb{R}^{K\times K} = \mathbf{W}\mathbf{W}^\top + \sigma^2\mathbf{I}$. Moreover, $\mathbf{M}^{-1}\mathbf{W}^\top$ can be seen as the transformation matrix that can be used to recover posterior expected values of the latent representation by $\mathbf{z}_n$ linearly projecting the original data to the representation space.

Overall, factor models provide a powerful tool for gaining insights into the underlying patterns and relationships in complex data structures, helping to identify important semantic and social structures and inform decision-making in a variety of applications. The flexibility and interpretability of factor models make them well-suited for analyzing and modeling a wide range of data types, including text data and social network data, and their ability to learn informative representations of the data makes them an important tool for representation learning.

2.2.5 Topic Modeling

Topic modeling [18] is an unsupervised machine learning technique modeling the latent structures in the collected text documents. The basic notion is that, each document d is "represented" by a vector θ_d contains the proportions of K topics, or underlying themes. In other words, that each document is assumed to be generated from the mixture of various unobserved topics. Furthermore, each topic is a distribution over words in a vocabulary and each word in the document is generated from one of those topics.

Topic modeling has been widely recognized as a powerful technique that can help us uncover hidden patterns in text data in many domains, including game studies [47, 65, 105], especially when analyzing large-scale text data. By identifying common themes or topics within a corpus of text data, we can gain insights into the underlying structure of the data and make predictions about its properties.

Another benefit of topic modeling, similar to other representation learning techniques, is that it can help us reduce the dimensionality of the analyzed data [36]. By distilling large volumes of text data into a smaller number of topics or concepts, the data can become easier to analyze, visualize and facilitate human interpretation. This is particularly useful when we need to identify the most important topics or

concepts in the data, such as when improving search results [106] or facilitate iterative decision-making when document-based relevance feedback are available [6].

Apart from the aforementioned features, topic modeling can also help protect sensitive information and preserve privacy [191, 153]. By representing data in terms of an abstraction, the underlying topics or themes, rather than original data containing specific words or phrases, exposing sensitive information can be avoided while it is still possible to understand the theme of collected document. This is particularly important when working with data that contains personally identifiable information or other information may arouse privacy concerns.

As mentioned, in topic modeling, each document is represented with the θ_d vector. The goal of a topic model is to infer, or discover the proportion of the topic mixture for each document and the word distribution of each topic over the vocabulary, based on the observed collection of documents. In principle, the data generation process of each document is as follows:

- For each topic k, generate a distribution over words, represented by a vector of topic probabilities $\beta_k \sim Dir(\gamma)$, where $\beta_{v,k}$ is the probability that the word v in the document comes from topic k.
- For each document d, generate a topic mixture according to $\theta_d \sim Dir(\alpha)$, where $\theta_{d,k}$ is the probability that a word in the document d comes from the k-th topic.
- For the n-th word in the d-th text document, do the following:
  - Choose a topic z_n for the word, according to the topic mixture representation vector θ_d.
  - Choose a word from the vocabulary according to the chosen topic. Suppose $z_n = k$ then the word is selected from the distribution β_k specific to the chosen topic.

Above, α and γ are hyper-parameters that control the smoothness of the topic and word distributions. This representation learning framework has also been extended for various purposes such as modeling correlation between topics [15], incorporating the observation time of documents [17] and author information [182].

2.2.6 Embedding Models

Another imperative line of research on representation learning is based on learning the representations that can predict the interactions between the observed item and its contexts. This representation learning approach is often called embedding models. Such models, also known as vectorized models, map high-dimensional data into a lower-dimensional vector space. By doing so, embedding models can capture important patterns and relationships in the data and represent them in a more manageable format. This makes them a powerful tool for a wide range of applications. This approach was first introduced in NLP as a language model, where the observed items and their contexts are different words [116]. The same notion and modeling framework have been later generalized to model other different co-appearance patterns of different items such as human genes [40] and online communities [171].

In embedding models, the main idea is to assign optimized embedding vectors to each different possible data item, e.g., a word in the context of NLP or a node in graph-structured data, including the center item and its context items [101]. items. Models such as word embedding [116], BERT and ELMo [100] has made impact expecially in NLP. This notion has also been applied to non-textual data [147].

One of the main benefits of embedding models is their ability to capture semantic relationships between data points. In the case of text data, for example, embedding models can capture the meaning of words and phrases based on their context, allowing for tasks such as language modeling, sentiment analysis, and machine translation. In the case of social network and graph data, embedding models can capture important features of the data such as community structure, user preferences, and social connections, allowing for tasks such as link prediction, node classification, and network visualization.

The pioneer work, word2vec was introduced by [116] is used in the context and modeling word co-occurrence. Let i and j denote the indices of two items (such as two words, genes, or communities) where the item j is the context of the item i. For example, in the item i can be a word in a piece of text, and item j can be any other word around it. Let $D = 1$ denotes that the item i and j are present in the same context, and let $D = 0$ indicates the opposite. The probability of the co-appearance ($D = 1$) of items i and j is defined as the sigmoid transformation of the inner product of their corresponding embedding vectors ρ_i and ρ_j, modeled as

$$P(D=1|i,j) = \left(\frac{1}{1-e^{-\rho_i{}^\top \rho_j}}\right). \tag{2.12}$$

Similarly, if item i does not exist ($D=1$) given the context time j, which means the non-existence of the item pair (i,j) the probability is therefore becomes

$$P(D=0|i,j) = 1 - P(D=1|i,j) = \left(\frac{1}{1-e^{\rho_i{}^\top \rho_j}}\right). \tag{2.13}$$

Since it is computationally expensive to consider all the non-existing pairs, negative sampling or sub-sampling is a commonly used practice when training embedding models. The integrated objective function becomes

$$\begin{aligned}
&\arg\max_{\rho} \prod_{(i,j)\in D} P(D=1|i,j) \prod_{(i,j)\in D'} P(D=0|i,j) \\
=&\arg\max_{\rho} \sum_{(i,j)\in D} \log p(D=1|i,j) + \sum_{(i,j)\in D'} \log p(D=0|i,j) \\
=&\arg\max_{\rho} \sum_{(i,j)\in D} \log\left(\sigma\left(\rho_i{}^\top \rho_j\right)\right) + \sum_{(i,j)\in D'} \log\left(\sigma\left(-\rho_i{}^\top \rho_j\right)\right).
\end{aligned}$$

In text data analysis, embedding models can help identify important semantic relationships between words and capture similarities between them, allowing the model to predict the meaning of new words based on their context. This can help with tasks such as document classification, named entity recognition, and question answering. Embedding models can also be applied to graph-structured and social network data [135] to identify community structures [28], predict user behavior [120], and recommend relevant items to users based on their social connections [58]. Examples of social network analysis tasks that can be performed using embedding models include link prediction, community detection, and network visualization.

Overall, embedding models are a flexible and powerful tool for modeling and analyzing a wide range of data types, and their ability to capture important patterns and relationships in the data makes them a valuable tool for a variety of applications.

Exponential family embedding (EFE) models can be seen as another canonical general-purpose embedding model compared to word2vec, in which the exponential family distribution is introduced to extend the embedding models to various domains. In EFE, the data points are generated as samples from an exponential family

distribution where the natural parameters are governed by the embedding ρ and context vectors α. That is, it can model not only the appearance of the item i given its context but also the values carried by the observed item and its contexts.

Let $x_n^{(i)}$ denote the value (a single value in EFE) of the item i at the location n, which has its context $\mathbf{c}_n$. In exponential family embeddings, the value of $x_n^{(i)}$ is generated from an exponential family distribution depending on its context $\mathbf{c}_n$ and their corresponding values $x_{\mathbf{c}_n}$.

$$x_n^{(i)} | \mathbf{c}_n \sim \textbf{ExpFam}\left(\eta_n\left(x_{\mathbf{c}_n}\right), t\left(x_n\right)\right) \tag{2.14}$$

where **ExpFam** can be any exponential family distribution with a corresponding link function g, $\eta_n\left(x_{\mathbf{c}_n}\right)$ is the natural parameter of the distribution, and $t\left(x_n\right)$ denotes the sufficient statistics.

The natural parameter is modeled as a function of an inner product of the embedding vector ρ and the context vector α so that

$$\eta_n\left(x_{\mathbf{c}_n}\right) = g\left(\rho_i^\top \frac{1}{|\mathbf{c}_n|} \sum_{n' \in \mathbf{c}_n} x_{n'}^{(i')} \alpha_{i'}\right). \tag{2.15}$$

As the exponential family offers the flexibility to model different observation distributions, the embedding models are no longer limited to modeling co-appearance (binary) observations. The EFE model has been applied to different domains such as grouped data [146] and graph-structured data [29].

2.2.7 Variational Autoencoder

Another notable approach is called variational autoencoder [84]. A variational autoencoder (VAE) is a generative model for representation learning when the true posterior distribution of $p_\theta(\mathbf{z}|\mathbf{x}) \propto p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})$ is be intricate. It is a probabilistic model that combines the strengths of autoencoders and variational inference. It consists of two parts: an encoder, which maps the input data $\mathbf{x}$ to a hidden representation $\mathbf{z}$, and a decoder, which maps the hidden representation back to the original input.

The primary notion of VAE is that it extends the standard autoencoder by introducing a probabilistic interpretation of the hidden representation $\mathbf{z}$. The encoder maps the input data to a variational distribution $q_\phi(\mathbf{z}|\mathbf{x})$ over the hidden representation, typically a Gaussian distribution where the parameter ϕ controls the mapping

process. The decoder is then trained to generate new samples from this distribution $p_\theta(\mathbf{x}|\mathbf{z})$, which can be used to reconstruct the original input data, where the generative process is controlled by parameter θ.

The key concept of VAE is that instead of learning a deterministic hidden representation $\mathbf{z}$ for a given input $\mathbf{x}$, it learns a probability distribution of the hidden representation given the input. This is done by assuming that the hidden representation $\mathbf{z}$ is sampled from a simple distribution (e.g., a Gaussian) with parameters that depend on the input x. Then the encoder network is trained to learn the parameters of this distribution. Mathematically, let $\mathbf{x}$ is the observed data and $\mathbf{z}$ denote the latent representations, the VAE can be formulated as:

- The encoder network learns the parameters of the recognition mode, a conditional variational probability distribution $q_\phi(\mathbf{z}|\mathbf{x})$
- The decoder network learns the parameters of a conditional probability distribution $p_\theta(\mathbf{x}|\mathbf{z})$.

The overall goal is to maximize the likelihood of the data, which is intractable to compute directly. To overcome this, VAE uses an alternative approach. Following the paradigm of variational inference [16], starting with the discrepancy between $q_\phi(\mathbf{z}|\mathbf{x})$ and the exact posterior distribution $p_\theta(\mathbf{z}|\mathbf{x})$:

$$\begin{aligned} KL(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x})) &= E_{z\sim q(\cdot|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}\right] \\ &= E_{\mathbf{z}\sim q(\cdot|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})p_\theta(\mathbf{x})}{p_\theta(\mathbf{x},\mathbf{x})}\right] \\ &= \log p_\theta(\mathbf{x}) + E_{\mathbf{z}\sim q(\cdot|\mathbf{x})}\left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{x},\mathbf{z})}\right] \end{aligned}$$

where KL is the Kullback-Leibler divergence, which measures the difference between two probability distributions. The Evidence Lower Bound (ELBO) can be derived as:

$$L\left(\phi,\theta|\mathbf{x}\right) = E_{\mathbf{z}\sim q(\cdot|\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] = \log p_\theta(\mathbf{x}) - KL(q_\phi(\cdot|\mathbf{x})|p_\theta(\cdot|\mathbf{x})). \tag{2.16}$$

Maximizing ELBO is equvilent to maximize the data likelihood $\log p_\theta(\mathbf{x})$ and

minimizing $KL(q_\phi(\mathbf{z}|\mathbf{x})|p_\theta(\mathbf{z}|\mathbf{x}))$. Therefore, a VAE model can be trained by maximizing ELBO as the objective function (optimizing ϕ and θ).

The sampling in $E_{\mathbf{z}\sim q(\cdot|\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right]$ can be done by using the reparametrization trick and stochastic gradient descent. For example, if the probabilistic encoder $q_\phi(\cdot|\mathbf{x})$ is normally distributed as $N(\mu, \mathbf{\Sigma})$ where $\phi = \{\mu, \mathbf{\Sigma}\}$. The data generating process $\mathbf{z} \sim q(\cdot|\mathbf{x})$ can be done with

$$\mathbf{z} = g_{\mu,\ \mathbf{\Sigma}}(\varepsilon) = \mu + \mathbf{L}\varepsilon,\ \ \varepsilon \sim N(0, \mathbf{I}) \tag{2.17}$$

where L is the Cholesky decomposition of $\mathbf{\Sigma}$ such that $\mathbf{\Sigma} = \mathbf{L}\mathbf{L}^\top$. The ELBO can be estimated as

$$E_{\varepsilon\sim N(0,\mathbf{I})}\left[\log \frac{p_\theta(\mathbf{x}, \mu + \mathbf{L}\varepsilon)}{q_\phi(\mathbf{z}|\mu + \mathbf{L}\varepsilon)}\right]. \tag{2.18}$$

The gradients used to update ϕ and θ can be obtained with chain rule. This principal framework of VAE has been widely applied to various machine learning tasks such as text classification [181] and modeling physical quantities [132].

2.3 Representing Games, Players through Social Media

Games and play have been data-intensive. Player-generated social media data can provide insights into games, players, and relevant contexts, and can influence player behavior and gaming culture. For example, online discussions can not only reveal players' opinions but also shape players' behavior, and the formation of online communities has played an important role in the landscape of game culture. Data analytics provides tools to investigate the interplay of data and game-play and enhance the understanding of games and players.

When it comes to data from social media, both the volume and granularity of digital data are growing, for example, the Steam platform contains not only the games but also the player reviews and their personal profiles into which the player experiences and the refined game-play history are encoded. Computational approaches are useful to process such data and obtain insights.

The complexity of data from social media can result in both difficulties and blessings. It increases the difficulty during data processing and analysis. On the other hand, if the model can handle and capture such complexity appropriately, better

insights can be obtained. The gained insights can help us better understand game culture. Representation learning is the tool of focus in this thesis to deal with the complexity of such data. That is, representing games, players with abstract but meaningful features that carry sufficient and distilled information.

It is worth noting that, at the crux of machine learning solutions and the above application area, there is the tension between explanation and prediction [68]. As machine learning has long focused on prediction performance, the ability to make sense of data and interpret models has been often ignored. Well-designed representation algorithms should offer a learning process that can integrate sufficient human insights and enhance the interpretation.

This section further outlines the data types that are dealt with in this thesis. The challenges when analyzing those data are also outlined. Note that the discussed data types and challenges can only cover a part of reality, as the real domain is often more complex. This will be discussed in more detail in Section

2.3.1 Data Types

The thesis primarily focuses on the data types as follows. It is important to note that in the world of games and play, there are other data types beyond those listed, e.g., player telemetry [189, 154]. Nevertheless, the following data types are the most important in social media and are often underutilized or understudied.

2.3.1.1 Numerical Data

Social media data contains a variety of numerical measurements. For example, posts on Reddit are measured with the number of shares and comments, and users are measured with number of karma points, which reflects the user's popularity [121] or reputation [82].

Another example is the player profiles on Steam. Player profile records showcase the history of the player, numerical values such as the number of badges and achievements reflect the player's skills and numerical values such as the number of friends and workshops reflect the player's social activity and social capital.

Those numerical values are often unstructured (without dynamic data schema) and mixed with different types of data. Online social media platforms often contain different types of data. For example, a Twitch streamer can have the statistics such

as their number of views (integer), and the lifetime of the streaming (real-valued) [145].

2.3.1.2 Text Data

A considerable volume of online data is in text format, including online game reviews and discussions. Compared to numerical values, the text is able to carry richer and more diverse content. Moreover, user discussions in online forums have both theoretical and practical implications. They can help improve our understanding of collective thinking while also facilitating practical applications such as improving user experience, increasing engagement, and supporting the democratic process. [5, 115] In particular, it has been suggested that game reviews are "one of the primary forms of videogame journalism" and serve a broader role than just being "shopping guides" as they cover diverse themes and offer game design suggestions, advice for enjoying games, and insight into game creators' intentions. Game reviews also contextualize the historical connections between games and help preserve video game history. [187].

The reviews and online discussions can also reflect players game related activities such as virtual purchases [23]. Before owning the game, other players' opinions affect the decision of possessing the game. After acquiring the game, the "tips and tricks" learned from other players' previous experiences [69] influence the strategy and style of play when actually playing the game. The experiences gained out of playing can become another review or part of the discussion online that has the potential to influence other players.

Online game reviews are a valuable source to understand players' experiences [26, 186]. For example, on the gaming platform Steam, players can write reviews to reflect on their experiences after they have purchased the game. Game reviews provide an interface to understand not only what is in the game but also what the game has brought to the players while playing, and after playing the game. Those reviews contain an abundance of information on players' experiences. However, the large amount of users leads to a large volume of written text reviews. Challenges arise when it comes to analyzing them with pure human effort.

2.3.1.3 Graph Structured Data

Data from social media are often stored in a graph structure. The graph structure can effectively preserve the interactions between entities. In general, a graph $G = (\mathbb{V}, \mathbb{E})$ is composed of a set of nodes $\mathbb{V}$ and a set of edges $\mathbb{E}$. Each node $v_i \in \mathbb{V}$ denotes an entity and each edge $e_{i,j} \in \mathbb{E}$ describes the interaction between nodes v_i and v_j. In some graphs such as a knowledge graph, each edge or node can carry attributes.

Graph data are ubiquitous in games and social media. For example, interactions between Twitch steamers [145] and Reddit hyperlink networks [91] can be both encoded as graph-structured data sets. One common challenge in representation learning regarding graph-structured data is to construct vector representations for nodes according to the interactions between nodes. A set of properly learned node representations can be used as input features in downstream tasks such as node classification and link prediction between nodes.

2.3.2 Key Challenges

The above-mentioned data types in social media lead to the following analytical challenges:

2.3.2.1 Latent Temporal Dynamics

How the data has changed over time is an interesting topic in social media data analysis. For example, in Reddit, the timestamps are often collectible and the data are often collected over a period of time. These features bring opportunities for understanding the evaluations of players' perceptions and experiences over time.

2.3.2.2 Multiple Sources

Social media data are not often centralized. Therefore, in some situations, it is needed to analyze data from different sources. For example, a game can be discussed in different places (Reddit, Steam, and Nintendo Forums). The major challenge when aggregating the data from multiple sources is to appropriately model both the shared and distinct (source-specific) patterns.

2.3.2.3 Cross-structure Modeling

Different representation techniques can be used to capture different underlying structures of data. How to appropriately learn those structures from data and how to model different-structured representations interact are challenging tasks. Nevertheless, the learned structures can offer valuable insights into data analysis, as various structures can facilitate the interpretation by providing multiple perspectives on the data.

2.3.2.4 Multiple Representations

It is natural that each entity can pose different characters and roles in different circumstances. For example, a player can play different roles in different games, or a word can have different meanings in different contexts. The multiple representation setting allows the flexibility of models to better capture uncertainty of how data arise from latent phenomena.

2.3.2.5 Heterogeneous Data

In social media, the collected data can be in mixed types. For example, text documents often carry metadata alone with the text such as the author's information, or document attributes such as the number of likes of a post. Modeling the interactions between the metadata and the text content yields various research opportunities. Similarly, in a graph, a node in a graph can contain attributes of different types.

3 EMPIRICAL ANALYSIS OF GAMES AND PLAY

This chapter demonstrate how representation learning can be used to **disentangle the dependencies of data** and **model temporal dynamics** with two empirical analysis works for game studies. Specifically, this chapter summarizes two empirical data analysis cases that utilize two representation learning techniques: the factor model and the topic model are presented.

The studies in Publication I uses a factor model to explore player typologies based on user profile data from Steam, a popular digital game distribution platform. The study demonstrates how factor models can be used to obtain insights into player behavior and preferences.

In Publication II, the structural topic model [143] (STM), a topic modeling technique is used to model and analyze the temporal dynamics of player discussions in response to updates and changes in the game *No Man's Sky* [66]. By leveraging topic modeling, the study gains a better understanding of how players' perceptions have changed in reaction to changes in the game and how these responses have evolved over time.

3.1 Publication I: Extracting Player Factors from Steam Profiles

Publication I leverages a factor model and Steam profiles to investigate player typologies. The factor model enables more flexible player typologies and the Steam profiles are proven to be a valuable data source from social media. In total, 8 player meaningful factors are extracted.

3.1.1 Factorized Player Typologies, and Steam User Profile

The exploration of player typologies has been a significant research topic in game studies. Understanding player behavior, motivation, and experiences is crucial, and the conventional strategy of classifying players, such as Bartle's four-player types

model [8], has faced criticism for its inflexibility and "clear-cut" categorization. Furthermore, players are unlikely to strictly belong to a specific type as they can have simultaneous motivations and traits [78, 64]. The notion of player typologies is rooted in segmenting players [8], but this approach has been criticized for being dichotomous and overly simplified.

Recently, several player typology frameworks have been proposed based on computational factor models, where the player typologies are captured as latent dimensions [51, 166, 184, 185] where each player is viewed as a composition of various factors.

The analysis in Publication I further leverages Steam profiles to extract player factors. This approach extensively utilizes online data and is not limited to using survey questionnaires. The Steam user profiles offer a good resource for studying player typologies from the perspective of how different components distribute among the player profiles.

In Publication I, a factor analysis model is conducted on a collection of 60267 unique user profiles. The data collection was collected in a "snowball" manner. A web crawler started from one randomly selected user from the leaderboard of top 10 Steam user. The crawler then iteratively go to the list of the user's friends profile URL. The list of users' URL was grown via crawling the friends of each of the existing users on the list. The features that were extracted through crawling include various elements such as Levels, Showcases, Badges, Number of Games, Screenshots, Workshop Items, Videos, Reviews, Guides, Artworks, Groups, Friends, Items Owned, Trades Made, Market Transactions, Achievements, Perfect Games, and Game Completion Rate, and Profile Customization.Profile Customization numerically summarizes four binary profile customization related variables: Avatar, Status, Background, and Favorite Badge customization (customized or not). For example, for a user who customized two of the four customizable items, the value of Profile Customization is then set to 0.5. In addition, to take the user activity into the account, each user's active time span was also collected, using the Steam API[1], based on the time when the user last logged off and the time when the user account was created. The profile duration of the account was further computed and utilized to normalize the profile features, by simply dividing each feature by the profile duration.

[1]https://steamcommunity.com/dev

3.1.2 Extracted Latent Player Factors

An exploratory factor analysis (EFA) [81, 134, 156] is employed to identify latent player factors among variables in the Steam user profiles. EFA allows for the reduction of data complexity and the discovery of relations between variables. The parallel analysis (PA) [70] technique was used to determine the number of factors, which has been shown to be effective in recent research. PA employs Monte Carlo simulation to create random samples of uncorrelated variables that are parallel to the observed data. In this study, the parallel analysis task was conducted with 5000 simulations and 8 latent factors are extracted. Moreover, to simplify the interpretation of the factor analysis results, the *varimax* technique [72] was employed for rotation, which maximizes the variance of each factor loading. The extracted factors and corresponding factor loadings can are displayed in Table 3.1 and 3.1. The names of the extracted factors were given by the authors of Publication I, the discussion was led by Xiaozhou Li who served the role as the fist author in the study. The process was based on the analysis of each factor loading on different variables. Each extracted factor is described as follows:

- **Elite**: focuses on social comparison and enhancing quantifiable social scores;
- **Achiever**: prioritizes mastering games and completing them thoroughly
- **Provider**: enjoys providing facilitation to others with gameplay guides (Guides) and game-related arts (Artwork)
- **Completer**: which focuses on finishing games and showcasing possessions
- **Improver**: which focuses on improving games through workshop items and reviews
- **Trader**: buys and sells game-related virtual items
- **Belonger**: focuses on social belonging and profile customization
- **Nostalgist**: which restores gameplay memories through screenshots and videos

It is worth noting that the eight factors aim to explore the various attributes of Steam users instead of arbitrarily categorizing each user into a single type. With this framework, an individual player can be represented by the strength on each factor and further visualized by a radar chart such as Figure 3.1 that illustrates their salient attributes. By reducing the original number of variables to the strength on the 8

Table 3.1 Loadings of the Extracted Factors (1)

Variable	Elite	Achiever	Provider	Completer
Level	**0.641**	-0.005	0.004	-0.002
Showcases	0.026	0.107	0.065	**0.828**
Badges	**0.954**	0.033	0.004	0.010
Games	0.019	**0.511**	0.020	0.016
Screenshots	-0.000	0.118	0.332	0.046
Workshop.Items	0.007	-0.045	0.042	0.127
Videos	0.002	-0.030	-0.066	0.046
Reviews	0.002	0.232	0.039	0.044
Guides	0.002	0.024	**0.879**	-0.031
Artwork	0.004	-0.010	**0.836**	0.101
Groups	0.078	0.017	0.020	0.031
Friends	**0.947**	0.002	0.004	0.043
Items.Owned	0.004	0.048	0.005	0.049
Trades.Made	-0.003	-0.142	-0.002	0.281
Market.Transactions	0.017	0.116	0.001	-0.063
Achievements	0.005	**0.865**	0.014	0.125
Perfect.Games	0.003	**0.847**	0.006	0.210
Game.Completion.Rate	0.008	0.274	0.013	**0.852**
Profile.Customization	**0.808**	-0.007	-0.008	-0.019

Table 3.2 Loadings of the Extracted Factors (2)

Variable	**Improver**	**Trader**	**Belonger**	**Nostalgist**
Level	0.008	-0.013	-0.263	0.002
Showcases	0.162	0.180	0.028	0.067
Badges	0.006	0.043	0.016	0.004
Games	0.108	0.365	0.030	0.088
Screenshots	0.344	0.039	0.022	**0.490**
Workshop.Items	**0.789**	-0.027	0.003	-0.082
Videos	-0.074	-0.022	-0.003	**0.901**
Reviews	**0.769**	0.039	0.018	0.113
Guides	-0.090	-0.003	-0.001	-0.002
Artwork	0.192	0.006	0.018	0.030
Groups	0.026	0.008	**0.951**	0.009
Friends	0.007	0.014	0.202	0.001
Items.Owned	-0.004	**0.733**	0.006	-0.022
Trades.Made	-0.063	**0.551**	0.003	-0.061
Market.Transactions	0.044	**0.645**	-0.007	0.049
Achievements	0.014	-0.010	-0.001	-0.011
Perfect.Games	0.105	-0.045	-0.002	-0.017
Game.Completion.Rate	0.054	-0.004	0.003	0.021
Profile.Customization	-0.015	-0.016	**0.553**	-0.007

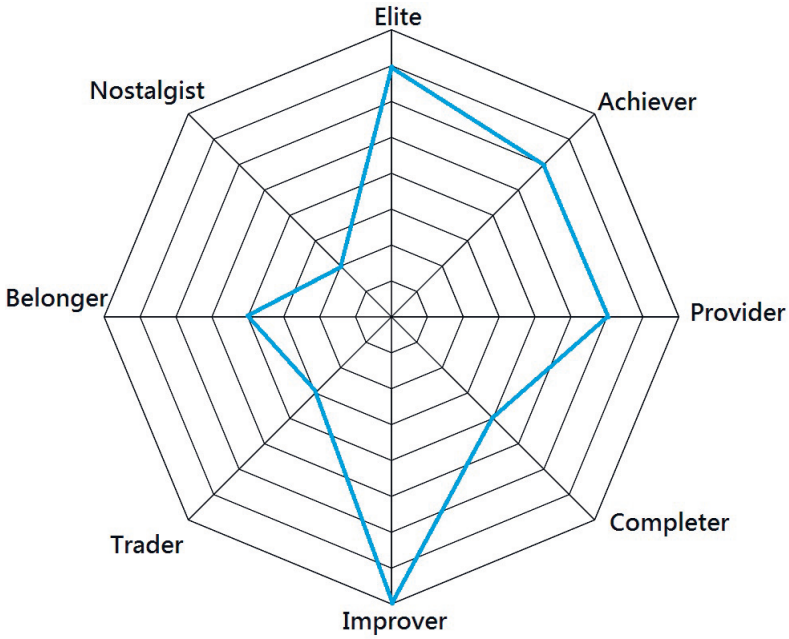

Figure 3.1 An Example of a User Preference Attributes Radar Chart

factors and normalizing the value, users can see their unique distribution over latent player factors. For example, in Figure 3.1, the user with a salient attribute of being an improver, being creative with workshop items, and contributing to improving games through reviews. The user also possesses relevantly strong attributes of being an elite, achiever, and provider. This indicates that the user favors gaining levels, badges, and achievements, and providing guides and artworks to the community.

Moreover, this framework can be further applied in personalized gamification design facilitated by a better granularity of the understanding of each player. Connections can be identified between attributes and established intrinsic motivation types, as well as other similar gamification design models or frameworks, based on the variables each attribute is associated with. Personalized gamification design may vary depending on the player's motivation and the design elements frameworks that are utilized.

3.2 Modeling Temporal Changes in Game Reviews

Publication II focuses on players' perceptions to game updates and changes. The game *No Man's Sky* was selected to analyzed as it exemplifies a game with constant changes including ongoing commitment to maintain and update. The work analyzes 85805 unique user reviews on *No Man's Sky* from its release date, August 12th 2016, to October 5th 2019.

3.2.1 Game Evolution and *No Man's Sky*

Software evolution is crucial for maintaining software quality, and with the adoption of incremental and agile development methods, user feedback plays a vital role in software product evolution [59]. Effective release planning is thus essential, and numerous studies have contributed to the practice of software release planning, particularly for mobile applications [148, 149, 160, 124, 168, 35, 151]. Similarly, online distribution platforms for video games allow developers to receive and respond to player feedback, making proper release planning critical, particularly for Early Release games [98].

In general situations, as a piece of software, digital games follow the principles of software evolution. A body of literature has addressed the challenges and issues of game development practice from software engineering perspectives [3, 79]. For example, assets, scopes, process, publishing, management, team organization, and third-party technology have been identified as the primary challenges in game development [79]. It has been suggested that game developers often attempt to adapt traditional software engineering methods to game development with solely certain adjustments, and tend to ignore the maintenance and verification [4]. Moreover, updating games correctly plays an important role in the process of perfecting the game and achieving better customer satisfaction [80].

More specifically, from a game-development perspective, the evolution of individual games over their life-cycle can be categorized as

- Emerging change: Creating, or designing a space for the players to mold their own game experiences.
- Reactive change: The changes are in response to direct or indirect feedback from the players.

- Pre-planned change: The changes of content that are already planned , designed, or even already produced, before the launch of the game.

The categories are not necessarily equal, but in reality they can overlap during the evolution due to the shared similarities [125].

The game which is analyzed, *No Man's Sky*, is an example of a mixture of the mentioned changes. It is an action-adventure survival game that was initially launched in August 2016. In the beginning, the game received strong criticism from players due to the lack of features that had been promised to be included. However, since its launch till the data collection in Publication II was conducted (October of 2019), the game has been continually updated with eight major updates to date, labeled as versions 1.00, 1.10, 1.20, 1.30, 1.50, 1.70, 1.75, and 2.00. These updates were released on 12 August 2016, 26 November 2016, 8 March 2017, 11 August 2017, 24 July 2018, 29 October 2018, 22 November 2018, and 14 August 2018, respectively [2].

On the other hand, user feedback is critical for improving products and services [34, 178, 24, 89]. The combination of collectable end user feedback and traceable software evolution allows effective requirements analysis through data analysis [130]. How to effectively analyze reviews to uncover critical user needs has been shown to be a prominent issue in many studies [52, 32, 63, 97], including video game user reviews [98].

In Publication II, a detailed investigation of player reviews was conducted, examining 85805 game reviews of *No Man's Sky*. The collected data spanned a period of time from August 12th, 2016, to October 5th, 2019. Among them, over half (44335) of the reviews were given within the first month of the game release and with 59.14% of the reviews that did not recommend the game. During the timespan of the data set the overall recommendation rate had increased to 53.03%. Besides review text and the recommend/not recommend flag for the review, features including review publication date, and the user's play hours were also collected for data analysis.

The analysis revealed a diverse range of topics discussed by players, along with temporal patterns of topic prevalence that emerged over time. Moreover, the study also revealed notable variations in temporal patterns between reviews that recommended the game and those that did not. The research further demonstrated how

[2]https://nomanssky.gamepedia.com/Patch_notes

updates to the game coincided with shifts in player discussions and presented concrete examples of how these updates might affect player feedback. These findings provide valuable insights into the dynamic nature of player reviews and the impact of game updates, offering promising prospects for future research in this domain.

3.2.2 Structural Topic Modeling

The collected data contains not only text reviews but also document-level covariates including: the user recommendation (recommend or not) indicates the general positive/negative evaluation of the game, thus it is taken as one of the covariates; we also take the posting time as a co- variate in order to model the evolution of the review content over time. To leverage this information, we chose a more advanced topic model STM. This technique employs machine learning-based topic modeling [18] to model each document as a combination of latent topics. As introduced in Section 2.2.5, by identifying these latent topics, topic modeling enables us to identify topics present in a corpus of documents that contain a diverse range of topic combinations. This approach is more flexible than hard clustering, which assumes that each document belongs to a specific cluster, as it represents a document in a more adaptable way. Each topic is represented as a distribution over words, and the likelihood of a word w appearing in a document d is estimated. The data generative process can be summarized with the joint probability:

$$p(w|d) = \sum_{k} p(k|d)p(w|d) \tag{3.1}$$

where $p(k|d)$ is the probability that the word coming from topic k out of all possible K topics. Such topics are not pre-specified by humans but are automatically learned by fitting the model to the data. Unlike, for example, principal component analysis, topic modeling is inherently designed for count data such as word counts in text documents. Topic modeling has been used in many domains including game studies (e.g. [45]).

STM is used to model and analyze how the topic prevalence is affected by document-level covariates. It models the topic prevalence of a document $p(k|d)$ with a vector $\theta_d = [p(1|d), \ldots, p(K|d)]^\top$ drawn from a distribution that depends on the covariates so that

$$\theta_d \sim LogisticNormal_{K-1}(\mathbf{\Gamma}^\top \mathbf{x}_d, \mathbf{\Sigma}) \tag{3.2}$$

where $\mathbf{x}_d$ is document-level covariates, $\mathbf{\Sigma}$ is the covariance matrix, and $\mathbf{\Gamma}$ is the coefficient matrix governs the interaction between topic prevalence and document-level covariates.

The only user-specified parameter which needs to be set up when training an STM model is the number of the topics K. To decide it, we assessed the held-out likelihood value as the criterion: to compute it, a subset (here 50%) of the documents is considered unobserved ("held out"), and the models are evaluated by their likelihood on this held-out portion. For each model setting, from $K = 5$ to $K = 100$, the held-out likelihood is computed 10 times with random initialization, and the final number of topics is chosen as the value where the held-out likelihood plateaus. We then optimize STM with the final K from 10 random initializations, choosing the result with the best semantic coherence [117] as the final model.

3.2.3 Extracted Themes

In total, 55 topics have been identified, covering a wide range of discussions. A selection of topics with their top words are shown in Table 3.3, the full table can be found in Publication. Each topic is further labeled through examining the its top words and example quotes (i.e., documents with high prevalence of the topic). Some topics such as **Evaluating Game-play** and **Moving and Looking** reflect general gaming activities. Other topics reflect positive experiences, such as **Enjoyment of Play Experience** and **Appreciation**, or negative experiences, such as **Bugs and Glitches** and **Disappointment of Promise and Hype**. Some topics are related to specific game mechanics, such as **Spaceship Travel and Combat**, **Material Collection**.

Each of the extracted topics reflects a meaningful aspect in game changes. Besides, the STM enables the modeling of the temporal dynamics of topic prevalence over time. Figure 3.2 displays the changes over time of a subset of the extracted topics among players who recommend or do not recommend the game. From these modeled dynamics, it is worth noting that players' perceptions are changing over time and these changes can potentially be affected by updates to the game. There are extracted topics such as **Updates and Added Content**, **Changes in Game**, and **Upgrades and Items** that are directly related to game updates. Furthermore, the

Table 3.3 A selection of extracted topics

Topic	Pr(%)	Top Words
Evaluating Game-play	4.46	play fun get buy hour pretty first bore good like couple look day soon gameplay
Disapointment of Promise and Hype	2.62	promise hype wait buy deliver disappoint title worth live hope game pay trash huge preorder
Appreciation	2.30	love time play start hour keep cool idea absolutely feel always beautiful first put experience
Enjoyment of Play Experience	2.30	enjoy explore like far bite thing play look feel may relax slow find although kind
Updates and Added Content	1.93	new update add game bring content community future major forward feature stick atla foundation improvement
Change of Game	1.80	stuff good make need like decent super lot big easy yet little still take slowly
Moving and Looking	1.57	around turn take away look see like move head walk way hit one figure blow
Bugs and Glitches	1.44	bug save break progress time con pro fix buggy hour glitches play start many file
Spaceship Travel and Combat	1.14	ship mine fly space land planet station tool fuel sell resource attack multi weapon combat
Material Collection	1.11	life find every material planet farm start time need minute thing walk take tutorial except
Upgrades and Items	1.06	ship upgrade inventory item slot suit fight find sentinel trade management space system need blueprint

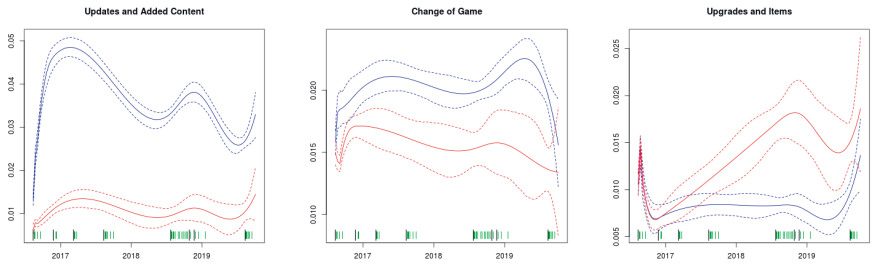

Figure 3.2 Selected topic prevalence over time.

difference in topic prevalence between players who recommend the game and those who do not recommend it shows that the overall evaluation can be either positive or negative once the player has experienced the updates. The findings have shown that the updates have played an important role in the life-cycle of *No Man's Sky* especially after the game was launched.

The study's findings highlight the role of updates and changes of a game in shaping players' perceptions. It also highlights the importance of creating a continuous and engaging experience for players.

4 LATENT FACTORS AS REPRESENTATIONS

This chapter tackles the research addresses regarding **disentangle the dependencies of data, cross-structure learning,** and **modeling temporal dynamics** with a focus on factor models. The objectives are approached from the perspectives of methodology development. It summarizes the methodological advances presented in Publications III and IV, which focus on the development of factor models for analyzing text data. Publication III proposes a combination of non-negative matrix factorization and Gaussian Process Latent Variable model to model the temporal dynamics and cross-domain relationships of text collected from multiple sources over time. Publication IV proposes a combination of factor model and topic model to analyze text data with document-level covariates. Note that this chapter highlights the developed methods, the details of the algorithms and experiments can be found in the respective papers.

The model proposed in Publication III is a combination of Gaussian Process Latent Variable model and truncated-normal likelihood. Furthermore, the model considers a situation where data are from multiple sources. Different from canonical methods such as non-negative matrix factorization (NMF) [95, 30, 129] and Bayesian group factor analysis (GFA) [169, 87], the usage truncated-normal likelihood is novel and effective since it takes care of not only the non-negative observed data but also the latent variables generated from a Gaussian Process.

The model proposed in Publication IV is inspired by the empirical work on player typology in Publication I. The proposed model introduces the factor structure to a topic model to handle the complexity of document-level covariates. In contrast, STM can only handle low-dimensional covariates. The capability of the developed model can contribute to not only game studies but also other fields such as political science.

4.1 Publication III: Multi-source Non-negative Matrix Factorization

In many data analysis situations, the observed data matrix $\mathbf{X}$ does not always contain real-valued numbers. Instead, e.g., in the case of textual data, the elements of the data matrix are non-negative. Let $\mathbf{X}^+$ denote a $N \times D$ term-document matrix where N is the size of vocabulary and D is the number of documents, and each element $x_{v,n} \in \mathbb{R}^+$. In such cases, it may be desirable to find a factorization where the factors are also non-negative. Such a factorization approximation becomes

$$\mathbf{X}^+ \approx \mathbf{Z}^+ \mathbf{W}^{+\top}. \tag{4.1}$$

This approximation leads to a line of research of NMF [95, 30, 129], where the lower-rank matrices $\mathbf{Z}^+ \in \mathbb{R}^{+N \times K}$ and $\mathbf{W}^+ \in \mathbb{R}^{+K \times D}$ are used to approximate the data matrix $\mathbf{X}^+$ containing only non-negative values. In text analytics, the data matrix $\mathbf{X}^+$ is typically a term-document matrix of N terms and D documents containing occurrence counts of N terms over D documents, or numerical statistics for text analytics such as term frequency–inverse document frequency (TF-IDF) values [150]. The matrix $\mathbf{W}^+$ can be interpreted as a topic loading matrix, where each document d, originally represented as a length-N term-count vector $\mathbf{x}_d^+$, is transformed into another length-K representation $\mathbf{w}_d^+$ that contains the topic loadings for K latent topics. On the other hand, $\mathbf{Z}^+$ is the topic content matrix of N terms across the K topics, where each column $\mathbf{z}_{\cdot k}$ is a discrete probability distribution over terms for topic k.

Besides text data analytics, NMF is also widely used in various domains such as bioinformatics [163] and image processing [95]. For simplicity, the notation + will be omitted in the forthcoming equations.

The NMF framework can be further extended to a situation where the input data are a time series of matrices $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(T)}$ for T timestamps. Moreover, the data matrix for each timestamp can also encode group information so that it is a composition of data matrices from different sources, for example, for data from m views, groups, or sources we can write

$$\mathbf{X}^{(t)} = [\mathbf{X}_1^{(t)}, \ldots, \mathbf{X}_m^{(t)}]. \tag{4.2}$$

Therefore, the data matrix for each timestamp is a composition of data matrices from m different sources, therefore, $\mathbf{X}^{(t)}$ can be decomposed with a GFA model

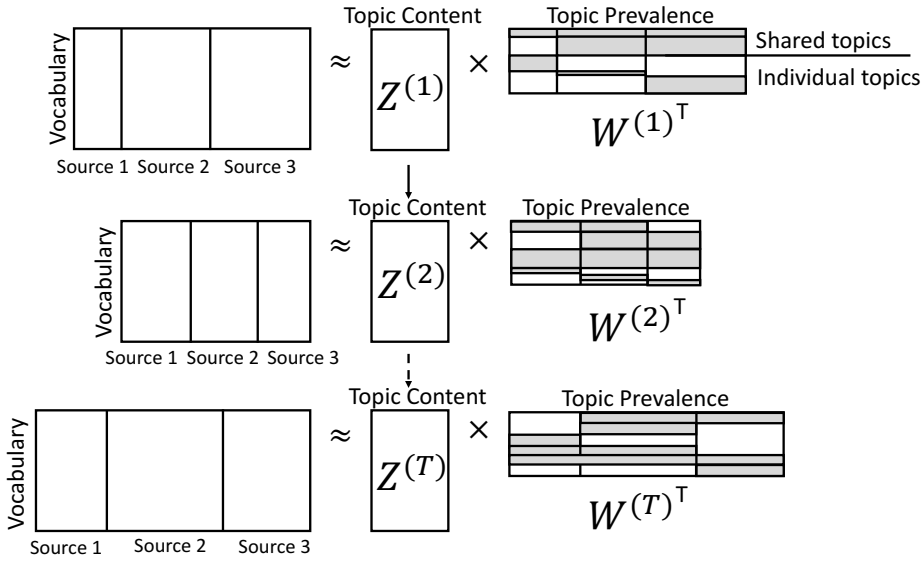

Figure 4.1 Illustration of the DNBGFA model.

[169, 87] as

$$\mathbf{X}^{(t)} \approx \mathbf{Z}^{(t)}\mathbf{W}^{(t)\top} \tag{4.3}$$

where the subpart of the data matrix corresponding to each group m is constructed with a group-wise weighting matrix $\mathbf{W}_m^\top$ and the shared factor matrix $\mathbf{Z}$.

Publication III considers such a situation encompassing analytical challenges including non-negativity, multiple sources, and temporal dynamics. And proposes the probabilistic dynamic non-negative Bayesian group factor (DNBGFA) model. The framework of DNBGFA is shown in Figure 4.1.

In the DNBGFA model, a truncated-Gaussian likelihood is used to model non-negative data, so that

$$p(\mathbf{X}^{(t)}|\mathbf{Z}^{(t)}, \mathbf{W}^{(t)}) = \prod_{n,d} N^+\left(x_{n,d}^{(t)}|\mathbf{z}_n^{(t)\top}\mathbf{w}_d^{(t)}, \sigma^2\right) \tag{4.4}$$

where $\mathbf{w}_d^{(t)}$ denotes the dth column of $\mathbf{W}^{(t)}$ representing the topic prevalence in document d, $\mathbf{z}_n^{(t)}$ denotes the nth row of $\mathbf{Z}^{(t)}$ representing the weight of the nth vocabulary word across the topics The σ^2 controls the noisiness of the observations.

4.1.1 Topic Content Matrix $\mathbf{Z}$

To achieve the non-negativity of the topic content matrix, each element $z_{k,n}^{(t)}$ of $\mathbf{Z}^{(t)}$ is parameterized by a softmax transformation

$$z_{k,n}^{(t)} = \frac{\exp(\eta_{k,n}^{(t)})}{\sum_{n'=1}^{N} \exp(\eta_{k,n'}^{(t)})} \tag{4.5}$$

to ensure that the summation of word proportions of each topic $\sum_{n'=1}^{N} z_{k,n'}^{(t)}$ is equal to 1. And for each term n, the variable $\eta_n = [\eta_{1,n}^{(1)} \dots \eta_{K,n}^{(1)} \dots \eta_{1,n}^{(T)} \dots \eta_{K,n}^{(T)}]^\top$ controls prevalence of the term in the topic content and the dependencies between its elements represent dependencies across sources and time. The Gaussian process latent variable model (GPLVM) [96] is used to model the word dependencies over topics and time. More specifically, for each term n:

$$\left[\eta_{1,n}^{(1)} \eta_{1,n}^{(2)} \dots \eta_{1,n}^{(T)} \dots \eta_{K,n}^{(1)} \dots \eta_{K,n}^{(T)}\right]^\top \sim \mathbf{N}\left(\mathbf{0}, \mathbf{\Sigma}_\eta\right) \tag{4.6}$$

where $\mathbf{\Sigma}_\eta = \boldsymbol{K}_\eta + \varepsilon_\eta \mathbf{I}$ and $\boldsymbol{K}_\eta$ consists of elements $K_{k,l}^{(\eta)}(t_i, t_j) = k_0^{(\eta)}(t_i, t_j)\delta_{k,l} + k_{k,l}^{(\eta)}(t_i, t_j)$. The k, l denote two different topics and $k_0^{(\eta)}(t_i, t_j)\delta_{k,l}$ is a kernel function which governs the within topic consistency over time, $k_{k,l}^{(\eta)}(t_i, t_j)$ governs the topic-topic interaction, and ε_η controls noisiness. $k_0^{(\eta)}$ is an radial basis function (RBF) kernel

$$k_0^{(\eta)}(t_i, t_j) = rbf_{(\xi,\iota)}(t_i, t_j) = \iota^2 \times e^{\frac{-||t_i - t_j||^2}{\xi^2}} \tag{4.7}$$

where ι and ξ are amplitude and width parameters, respectively. And the topic-topic interaction kernel $k_{k,l}^{(\eta)}(t_i, t_j)$ is constructed

$$k_{k,l}^{(\eta)}(t_i, t_j) = e^{-\lambda_\eta |t_i - t_j|} r_k^{(t_i)} r_l^{(t_j)} \tag{4.8}$$

which consists of an exponential time decay term $\lambda_\eta \sim Gamma(a, b)$ and products $r_k^{(t_i)} r_l^{(t_j)}$ where for each topic the vector $\mathbf{r}_k = [r_k^{(t_1)}, \dots, r_k^{(t_T)}]^\top$ is drawn as a realization of a Gaussian process (GP) as

$$\mathbf{r}_k \sim GP(\mathbf{0}, \mathbf{\Sigma}_\mathbf{r}) \ , \quad \mathbf{\Sigma}_\mathbf{r} = \boldsymbol{K}_\mathbf{r} + \varepsilon_r \mathbf{I} \tag{4.9}$$

where $K_{\mathbf{r}}$ consists of elements $K^{(r)}(t_i, t_j) = k_0^{(r)}(t_i, t_j)$ and ε_r controls noisiness. The kernel $k_0^{(r)}$ is also an RBF kernel.

4.1.2 Topic Prevalence $\mathbf{W}$

Similar to the topic content model, to enforce non-negativity, each $w(d)_{m,k}^{(t)}$ of $\mathbf{W}^{(t)}$ is sampled from a truncated normal distribution with mean 0 and a source-wise variance $e^{\alpha_{m,k}^{(t)}}$:

$$w(d)_{m,k}^{(t)} \sim N^{+}(0, e^{\alpha_{m,k}^{(t)}}) \,. \tag{4.10}$$

The source-wise latent variables $\alpha_{m,k}^{(t)}$ which control the sparsity of topic in data sources m and time slices t are again a realization of a GPLVM so that

$$\left[\alpha_{1,k}^{(1)} \dots \alpha_{1,k}^{(T)}, \dots, \alpha_{M,k}^{(1)} \dots \alpha_{M,k}^{(T)}\right]^{\top} \sim \mathbf{N}(\mathbf{0}, \mathbf{\Sigma}_{\alpha}). \tag{4.11}$$

Here, $\mathbf{\Sigma}_{\alpha} = \boldsymbol{K}_{\alpha} + \varepsilon_{\alpha}\mathbf{I}$. The overall noisiness is controled by ε_{α} controls and $\boldsymbol{K}_{\alpha}$ consists of elements

$$K_{m,n}^{(\alpha)}(t_i, t_j) = k_0^{(\alpha)}(t_i, t_j)\delta_{m,n} + k_{m,n}^{(\alpha)}(t_i, t_j) \tag{4.12}$$

$k_0^{(\alpha)}(t_i, t_j)\delta_{m,n}$ is a kernel function that governs the within source consistency of topic prevalence over time and $k_{m,n}^{(\alpha)}(t_i, t_j)$ governs the cross-source interactions, constructed as

$$k_{m,n}^{(\alpha)}(t_i, t_j) = e^{-\lambda_{\alpha}|t_i - t_j|} s_m^{(t_i)} s_n^{(t_j)}. \tag{4.13}$$

The kernel values above are otherwise again composed of products of two terms, an exponential time decay term with decay variable $\lambda_{\alpha} \sim Gamma(c, g)$ and the products $s_m^{(t_i)} s_n^{(t_j)}$ that control the topic-prevalence related correlation of sources across time in a flexible manner. In detail, for each source m the vector $\mathbf{s}_m = [s_m^{(1)}, \dots, s_m^{(T)}]^{\top}$ is generated from an independent GP as

$$\mathbf{s}_m \sim \mathbf{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{s}}) \quad , \quad \mathbf{\Sigma}_{\mathbf{s}} = \boldsymbol{K}_s + \varepsilon_s \mathbf{I} \tag{4.14}$$

where $K_{\mathbf{s}}$ consists of elements

$$K^{(s)}(t_i, t_j) = k_0^{(s)}(t_i, t_j). \tag{4.15}$$

As before, covariances $k_0^{(\alpha)}$ and $k_0^{(s)}$ are obtained by RBF kernel, whose hyperparameters control the time dependence.

4.2 Publication IV: Cross-factor Topic Model

One key challenge in data analysis is analyzing text data and document-level covariates collectively. Publication IV proposes a solution, the Cross-factor Topic Model (CFTM) which can incorporate the document-level covariates when analyzing text data. The key notion is to extract the factorization structure out of the covariates and model its relationship with the topic structure extracted from the text.

4.2.1 Generating Covariates From Latent Factors

The CFTM model assumes that each document d arises from a latent variable which is a nonnegative factor loading vector $\mathbf{\Lambda}_d$ over L factors:

$$\mathbf{\Lambda}_d = [\lambda_{d,1}, \dots \lambda_{d,L}]^\top \sim Dir(\alpha). \tag{4.16}$$

And the covariates are directly generated from an exponential family distribution as

$$x_d^{(p)} | \mathbf{\Lambda}_d, \phi^{(p)} \sim \mathbf{ExpFam}\left(\zeta\left(\mathbf{\Lambda}_d, \phi^{(p)}\right), T\left(x_d^{(p)}\right)\right) \tag{4.17}$$

in which $T\left(x_d^{(p)}\right)$ is the sufficient statistic and the natural parameter ζ is a weighted average of factor-wise parameters $\phi_l^{(p)} \sim N(0, \sigma_\phi^2)$ weighted by the document-specific factor loadings $\mathbf{\Lambda}_d$, so that

$$\zeta\left(\mathbf{\Lambda}_d, \phi^{(p)}\right) = g^{(p)}\left(\sum_{l=1}^{L} \phi_l^{(p)} \lambda_{d,l}\right) \tag{4.18}$$

where g is the link function of the exponential-family model. For example, if a Gaussian with a known variance σ^2 is taken as the distribution, the covariate $\mathbf{x}^{(p)}$ is generated as

$$x_d^{(p)} \sim N(\sum_{l=1}^{L} \phi_l^{(p)} \lambda_{d,l}, \sigma^2). \tag{4.19}$$

4.2.2 Generating Textual Content

The factor loading vector $\mathbf{\Lambda}_d$ further participates in the generation of the textual content of documents. The factors can influence the textual content in two ways, the topic prevalence and topic content.

When it comes to the topic prevalence, $\mathbf{\Lambda}_d$ influences the topic prevalence by participating the generation of the auxiliary variable η_d in each document d so that

$$\eta_{d,1:(K-1)} \sim N(\mathbf{\Gamma}^\top \mathbf{\Lambda}_d, \mathbf{\Sigma}_\eta) \tag{4.20}$$

and the $\eta_{d,K}$ is fixed to 0. The topic prevalence vector for each document $\theta_d = [\theta_{d,1}, \ldots, \theta_{d,K}]$. The coefficient matrix $\mathbf{\Gamma} \in \mathbb{R}^{K \times L}$ controls the interaction between factors and topics at the topic prevalence level. For each topic $k \in \{1, \ldots K-1\}$ a L-length coefficient vector is generated as

$$\mathbf{\Gamma}_k \sim N(0, \sigma_\gamma^2 \mathbf{I}_L) \tag{4.21}$$

On the other hand, when it comes to the topic content, a structure of sparse additive generative models (SAGE) [42] is used to model how γ influences the topic content. The word generation is conditional on the attached latent vector β_d on each document d. The β_d of length V is used to generate the word content of the document. The v:th element of the latent vector is defined as

$$\beta_{d,v} = \kappa_v^{(w)} + \sum_k \theta_{d,k} \kappa_{v,k}^{(t)} + \sum_l \lambda_{d,l} \kappa_{v,l}^{(f)} + \sum_k \sum_l \theta_{d,k} \lambda_{d,l} \kappa_{v,l,k}^{(i)} + \varepsilon_\beta \tag{4.22}$$

where $\varepsilon_\beta \sim N(0, \sigma_\beta^2)$. The $\kappa^{(w)} = \left[\kappa_1^{(w)}, \ldots, \kappa_V^{(w)}\right]^\top$ is a vector of length V controlling the overall word prevalence. The $\kappa_v^{(t)}$ denotes elements of the overall topic content latent matrix $\kappa^{(t)}$ which is a is a $V \times K$ matrix, factor influence $\kappa^{(f)}$ is a $V \times L$ matrix, and $\kappa_{v,l,k}^{(i)}$ denotes elements of $\kappa^{(i)}$, a $V \times L \times K$ array which governs factor-topic interactions on the topic content level, that is, the value of $\kappa_{v,l,k}^{(i)}$ reflects the strength of how much the factor l alters the word probability of v in topic k.

Finally, to generate the observed words in the document, for the nth word in document d, the word $w_n^{(d)}$ is generated as

$$w_n^{(d)} \sim MN\left(softmax\left(\beta_d\right)\right) . \tag{4.23}$$

where MN denotes a multinomial distribution and $softmax\left(\beta_d\right)$ is acting as the overall word distribution of a document.

4.3 Applications

This section demonstrates the application of the developed models wich as focus on analyzing text data from internet. The DNBGFA is applied to jointly anlayze text data from three Finnish text sources online and CFTM is used to analyze review and player data from Steam.

4.3.1 Modeling the Temporal Dynamics of Online Content in Finnish News and Social Media

The DNBGFA model can be used to analyze the temporal dynamics of text data such as Finnish news and social media. In this thesis, it was used to analyze text data from three sources including Helsingin Sanomat (a Finnish newspaper), the Finnish Twitter Census[1], and Suomi24 (a Finnish online forum; text from sections Talous (Economics) and Yhteiskunta (Society) are used). The collected data were separated to 12 time slices (months) from September 2011 to August 2012. For each data source and time slice, the longest 150 documents were collected and analyzed. Stop words and rare terms were removed, text was lemmatized, and the TF-IDF weighted term-document matrices were utilized to train a DNBGFA model.

Partial model output is displayed in Figure 4.2 with a focus on the topic **Media**. Its content evolution over time is displayed in sub-figure (a), and the evolution of topic sparsity in three different sources is shown in sub-figure (b). The results demonstrate a potential trend of shifting from news (with top words read reporter, and paragraph) in earlier time slices to social media (with top words Facebook, source, and computer) in later time slices. In addition, the topic sparsity, reflecting the popularity, has started to rise rapidly from November 2011 in all three text sources.

4.3.2 Exploring Player Experiences Across Factors

The CFTM was employed in this thesis to jointly analyze player reviews and player profile data from Steam. The game reviews as well as the player profile data of a first-

[1] www.finnishtwitter.com

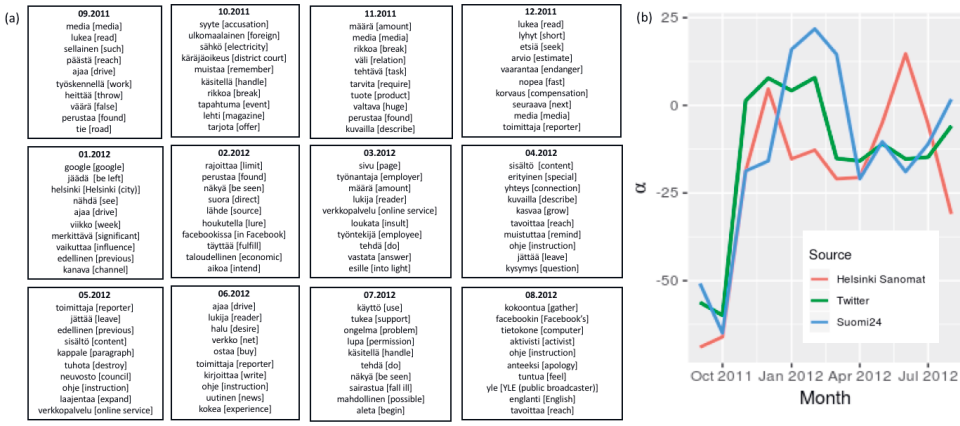

Figure 4.2 Evolution of the topic "Media" over time of topic content and topic sparsity (prevalence)

person shooter game, *Doom Eternal* [13], were used to train a CFTM model. The data was collected from Steam. In total, 22 continuous variables such as the number of achievements and played time, etc., were collected in the player profile data. For text content, numbers, punctuation, and stop words were removed, and the text was lemmatized. Finally, a collection of 1144 reviews with their corresponding player profiles was jointly analyzed.

The extracted topics are presented in Table 4.1, with each topic reflecting a unique aspect of player experiences. Moreover, the CFTM can be used to investigate wording differences of different player factors. As displayed in Figure 4.3, in the topic **Support and Services**, players with a high loading of the factor **Doom-focused Player** tend to use words including 'doom,' 'account,' 'feel,' and 'weapon' in the topic **Feelings and Experiences**. On the other hand, players with a high loading of the factor **Game Collector** prefer using words including 'rip' and 'tear' in both topics **Support and Services** and **Feelings and Experiences**. In general, players identified as **Doom-focused Players** reflect more details of game mechanics in their reviews compared to players identified as **Game Collectors**.

Table 4.1 Extracted topics

Topic	Top Words
Fighting	rep tear dream frankly potato neon success kar smoothly hugo
Support	support response week account anayway offline everyting paste team appove
Visuals and Features	doom dream neon march hdr doot mayhem replayability market kickass
Damage and Survival	damage thing though run challenge player bad lot combat people
Movement and Weapons	dash weapon contain minute maykr grenade teleport switch pad ammo
Feelings and Experiences	really game good recommend software feel level learn start whole

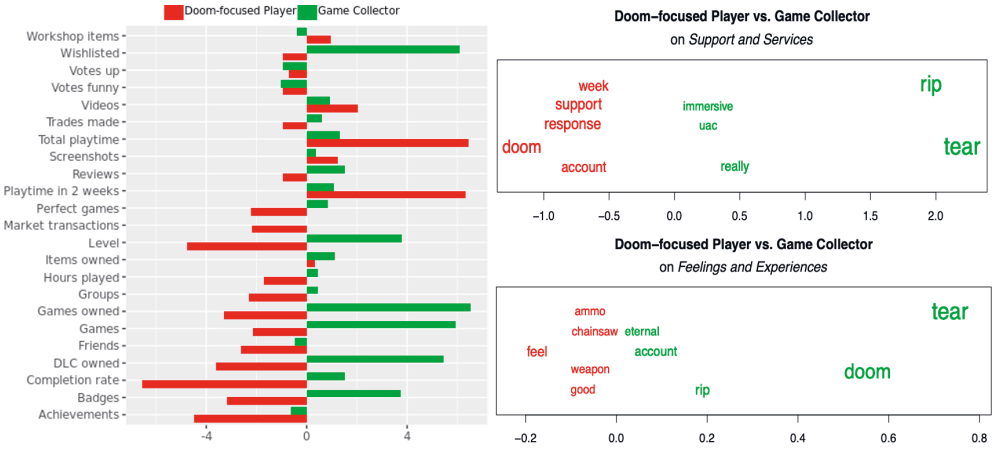

Figure 4.3 CFTM results for Doom Eternal

5 EMBEDDING VECTORS AS REPRESENTATIONS

In this chapter, the research objectives **multiple representation learning** and **learning from heterogeneous data** are addressed from the perspective of developing embedding models. This chapter summarizes the methodology development in Publication V and Publication VI. Publication V focuses on random-walk based graph embedding [135, 61], introducing the notion of multiple representation learning. Publication VI focuses on learning embedding vectors in general, developing a Gaussian Copula-based embedding model to learn latent representations in a heterogeneous data setting.

Publication V presents a key insight into representation learning by allowing each node to carry more than one embedding vector. This offers greater flexibility in learning representations. Additionally, the publication introduces the notion of Bayesian non-parametric, which determines the number of underlying representations based on the complexity of observed data, eliminating the need for additional tasks and simulations.

When analyzing data, heterogeneous data settings are common but have not been properly addressed. Publication VI demonstrates how to incorporate the Gaussian Copula model, a canonical solution for dealing with heterogeneous data. By using this model, the representation learning algorithm can handle inter-relations between data properly and make better use of the information provided by heterogeneous data.

5.1 Publication V: Learning Multiple Representations on a Graph

Publication V adopts the Bayesian non-parametric approach to learn multiple representations for node embedding in a graph. Although prior research has explored the idea of learning multiple representations for nodes in a graph, existing models that consider multiple representations often have limitations such as a fixed number

of representations, or requiring additional tasks or simulations [159, 99, 131, 33, 43]. In contrast, the proposed model in Publication V only relies on the generated random walks, without any additional constraints or requirements.

5.1.1 Random-Walk Based Graph embedding

Let $G = (\mathbb{V}, \mathbb{E})$ denote a graph with a set of nodes/vertices $\mathbb{V}$ and a set of edges and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ indicating the connectivity between nodes. Random-walk based graph embedding [135] is a representation learning technique that learns embedding vectors for each node $v \in \mathbb{V}$ based on a node sequence generated from a graph by performing a random walk. More concretely, a random walk $\mathbf{w} = \{w_1, \ldots, w_L\}$ of length L is a simulated sequence of nodes over the graph G where for each node in the sequence, the next node is chosen at random from its alters or neighbors.

The notion concept of random-walk based graph embedding is that the representation learning is learned from a generated sequence $\mathbf{w}$ is a process of sampling from the graph, therefore, it is able to capture the characteristics of the graph such as complexity and connectivity of nodes. Moreover, since $\mathbf{w}$ is "sampled" from the graph G, it can be modeled by a probabilistic model that generates $\mathbf{w}$. Note that there are approaches [61, 142] employing more sophisticated sampling strategies to generate the node sequence, here the proposed model considers the basic setting of random walk. Similar to the negative samples in a word embeddings setting, which was introduced in Section 2.2.6, in each location, a number of "negative samples" are also sampled for each location $l \in 1 \ldots L$ in the te random walk. That is, if at the location l, $w_l = v$, the corresponding negative samples are sampled from the set of nodes $\mathbb{V} \setminus v$.

After the sequences (including negative samples) are generated, a "language model" is trained to model the co-occurrence of nodes in contexts. Here, the context is defined as a sliding window with a fixed length over the sequence. The node here is treated as a "word" in a language model such as word2vec [116]. The appearance of a node is conditional on its context nodes in the generated sequence.

5.1.2 Bayesian Non-parametric models

Bayesian non-parametric models are a way to offer a high level of flexibility in statistical models where the number of parameters is not fixed but decided by the com-

plexity of observed data. Publication III assumes that each node v can posse more than one underlying embedding vector and introduces a Bayesian non-parametric model to flexibly learn the number of multiple latent embedding vectors from data. After a random walk over the graph is performed, let $n \in 1 \dots N$ denote the index of the location of the generated node sequence, $v \in 1 \dots V$ denote the index the node, and $\rho_{n,v}$ denotes the embedding vector used by the node v at the location n. If the node v selects its s-th embedding vector , it is denoted as

$$\rho_{n,v} = \rho_v^{(s)}. \tag{5.1}$$

The selection is determined by a stochastic process G_v so that

$$\rho_v^{(s)} \sim G_v(G_0, \gamma). \tag{5.2}$$

Here, G_0 is a base distribution and γ is a concentration hyper-parameter. More specifically, an infinite number of possible embedding vectors can be sampled from G_0 where G_v is a draw from it with probabilities $\{\rho_v^{(1)}, \dots, \rho_v^{(s)}, \dots, \rho_v^{(S)}, \dots, \}$ where S is the number of already observed embedding vectors. The number S is not manually specified but learned from data through iterations.

Dirichlet process [48] has been a typical choice for G_v as a Bayesian non-parametric prior. In Dirichlet process, the predictive probability of that the embedding vector $\rho_{n,v}$ is sampled is expressed as

$$P(\rho_{n,v} | \{\rho_{n',v}; n' \in \mathbf{n}_{v,<n}\}) = \begin{cases} \frac{|\mathbf{n}_{v,<n}^{(s)}|}{\sum_{s'} |\mathbf{n}_{v,<n}^{(s')}| - 1 + \gamma} & \rho_{v,n} = \rho_v^{(s)}, \ \forall \rho_v^{(s)} \in \{\rho_v^{(1)} \dots \rho_v^{(S_v)}\} \\ \frac{\gamma}{\sum_{s'} |\mathbf{n}_{v,<n}^{(s')}| - 1 + \gamma} & \rho_{v,n} = \rho_v^{(S_v+1)} \sim G_0 \end{cases} \tag{5.3}$$

which is a Dirichlet distribution which is proportional to the numbers of occurrences of the previously sampled embedding vectors of v at earlier locations $n' < n$, for both positive and negative samples.

The S_v denotes the number of different embedding vectors used for v prior to the location n and $|\mathbf{n}_{v,<n}^{(s)}|$ is the total number of locations before n where the embedding vector $\rho_v^{(s)}$ has been selected, and the hyper-parameter γ governs the generation of a

new embedding vector.

Despite the fact that the Dirichlet process has been the most commonly selected choice in Bayesian nonparametric modeling, it suffers from an issue called "the rich get richer". That is, as shown in the Equation 5.1.2, the generative process it tends to repeat those "popular" previous embedding vectors, therefore, the first few embedding vectors can become overly dominant over iterations, which can limit flexibility in both modeling and inference.

An alternative to Dirichelt process is the uniform process [170]. It was proposed to address the issue of "the rich get richer". Different from the generative process described in Equation 5.1.2, in the uniform process, the parameters are generated with the predictive probability

$$P(\rho_{n,v}|\{\rho_{n',v}; n' \in \mathbf{n}_{v,<n}\}) = \begin{cases} \frac{1}{S_v+\gamma} & \rho_{n,v} = \rho_v^{(s)}, \forall \rho_v^{(s)} \in \{\rho_v^{(1)} \dots \rho_v^{(S_v)}\} \\ \frac{\gamma}{S_v+\gamma} & \rho_{n,v} = \rho_v^{(S_v+1)} \sim G_0 \end{cases} \tag{5.4}$$

where the embedding vector $\rho_{n,v}$ is generated independently from the occurrence frequencies of previous generated values. The generation is only controlled by a concentration hyper-parameter γ. It is worth noting that the uniform process has been neglected by the machine learning research community. Most of the applications still employing Dirichlet processes as their Bayesian non-parametric priors.

5.1.3 Generating Random Walks with Embedding Vectors

Let $\rho_{n,v} \in \mathbb{R}^D$ denote the embedding vector of the node v at the location n of the random walk and $\alpha_v \in \mathbb{R}^D$ denote the context vector of the vertex v. The proposed model in Publication V can be summarized with the generative process shown below:

1. For each node $v \in \mathbb{V}$:
    - Generate the Bayesian non-parametric stochastic process, $G_v \sim NP(G_0, \gamma)$
    - Generate the context vector, $\alpha_v \sim N(0, \sigma_0^2 I)$
2. For each walk $\mathbf{w} = \{w_1, \dots, w_L\} \in \mathcal{W}$

- For location n:
    - Generate embedding vector, $\rho_{n,v} \sim G_v$
    - Compute the natural parameter, $\eta_{n,v} = g\left(\rho_{n,v}^{\top} \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \alpha_{v'}\right)$
    - Sample from the distribution, $x_{n,v} \sim P(\eta_{n,v})$

Where P is an exponential family distribution with the natural parameter η, More formally, if the vertex appears at the location n, the positive likelihood is then defined as

$$p(x_{n,v} = 1) = f(x_{n,v} = 1 | \eta_n\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)) \tag{5.5}$$

where f is the corresponding probability density function of the exponential family distribution. A "negative likelihood" are used to model the situation when a node does not appear at location n. The corresponding likelihood of the non-appearance is

$$p(x_{n,v} = 0) = f(x_{n,v} = 0 | \eta_n\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)) \,. \tag{5.6}$$

In the proposed method in Publication V, an exponential family distribution is used to model the co-occurrence patterns of nodes. If it is a Bernoulli distribution, the natural parameter p_n depending on the context nodes is defined as

$$p_n = \mathcal{S}\left(\rho_{n,v}^{\top} \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \alpha_{v'}\right) \tag{5.7}$$

where $|\mathbf{c}_n|$ is the number of distinct context nodes and $\mathcal{S} = \frac{1}{1+e^{-x}}$ is sigmoid function.

The appearance of the node v at the location n, i.e. whether $x_{n,v} = 1$ or $x_{n,v} = 0$, is thus Bernoulli distributed with parameter so that

$$x_{n,v} \sim Bern(p_n) \,. \tag{5.8}$$

The notion of using Bernoulli likelihood to model the co-appearance of the nodes is inline with the Skip-gram based models [116]. When it comes to modeling the number of occurrences of nodes, Poisson and Gaussian distributions are employed. If a Poisson distribution is chosen, the natural parameter λ_n is defined as

$$\lambda_n = \exp\left(\rho_{n,v}^{\top} \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \alpha_{v'}\right) \tag{5.9}$$

with the exponential function as the link function. And $|\mathbf{c}_n|$ is again the number of distinct nodes in the context and $x_{n,v'}$ denotes the number of occurrences of node v' in the context. The appearance of the node v is generated from a Poisson distribution, so that

$$x_{n,v} \sim Pois(\lambda_n). \tag{5.10}$$

When is comes to a Gaussian distribution, it is similar to the settings for Poisson distribution. The natural parameter is defined as

$$\mu_n = \rho_{n,v}^{\top} \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \alpha_{v'} \tag{5.11}$$

without a specific link function, and the appearance of the node v at the location n is generated as

$$x_{n,v} \sim Norm(\mu_n, \sigma) \tag{5.12}$$

where σ is set as a fixed hyper-parameter. The primary difference of the Poisson and Gaussian setting in contrast to the Bernoulli distribution setting is that they take the number of occurrences of nodes in the context into account. The difference yields different process when constructing the natural parameter and offers a better model flexibility.

5.2 Publication VI: Learning Embedding Vectors from Heterogeneous Data

Publication VI proposes an embedding model for heterogeneous data. Modern domains often involve multiple data types with varied distributions, which brings challenges, especially when modeling the relationships between the data types due to the complexity. Moreover, Naive solutions that ignore the characteristics of the individual data types. Publication VI's model addresses these challenges by considering the varied scales and shapes of the different data types and modeling their relationships.

5.2.1 Gaussian Copula Models

The key notion of the proposed model in Publication VI is to use a Gaussian copula to model the dependencies between variables having arbitrary data types and marginal distributions. A J-dimensional copula $\mathbb{C}$ is a multivariate cumulative distribution function on $[0,1]^J$. Each univariate marginal distribution of $\mathbb{C}$ is uniformly distributed on $[0,1]$. More concretely, given a set of uniform distributed random variables $U_1, \ldots, U_J$, a copula is the joint cumulative distribution

$$\mathbb{C}(u_1, \ldots, u_J) = P(U_1 \leq u_1, \ldots, U_J \leq u_J) \,. \tag{5.13}$$

According to Sklars' theorem [155], let $\mathbf{x}$ denote a random vector of length J, and let $j \in 1 \ldots J$ denote the index the elements (random variables) in $\mathbf{x}$. The joint cumulative distribution of the variables in $\mathbf{x}$ can be modeled by a copula

$$F(x_1, \ldots x_J) = \mathbb{C}\left(F_1(x_1), \ldots, F_J(x_J)\right) \tag{5.14}$$

where F is the joint CDF and $F_j(x) = P(X_j \leq x)$ is the j-th marginal CDF. A copula can be obtained from the right-hand side, since each marginal cumulative distribution function (CDF) value lies within the range $[0,1]$, without regard to the underlying distributions of individual marginal CDFs. When each F_j is continuous, the copula $\mathbb{C}$ is unique. This feature makes the copula a valuable tool to capture the interdependencies between variables, even when they have diverse types and dissimilar marginal CDFs such as $x_1, \ldots, x_J$.

The Gaussian copula is utilized in Publication VI for representation learning tasks. A Gaussian copula is defined with a J-dimensional Gaussian cumulative distribution function (CDF) Φ_J, so that

$$\mathbb{C}(u_1, \ldots u_J) = \Phi_J\left(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_J)|\mathbf{C}\right) \tag{5.15}$$

with a correlation matrix $\mathbf{C}$, and Φ^{-1} refers to the inverse function of the standard univariate Gaussian CDF. The Gaussian copula enables modeling of the joint CDF of observed data as follows:

$$F(x_1, \ldots x_J) = \mathbb{C}(F_1(x_1), \ldots F_J(x_J)) = \Phi_J\left(\Phi^{-1}(F_1(x_1)), \ldots, \Phi^{-1}(F_J(x_J))|\mathbf{C}\right) \,. \tag{5.16}$$

A Gaussian copula can be represented with a latent Gaussian variable model. In which a latent vector $\mathbf{z}$ is generated from a Gaussian distribution

$$z \sim N(0, \Omega) \tag{5.17}$$

with zero mean and a covariance matrix Ω which corresponds to the correlation matrix $\mathbf{C}$ in equation (5.15). Then for each j, the observed data x_j is obtained from

$$x_j = F_j^{-1}\left(\Phi\left(\frac{z_j}{\sqrt{\omega_{jj}}}\right)\right) \tag{5.18}$$

which is the inverse of the univariate marginal F_j^{-1} according to the generated latent variable z_j where ω_{jj} is the j-th diagonal element of Ω.

5.2.2 Generating Heterogenous Data

The proposed Gaussian Copula Embeddings (GCE) is based on the latent representation which is shown in Equations (5.17) and (5.18). More specifically, each item i is corresponding to an latent embedding vector $\rho_i \in \mathbb{R}^{K\times 1}$ which is generated from a multivariate normal prior distribution

$$\rho_i \sim N(0, I) \tag{5.19}$$

and the generated embedding vector ρ_i is involved in the generation of the latent variable vector $\mathbf{z}_n$

$$z_n^{(i)} \sim N(0, \mathbf{I} + R_n R_n{}^{\top}) \Longleftrightarrow z_n^{(i)} \sim N(R_n \rho_i, I) \tag{5.20}$$

where the embedding vectors of the items in the context c_n for all observation variables are used to construct the matrix $R_n \in \mathbb{R}^{K\times J}$.

Following the latent variable representation, all J observations of an item at a location will be generated based on the latent variable z. That is, each observed variable j, the corresponding column $r_{n,j}$ of the matrix R_n is constructed as

$$r_{n,j} = \frac{1}{|c_n|} \sum_{i' \in c_n} \alpha_{i',j} \tag{5.21}$$

where i' are the items in the context of the location n, c_n. The prior of α is again a

multivariate normal distribution

$$\alpha_{i',j} \sim \mathbf{N}(0, \lambda_\alpha^{-1}\mathbf{I}) \tag{5.22}$$

with a diagonal covariance matrix with precision hyer-parameter λ_α which controls the constraints on α. By taking the advantage of the exchangeability in equation (5.20), the generating process can be further re-written as

$$z_n^{(i)} \sim N(\mu_n^{(i)}, I), \text{ where } \mu_n^{(i)} = [\mu_{n,1}^{(i)}, \ldots, \mu_{n,J}^{(i)}] \text{ and } \mu_{n,j}^{(i)} = \rho_i{}^\top \frac{1}{|c_n|} \sum_{i' \in c_n} \alpha_{i',j}\,. \tag{5.23}$$

According to the Gaussian copula, the observed data are then obtained from the latent variables z. Let $x_{n,j}^{(i)}$ denote the jth observed value of the location n from the item i, it is obtained as

$$x_{n,j}^{(i)} = F_j^{-1}\left(\Phi\left(\frac{z_{n,j}^{(i)}}{\sqrt{1 + \sum_{k=1}^{K} r_{n,j,k}^2}}\right)\right) \tag{5.24}$$

where $z_{n,j}^{(i)}$ is the jth element of the latent vector $z_n^{(i)}$ and $r_{n,j,k}$ is the kth dimension of the context representation column $r_{n,j}$, and F_j^{-1} is the inverse CDF of the marginal distribution of variable j.

5.3 Applications

This section demonstrates how to use the developed methods for data analysis in the context of online communities. Data from two popular platforms, Twitch and Reddit, are analyzed for different tasks. Twitch data are used to perform a link prediction task, and data from Reddit are used to visualize the relationships between different online communities.

5.3.1 Predicting Connections bewteen Twitch Streamers

Twitch is a popular live game streaming platform where users, known as streamers, broadcast themselves playing video games and engaging in various activities in real-time. The platform plays an important role in terms of player communities. This section demonstrates how the developed non-parametric graph embedding technique

can enhance the predictive performance of connections between Twitch streamers. The relationship among Twitch streamers can be encoded by a graph where each streamer is a node and edges between nodes are used to represent mutual friendship.

Using the non-parametric graph embedding technique, each streamer, or node in the graph can comprise more than one embedding vectors depending on its interactions with other streamers. A link prediction task was performed on a data set of 7126 nodes and 35324 edges [144]. Specifically, 50% of the edges were first removed randomly into a held-out test set while the remaining training graph was still connected. A logistic regression classifier was trained based on the embedding vectors as features learned from the reduced training graph. The classifier was later used to classify the held-out test-set edges. To leverage the learned multiple representations, when training the classifier, the logistic regression was trained with sample weights, that is, each embedding vector $\rho_v^{(s)}$ and its expected weight.

To validate the effectiveness, the model is compared to other representation learning works including Deepwalk [135], node2vec [61], struc2vec [142], EFGE [29], and Splitter [43]. The area under the curve (AUC) is used to evaluate the binary link classification. The results in Table 5.1 show that the multiple representations learned with the developed model can enhance the performance of link prediction.

5.3.2 Visualizing Reddit Online Communities

Reddit is an online social platform where users contribute text, links, images, and videos to topic-specific "subreddits," which are community forums. In particular, the hyperlinks between subreddits allow users to navigate between different communities, aiding in sharing and accessing information, references, and discussions.

The Gaussian copula embedding model is employed to analyze the Reddit Hyperlink Network dataset [91], which contains 858,488 hyperlinks connecting 55,863 subreddits. For each hyperlink, the data set records the source and destination subreddits, along with the hypertext description, including the number of words, sentiments, and fractions of five distinct character types (i.e., alphabetical, digits, uppercase characters, special characters, and white space). The Gaussian copula embedding model is trained based on pairs of source and destination subreddits. In each hyperlink, the source subreddit is treated as the context for the destination subreddit. The five fractions of different character types, the word count, and the sentiment are considered as observed variables (each learning a context vector α).

Table 5.1 Results for Link Prediction

	Twitch		
	D = 50	D = 100	D = 150
Deepwalk	0.659	0.649	0.672
node2vec	0.681	0.691	0.698
struc2vec	0.830	0.828	0.840
EFGE (Bern)	0.681	0.687	0.707
EFGE (Pois)	0.679	0.708	0.714
EFGE (Norm)	0.791	0.791	0.802
Splitter	0.836	0.823	0.823
dp-emb (Bern)	0.757	0.787	0.782
dp-emb (Pois)	0.656	0.704	0.716
dp-emb (Norm)	0.847	0.845	**0.871**
up-emb (Bern)	0.750	0.788	0.784
up-emb (Pois)	0.658	0.706	0.714
up-emb (Norm)	**0.849**	**0.846**	0.869

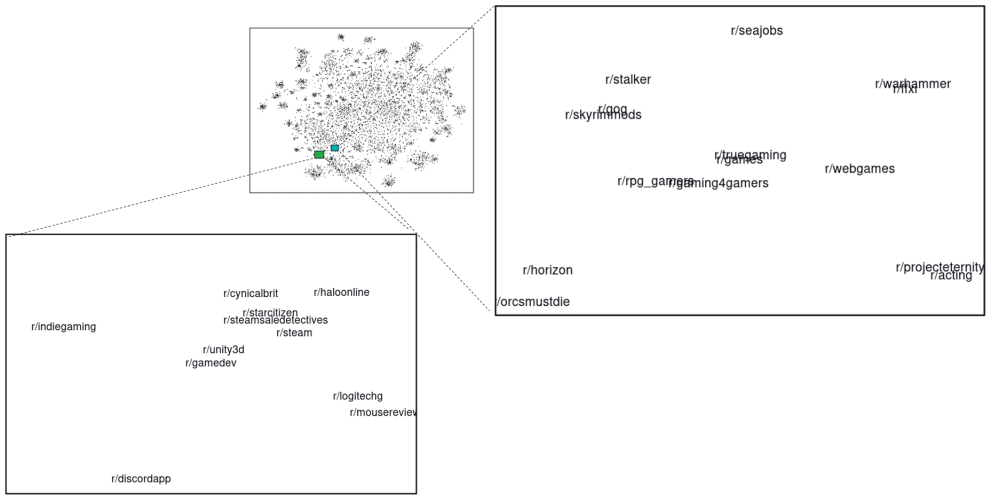

Figure 5.1 t-SNE visualization of learned embedding vectors ρ

Figure 5.1 displays the t-SNE visualization [67, 167] of the learned embedding vectors ρ. The locations reflect the relationships between subreddits. Specifically, the green area contains the subreddits related to game developers (e.g., r/gamedev and r/unity3d), and the light blue-green area contains subreddits that are more player-centric (e.g., r/games and r/webgames), as well as subreddits related to individual games (e.g., r/stalker and r/horizon). This implies that the game and player communities are in general close to each other indicated by the geometric locations of their embedding vectors, but there are still nuances in terms of community formation.

6 DISCUSSION AND CONCLUSION

The thesis is committed to exploring the challenges and potential of applying representation learning techniques in social media data with a focus on game-related data. When capitalizing on such data, one of the major challenges is its complex nature. Rich, diverse, and often unstructured are the features of such data, which makes them difficult to analyze using conventional data analysis methods. Representation learning techniques can help address this challenge by learning meaningful representations from the data, which facilitates effectiveness and interpretation.

The thesis makes a significant contribution to the field of game studies by demonstrating how representation learning techniques can be applied to player-generated data in social media. The empirical analysis presented in Publications I and II provides examples of how representation learning techniques can be used to understand game cultures, such as by identifying player typologies and key themes and topics that are relevant to players. The factor model used in Publication II for the discovery of player typologies offers novel approaches to understanding players. The dichotomous, clear-cut "player types" are substituted with the notion of latent "player factors", the learned representation from players' profile data. Introducing this viewpoint facilitates a more flexible understanding of player behavior and better fits realistic situations. In Publication II, topic modeling is used to investigate the temporal changes in player perceptions in response to game changes. The response to game changes is studied from the "player-centered", and "bottom-up" perspectives. Compared to previous "production-centered" research focusing on the same topic, representation learning has proven to be effective when it comes to investigating this aspect of game culture.

The developed methods in Publications III-VI have drawn upon various machine learning and statistical techniques such as Gaussian processes, Bayesian nonparametrics, and copula models. Each developed method has focused on a specific issue related to representation learning. Moreover, the developed methods have

broader applications beyond game studies, such as in political science. For example, the methodology developed in Publication IV was also used to analyze an open-access dataset, the Finnish election compass to understand the spectrum of politicians' political positions. One of the key aspects of the thesis is its focus on "shallow" or simpler model architecture. Shallow models are computationally efficient and easy to interpret, which makes them highly usable and applicable in a wide range of scenarios. The thesis demonstrates that shallow models can achieve competitive performance in many tasks, despite their simplicity. The thesis argues that the focus on shallow models enables better usability and the potential for further generalization. The DNBGFA was employed to model the temporal dynamics of text from various data sources, including Helsinki Sanomat, Finnish Tweets, and Suomi24. The CFTM was utilized to extract both underlying topics and factors from a dataset of the game *Doom Eternal*, collected from the Steam platform. The non-parametric graph embedding model was employed to analyze a dataset collected from Twitch and demonstrated its ability to improve performance in terms of predicting connections between Twitch streamers. The Gaussian copula embedding model was utilized to understand the relationships between different online communities. Note that, although these methods were primarily developed within the context of game studies, they are anticipated to be applied in other research domains and creating a more extensive impact.

The methods developed in this study have significant implications for future research in both game studies and machine learning. For example, the CFTM model presented in Publication IV has the potential to be applied to various empirical analysis tasks. The notion of multiple representations proposed in Publication V can be utilized to diversify the results of graph-based recommendation systems by introducing the flexibility of allowing multiple representations for each node. Finally, the Gaussian copula-based embedding model introduced in Publication VI can be integrated into, e.g., a layer of a deep learning model, to enhance more inclusive data analysis.

Regarding the research questions, the inquiry denoted as *RQ.1*: "How to distill the crucial information from the dependencies and uncover the perpendicular, or uncorrelated dimensions that reveal the underlying structure of the data?" is answered by the first empirical analysis presented in Chapter 3. The efficacy of the factor model, particularly in the context of player typologies, is demonstrated

to be promising in the extraction of crucial information from mutually dependent data. Furthermore, the extracted factors are meaningful in exploring and explaining player typologies. The *RQ.2*: "How to leverage representation learning techniques to model and understand such evolution of data over time?" is addressed by the second empirical analysis in Chapter 3, where the temporal dynamics of player perception are modeled and analyzed. The DNBGFA developed in Chapter 4 further leverages a Gaussian process latent variable model to uncover the temporal dynamics from non-negative matrix data. *RQ.3*: "How do different underlying structures interact with each other? How to model and interpret such interactions?" is addressed by the model CFTM presented in Chapter 4, as it integrates a factor model and a topic model for cross-structured data analysis. The non-parametric graph embedding model presented in Chapter 5 addresses *RQ.4*: "What is the appropriate approach to introduce diversity to representation learning? How to take advantage of the learned representations?" by introducing Bayesian non-parametrics for multiple representation learning in graph-structured data. Finally, *RQ.5*: "How to effectively analyze and integrate heterogeneous data from various sources to derive meaningful insights? How to interpret and visualize the modeling results?" is addressed by the Gaussian copula embedding model presented in Chapter 5, proposing a Gaussian copula-based framework for learning representations from heterogeneous data.

Despite the contributions made by this thesis, there are still limitations and ongoing challenges that need to be addressed. The contributions (Publication I and II) on empirical analysis are limited to using existing techniques, the future research will focus on applying techniques developed in Publication III to Publication VI to more empirical data analysis tasks. On the other hand, the data types that this thesis investigates are still limited, analyzing and capitalizing on other data types such as images, videos, and audio that are also heavy in games and play are important area of research that requires further investigation.

The emergence of AI-generated content (e.g., GPT [139, 21] and Dalle-2 [140]) poses challenges that need to be addressed, as they can make an impact on the credibility of online data, particularly in social media, which is often used to analyze and represent real human behaviors. Further research is needed to evaluate the impact of machine-generated content and develop appropriate responses [119].

In conclusion, the thesis provides insights into the application of representation learning techniques to analyzing player-generated content in social media. The de-

veloped methods and techniques can be further applied to fields other than game studies that also require data analysis. This thesis has used both empirical analysis and methodology development to demonstrate that in a wide range of scenarios, shallow models can be highly effective and applicable. Also, limitations and ongoing challenges of this research are also highlighted. Nevertheless, the thesis provides a solid foundation for further research in this area.

REFERENCES

[1] Espen J Aarseth. *Cybertext: Perspectives on ergodic literature*. JHU Press, 1997.

[2] Ehsan Adeli et al. "Representation learning with statistical independence to mitigate bias". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2021, pp. 2513–2523.

[3] Carina Alves, Geber Ramalho, and Alexandre Damasceno. "Challenges in requirements engineering for mobile games development: The meantime case study". In: *15th IEEE International Requirements Engineering Conference (RE 2007)*. IEEE. 2007, pp. 275–280.

[4] Apostolos Ampatzoglou and Ioannis Stamelos. "Software engineering research for computer games: A systematic review". In: *Information and Software Technology* 52.9 (2010), pp. 888–901.

[5] Pablo Aragón, Vicenç Gómez, and Andreas Kaltenbrunner. "Visualization tool for collective awareness in a platform of citizen proposals". In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 10. 1. 2016, pp. 756–757.

[6] Mennatallah El-Assady, Rita Sevastjanova, Fabian Sperrle, Daniel Keim, and Christopher Collins. "Progressive learning of topic modeling parameters: A visual analytics framework". In: *IEEE transactions on visualization and computer graphics* 24.1 (2017), pp. 382–391.

[7] Pippin Barr, James Noble, and Robert Biddle. "Video game values: Human–computer interaction and games". In: *Interacting with Computers* 19.2 (2007), pp. 180–195.

[8] Richard Bartle. "Hearts, clubs, diamonds, spades: Players who suit MUDs". In: *Journal of MUD research* 1.1 (1996), p. 19.

[9] Chris Bateman, Rebecca Lowenhaupt, Lennart E Nacke, et al. "Player Typology in Theory and Practice." In: *DiGRA conference*. 2011.

[10] Yoshua Bengio, Aaron Courville, and Pascal Vincent. "Representation learning: A review and new perspectives". In: *IEEE transactions on pattern analysis and machine intelligence* 35.8 (2013), pp. 1798–1828.

[11] Kelly Bergstrom and Nathaniel Poor. "Reddit gaming communities during times of transition". In: *Social Media+ Society* 7.2 (2021), p. 20563051211010167.

[12] Kelly Bergstrom and Nathaniel Poor. "Signaling the Intent to Change Online Communities: A Case From a Reddit Gaming Community". In: *Social Media+ Society* 8.2 (2022), p. 20563051221096817.

[13] Bethesda Softworks. *Doom Eternal*. Maryland, U.S.: [PlayStation 4, Statia, Microsoft Windows, Xbox One, Nintendo Switch, PlayStation 5, Xbox Series X/S] Hello Games, 2020.

[14] Jeremy Blackburn et al. "Cheaters in the steam community gaming social network". In: *arXiv preprint arXiv:1112.4915* (2011).

[15] David Blei and John Lafferty. "Correlated topic models". In: *Advances in neural information processing systems* 18 (2006), p. 147.

[16] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. "Variational inference: A review for statisticians". In: *Journal of the American statistical Association* 112.518 (2017), pp. 859–877.

[17] David M Blei and John D Lafferty. "Dynamic topic models". In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 113–120.

[18] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[19] Danny Plass-Oude Bos et al. "Human-computer interaction for BCI games: Usability and user experience". In: *2010 International Conference on Cyberworlds*. IEEE. 2010, pp. 277–281.

[20] Emily Brown and Paul Cairns. "A grounded investigation of game immersion". In: *CHI'04 extended abstracts on Human factors in computing systems*. 2004, pp. 1297–1300.

[21] Tom Brown et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[22] David Buckingham and Andrew Burn. "Game literacy in theory and practice". In: *Journal of Educational Multimedia and Hypermedia* 16.3 (2007), pp. 323–349.

[23] Denis Bulygin and Ilya Musabirov. "How People Reflect On The Usage Of Cosmetic Virtual Goods: A Structural Topic Modeling Analysis Of R/Dota2 Discussions". In: *Higher School of Economics Research Paper No. WP BRP* 60 (2020).

[24] Margaret Burnett, Curtis Cook, and Gregg Rothermel. "End-user software engineering". In: *Communications of the ACM* 47.9 (2004), pp. 53–58.

[25] Marc Busch et al. "Player type models: Towards empirical validation". In: *Proceedings of the 2016 CHI conference extended abstracts on human factors in computing systems*. 2016, pp. 1835–1841.

[26] Irina Busurkina, Valeria Karpenko, Ekaterina Tulubenskaya, and Denis Bulygin. "Game experience evaluation. A study of game reviews on the steam platform". In: *Digital Transformation and Global Society: 5th International Conference, DTGS 2020, St. Petersburg, Russia, June 17–19, 2020, Revised Selected Papers 5*. Springer. 2020, pp. 117–127.

[27] Marcus Carter, Martin Gibbs, and Mitchell Harrop. "Metagames, paragames and orthogames: A new vocabulary". In: *Proceedings of the international conference on the foundations of digital games*. 2012, pp. 11–17.

[28] Sandro Cavallari, Vincent W Zheng, Hongyun Cai, Kevin Chen-Chuan Chang, and Erik Cambria. "Learning community embedding with community detection and node embedding on graphs". In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 2017, pp. 377–386.

[29] Abdulkadir Celikkanat and Fragkiskos D Malliaros. "Exponential family graph embeddings". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 04. 2020, pp. 3357–3364.

[30] Ali Taylan Cemgil. "Bayesian inference for nonnegative matrix factorisation models". In: *Computational intelligence and neuroscience* 2009 (2009).

[31] Hsiang Chen, Rolf T Wigand, and Michael S Nilan. "Optimal experience of web activities". In: *Computers in human behavior* 15.5 (1999), pp. 585–608.

[32] Ning Chen, Jialiu Lin, Steven CH Hoi, Xiaokui Xiao, and Boshen Zhang. "AR-miner: mining informative reviews for developers from mobile app marketplace". In: *Proceedings of the 36th International Conference on Software Engineering*. ACM. 2014, pp. 767–778.

[33] Yujun Chen, Juhua Pu, Xingwu Liu, and Xiangliang Zhang. "Gaussian mixture embedding of multiple node roles in networks". In: *World Wide Web* 23.2 (2020), pp. 927–950.

[34] Yooncheong Cho, Il Im, Roxanne Hiltz, and Jerry Fjermestad. "An analysis of online customer complaints: implications for web complaint management". In: *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE. 2002, pp. 2308–2317.

[35] Adelina Ciurumelea, Andreas Schaufelbühl, Sebastiano Panichella, and Harald C Gall. "Analyzing reviews and code of mobile apps for better release planning". In: *2017 Imimno2011optimizingEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE. 2017, pp. 91–102.

[36] Steven P Crain, Ke Zhou, Shuang-Hong Yang, and Hongyuan Zha. "Dimensionality reduction and topic modeling: From latent semantic indexing to latent dirichlet allocation and beyond". In: *Mining text data* (2012), pp. 129–161.

[37] David Crookall. *Thirty years of interdisciplinarity*. 2000.

[38] Nicki Skafte Detlefsen, Søren Hauberg, and Wouter Boomsma. "Learning meaningful representations of protein sequences". In: *Nature communications* 13.1 (2022), pp. 1–12.

[39] David O Dowling, Christopher Goetz, and Daniel Lathrop. "One year of# GamerGate: The shared Twitter link as emblem of masculinist gamer identity". In: *Games and Culture* 15.8 (2020), pp. 982–1003.

[40] Jingcheng Du et al. "Gene2vec: distributed representation of genes based on co-expression". In: *BMC genomics* 20.1 (2019), pp. 7–15.

[41] Ben Egliston. "Quantified play: Self-tracking in videogames". In: *Games and Culture* 15.6 (2020), pp. 707–729.

[42] Jacob Eisenstein, Amr Ahmed, and Eric P Xing. "Sparse additive generative models of text". In: *Proceedings of the 28th international conference on machine learning (ICML-11)*. 2011, pp. 1041–1048.

[43] Alessandro Epasto and Bryan Perozzi. "Is a single embedding enough? learning node representations that capture multiple social contexts". In: *The world wide web conference*. 2019, pp. 394–404.

[44] Markku Eskelinen. "The gaming situation". In: *Game studies* 1.1 (2001), p. 68.

[45] Ali Faisal and Mirva Peltoniemi. "Establishing video game genres using data-driven modeling and product databases". In: *Games and Culture* 13.1 (2018), pp. 20–43.

[46] Gerald Farca. "The Emancipated Player." In: *DiGRA/FDG*. 2016.

[47] Justin Farrell. "Corporate funding and ideological polarization about climate change". In: *Proceedings of the National Academy of Sciences* 113.1 (2016), pp. 92–97.

[48] Thomas S Ferguson. "A Bayesian analysis of some nonparametric problems". In: *The annals of statistics* (1973), pp. 209–230.

[49] Tim Finin et al. "The information ecology of social media and online communities". In: *AI Magazine* 29.3 (2008), pp. 77–77.

[50] Gonzalo Frasca. "Ludologists love stories, too: notes from a debate that never took place." In: *DiGRA conference*. 2003, pp. 4–6.

[51] Benjamin Fritz and Stefan Stöckl. "Why do We Play? Towards a Comprehensive Player Typology". In: *Games and Culture* (2022), p. 15554120221094844.

[52] Bin Fu et al. "Why people hate your app: Making sense of user feedback in a mobile app store". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013, pp. 1276–1284.

[53] Alexander R Galloway. *Gaming: Essays on algorithmic culture*. Vol. 18. U of Minnesota Press, 2006.

[54] Enrico Gandolfi. "To watch or to play, it is in the game: The game culture on Twitch. tv among performers, plays and audiences". In: *Journal of Gaming & Virtual Worlds* 8.1 (2016), pp. 63–82.

[55] Ercilia Garcıa-Álvarez, Jordi López-Sintas, and Alexandra Samper-Martınez. "The social network gamer's experience of play: A netnography of restaurant city on facebook". In: *Games and Culture* 12.7-8 (2017), pp. 650–670.

[56] Robert M Geraci and Nat Recine. "Enlightening the galaxy: How players experience political philosophy in Star Wars: The Old Republic". In: *Games and Culture* 9.4 (2014), pp. 255–276.

[57] Meghan Gestos, Jennifer Smith-Merry, and Andrew Campbell. "Representation of women in video games: A systematic review of literature in consideration of adult female wellbeing". In: *Cyberpsychology, Behavior, and Social Networking* 21.9 (2018), pp. 535–541.

[58] László Grad-Gyenge, Attila Kiss, and Peter Filzmoser. "Graph embedding based recommendation techniques on the knowledge graph". In: *Adjunct publication of the 25th conference on user modeling, adaptation and personalization*. 2017, pp. 354–359.

[59] Des Greer and Guenther Ruhe. "Software release planning: an evolutionary and iterative approach". In: *Information and software technology* 46.4 (2004), pp. 243–253.

[60] Lucio Gros, Nicolas Debue, Jonathan Lete, and Cécile Van De Leemput. "Video game addiction and emotional states: possible confusion between pleasure and happiness?" In: *Frontiers in psychology* 10 (2020), p. 2894.

[61] Aditya Grover and Jure Leskovec. "node2vec: Scalable feature learning for networks". In: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 2016, pp. 855–864.

[62] Quanquan Gu, Jie Zhou, and Chris Ding. "Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs". In: *Proceedings of the 2010 SIAM international conference on data mining*. SIAM. 2010, pp. 199–210.

[63] Emitza Guzman and Walid Maalej. "How do users like this feature? a fine grained sentiment analysis of app reviews". In: *2014 IEEE 22nd international requirements engineering conference (RE)*. IEEE. 2014, pp. 153–162.

[64] Juho Hamari and Janne Tuunanen. "Player Types: A Meta-synthesis". In: *Transactions of the Digital Games Research Association* 1.2 (2014).

[65] Timothy R Hannigan et al. "Topic modeling in management research: Rendering new theory from textual data". In: *Academy of Management Annals* 13.2 (2019), pp. 586–632.

[66] Hello Games. *No Man's Sky*. Guildford, UK: [PlayStation 4, Microsoft Windows, Xbox One] Hello Games, 2016.

[67] Geoffrey E Hinton and Sam Roweis. "Stochastic neighbor embedding". In: *Advances in neural information processing systems* 15 (2002).

[68] Jake M Hofman et al. "Integrating explanation and prediction in computational social science". In: *Nature* 595.7866 (2021), pp. 181–188.

[69] CW Hon and IK Hartono. "Analysis and Design of E-Commerce on the Game Information Portal". In: *IOP Conference Series: Earth and Environmental Science*. Vol. 426. 1. IOP Publishing. 2020, p. 012171.

[70] John L Horn. "A rationale and test for the number of factors in factor analysis". In: *Psychometrika* 30 (1965), pp. 179–185.

[71] Johan Huizinga. *Homo ludens: A study of the play-element in culture*. Routledge, 2014.

[72] Lloyd G Humphreys and Richard G Montanelli Jr. "An investigation of the parallel analysis criterion for determining the number of common factors". In: *Multivariate Behavioral Research* 10.2 (1975), pp. 193–205.

[73] Kai Huotari and Juho Hamari. "A definition for gamification: anchoring gamification in the service marketing literature". In: *Electronic Markets* 27.1 (2017), pp. 21–31.

[74] Mohsen Jamali and Martin Ester. "A matrix factorization technique with trust propagation for recommendation in social networks". In: *Proceedings of the fourth ACM conference on Recommender systems*. 2010, pp. 135–142.

[75] Daniel James Joseph. "The discourse of digital dispossession: Paid modifications and community crisis on steam". In: *Games and Culture* 13.7 (2018), pp. 690–707.

[76] Jesper Juul. "A clash between game and narrative". In: *Danish literature* (1999).

[77] Jesper Juul. "Games telling stories". In: *Game studies* 1.1 (2001), p. 45.

[78] Kirsi Pauliina Kallio, Frans Mäyrä, and Kirsikka Kaipainen. "At least nine ways to play: Approaching gamer mentalities". In: *Games and Culture* 6.4 (2011), pp. 327–353.

[79] Christopher M Kanode and Hisham M Haddad. "Software engineering challenges in game development". In: *2009 Sixth International Conference on Information Technology: New Generations*. IEEE. 2009, pp. 260–265.

[80] Jussi Kasurinen, Andrey Maglyas, and Kari Smolander. "Is requirements engineering useless in game development?" In: *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer. 2014, pp. 1–16.

[81] Kevin M Kieffer. "An Introductory Primer on the Appropriate Use of Exploratory and Confirmatory Factor Analysis." In: *Research in the Schools* 6.2 (1999), pp. 75–92.

[82] Danielle K Kilgo et al. "Led it on Reddit: An exploratory study examining opinion leadership on Reddit". In: *First Monday* (2016).

[83] Sangkyun Kim, Kibong Song, Barbara Lockee, and John Burton. "What is gamification in learning and education?" In: *Gamification in learning and education*. Springer, 2018, pp. 25–38.

[84] Diederik P Kingma and Max Welling. "Auto-encoding variational {Bayes}". In: *Int. Conf. on Learning Representations*. 2014.

[85] Graeme Kirkpatrick. *The formation of gaming culture: UK gaming magazines, 1981-1995*. Springer, 2015.

[86] J Matias Kivikangas et al. "A review of the use of psychophysiological methods in game research". In: *journal of gaming & virtual worlds* 3.3 (2011), pp. 181–199.

[87] Arto Klami, Seppo Virtanen, Eemeli Leppäaho, and Samuel Kaski. "Group factor analysis". In: *IEEE transactions on neural networks and learning systems* 26.9 (2014), pp. 2136–2147.

[88] Rune Klevjer and Jan Fredrik Hovden. "The structure of videogame preference". In: *Game Studies* 17.2 (2017), p. 16.

[89] Andrew J Ko et al. "The state of the art in end-user software engineering". In: *ACM Computing Surveys (CSUR)* 43.3 (2011), p. 21.

[90] Philip Kotler and KL Keller. "Marketing management, 12e Prentice-Hall". In: *Upper Saddle River, NJ, USA* (2006).

[91] Srijan Kumar, William L Hamilton, Jure Leskovec, and Dan Jurafsky. "Community interaction and conflict on the web". In: *Proceedings of the 2018 world wide web conference*. 2018, pp. 933–943.

[92] Daria J Kuss. "Internet gaming addiction: current perspectives". In: *Psychology research and behavior management* 6 (2013), p. 125.

[93] Sybille Lammes. "Approaching game-studies: towards a reflexive methodology of games as situated cultures." In: *DiGRA Conference*. 2007.

[94] Filip Lange-Nielsen. "The Power-up Experience: A study of Power-ups in Games and their Effect on Player Experience." In: *DiGRA Conference*. 2011.

[95] Daniel D Lee and H Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization". In: *Nature* 401.6755 (1999), pp. 788–791.

[96] Ping Li and Songcan Chen. "A review on gaussian process latent variable models". In: *CAAI Transactions on Intelligence Technology* 1.4 (2016), pp. 366–376.

[97] Xiaozhou Li, Zheying Zhang, and Kostas Stefanidis. "Mobile App Evolution Analysis Based on User Reviews." In: *The 17th International Conference on Intelligent Software Methodologies, Tools, and Techniques*. 2018, pp. 773–786.

[98] Dayi Lin, Cor-Paul Bezemer, Ying Zou, and Ahmed E Hassan. "An empirical study of game reviews on the Steam platform". In: *Empirical Software Engineering* 24.1 (2019), pp. 170–207.

[99] Ninghao Liu et al. "Is a single vector enough? exploring node polysemy for network embedding". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2019, pp. 932–940.

[100] Peter J Liu et al. "Generating Wikipedia by Summarizing Long Sequences". In: *International Conference on Learning Representations*. 2018.

[101] Qi Liu, Matt J Kusner, and Phil Blunsom. "A survey on contextual embeddings". In: *arXiv preprint arXiv:2003.07278* (2020).

[102] Xin Liu, Tsuyoshi Murata, Kyoung-Sook Kim, Chatchawan Kotarasu, and Chenyi Zhuang. “A general view for network embedding as matrix factorization”. In: *Proceedings of the Twelfth ACM international conference on web search and data mining*. 2019, pp. 375–383.

[103] Christian E Lopez and Conrad S Tucker. “The effects of player type on performance: A gamification case study”. In: *Computers in Human Behavior* 91 (2019), pp. 333–345.

[104] Chien Lu, Oğuz’Oz’ Buruk, Lobna Hassan, Timo Nummenmaa, and Jaakko Peltonen. “" Switch" up your exercise: An empirical analysis of online user discussion of the Ring Fit Adventure exergame”. In: CEUR Workshop Proceedings. 2021.

[105] Chien Lu, Jaakko Peltonen, Timo Nummenmaa, Xiaozhou Li, and Zheying Zhang. “What makes a trophy hunter? An empirical analysis of Reddit discussions”. In: *GamiFIN Conference 2020: Proceedings of the 4th International GamiFIN Conference*. CEUR-WS. 2020.

[106] Baojun Ma, Nan Zhang, Guannan Liu, Liangqiang Li, and Hua Yuan. “Semantic search for public opinions on urban affairs: A probabilistic topic modeling-based approach”. In: *Information Processing & Management* 52.3 (2016), pp. 430–445.

[107] Siyuan Ma and Mikhail Belkin. “Diving into the shallows: a computational perspective on large-scale shallow learning”. In: *Advances in neural information processing systems* 30 (2017).

[108] Adrienne Lynne Massanari. “Participatory culture, community, and play”. In: *Learning from* (2015).

[109] Frans Mäyrä. *An introduction to game studies*. Sage, 2008.

[110] Frans Mäyrä. “The Contextual Game Experience: On the Socio-Cultural Contexts for Meaning in Digital Play.” In: *DiGRA Conference*. Citeseer. 2007.

[111] Frans Mäyrä, Jussi Holopainen, and Mikael Jakobsson. “Research methodology in gaming: An overview”. In: *Simulation & Gaming* 43.3 (2012), pp. 295–299.

[112] Frans Mäyrä, Jan Van Looy, and Thorsten Quandt. “Disciplinary identity of game scholars: An outline”. In: *Digital Games Research Association (DiGRA-2013)*. 2013.

[113] Game Studies Frans Mäyrä. “Getting into the game: doing multidisciplinary game studies”. In: *The video game theory reader 2*. Routledge, 2008, pp. 335–352.

[114] Ben Medler. “Player dossiers: Analyzing gameplay data as a reward”. In: *Game Studies* 11.1 (2011).

[115] Alexey N Medvedev, Renaud Lambiotte, and Jean-Charles Delvenne. “The anatomy of Reddit: An overview of academic research”. In: *Dynamics On and Of Complex Networks III: Machine Learning and Statistical Physics Approaches 10* (2019), pp. 183–204.

[116] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. “Distributed representations of words and phrases and their compositionality”. In: *Advances in neural information processing systems*. 2013, pp. 3111–3119.

[117] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. “Optimizing semantic coherence in topic models”. In: *Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics. 2011, pp. 262–272.

[118] Felix Ming, Fai Wong, Zhenming Liu, and Mung Chiang. “Stock market prediction from WSJ: text mining via sparse matrix factorization”. In: *2014 IEEE International Conference on Data Mining*. IEEE. 2014, pp. 430–439.

[119] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. “DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature”. In: *arXiv preprint arXiv:2301.11305* (2023).

[120] Alexander Modell, Jonathan Larson, Melissa Turcotte, and Anna Bertiger. “A graph embedding approach to user behavior anomaly detection”. In: *2021 IEEE International Conference on Big Data (Big Data)*. IEEE. 2021, pp. 2650–2655.

[121] Donn Morrison and Conor Hayes. "Here, have an upvote: Communication behaviour and karma on Reddit". In: *INFORMATIK 2013–Informatik angepasst an Mensch, Organisation und Umwelt* (2013).

[122] Lennart Nacke and Anders Drachen. "Towards a framework of player experience research". In: *Proceedings of the second international workshop on evaluating player experience in games at FDG*. Vol. 11. 2011.

[123] Lennart Nacke et al. "Playability and player experience research". In: *Proceedings of digra 2009: Breaking new ground: Innovation in games, play, practice and theory*. DiGRA. 2009.

[124] Maleknaz Nayebi, Bram Adams, and Guenther Ruhe. "Release Practices for Mobile Apps–What do Users and Developers Think?" In: *2016 ieee 23rd international conference on software analysis, evolution, and reengineering (saner)*. Vol. 1. IEEE. 2016, pp. 552–562.

[125] Timo Nummenmaa, Annakaisa Kultima, Kati Alha, and Tommi Mikkonen. "Applying Lehman's Laws to Game Evolution". In: *Proceedings of the 2013 International Workshop on Principles of Software Evolution*. IWPSE 2013. Saint Petersburg, Russia: ACM, 2013, pp. 11–17. ISBN: 978-1-4503-2311-6. DOI: 10.1145/2501543.2501546. URL: http://doi.acm.org/10.1145/2501543.2501546.

[126] Mark O'Neill, Elham Vaziripour, Justin Wu, and Daniel Zappala. "Condensing steam: Distilling the diversity of gamer behavior". In: *Proceedings of the 2016 internet measurement conference*. 2016, pp. 81–95.

[127] Edward Orehek and Lauren J Human. "Self-expression on social media: Do tweets present accurate and positive portraits of impulsivity, self-esteem, and attachment style?" In: *Personality and social psychology bulletin* 43.1 (2017), pp. 60–70.

[128] Randy J Pagulayan, Kevin Keeker, Dennis Wixon, Ramon L Romero, and Thomas Fuller. "User-centered design in games". In: *The human-computer interaction handbook*. CRC Press, 2002, pp. 915–938.

[129] John W Paisley, David M Blei, and Michael I Jordan. *Bayesian Nonnegative Matrix Factorization with Stochastic Variational Inference*. 2014.

[130] Fabio Palomba et al. “User reviews matter! tracking crowdsourced reviews to support evolution of successful apps”. In: *2015 IEEE international conference on software maintenance and evolution (ICSME)*. IEEE. 2015, pp. 291–300.

[131] Chanyoung Park et al. “Unsupervised differentiable multi-aspect network embedding”. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020, pp. 1435–1445.

[132] SM Park et al. “Optimization of physical quantities in the autoencoder latent space”. In: *Scientific Reports* 12.1 (2022), p. 9003.

[133] V Paul Pauca, Farial Shahnaz, Michael W Berry, and Robert J Plemmons. “Text mining using non-negative matrix factorizations”. In: *Proceedings of the 2004 SIAM international conference on data mining*. SIAM. 2004, pp. 452–456.

[134] Karl Pearson. “LIII. On lines and planes of closest fit to systems of points in space”. In: *The London, Edinburgh, and Dublin philosophical magazine and journal of science* 2.11 (1901), pp. 559–572.

[135] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. “Deepwalk: Online learning of social representations”. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014, pp. 701–710.

[136] Cody Phillips, Madison Klarkowski, Julian Frommel, Carl Gutwin, and Regan L Mandryk. “Identifying commercial games with therapeutic potential through a content analysis of Steam reviews”. In: *Proceedings of the ACM on Human-Computer Interaction* 5.CHI PLAY (2021), pp. 1–21.

[137] Claudia Plant, Sonja Biedermann, and Christian Böhm. “Data compression as a comprehensive framework for graph drawing and representation learning”. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020, pp. 1212–1222.

[138] Jiezhong Qiu et al. “Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec”. In: *Proceedings of the eleventh ACM international conference on web search and data mining*. 2018, pp. 459–467.

[139] Alec Radford et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9.

[140] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. "Hierarchical text-conditional image generation with clip latents". In: *arXiv preprint arXiv:2204.06125* (2022).

[141] Patricio E Ramırez-Correa, F Javier Rondán-Cataluna, and Jorge Arenas-Gaitán. "A posteriori segmentation of personal profiles of online video games' players". In: *Games and Culture* 15.3 (2020), pp. 227–247.

[142] Leonardo FR Ribeiro, Pedro HP Saverese, and Daniel R Figueiredo. "struc2vec: Learning node representations from structural identity". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 385–394.

[143] Margaret E Roberts, Brandon M Stewart, and Edoardo M Airoldi. "A model of text for experimentation in the social sciences". In: *Journal of the American Statistical Association* 111.515 (2016), pp. 988–1003.

[144] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. *Multi-scale Attributed Node Embedding*. 2019. arXiv: 1909.13021 [cs.LG].

[145] Benedek Rozemberczki and Rik Sarkar. "Twitch gamers: a dataset for evaluating proximity preserving and structural role-based node embeddings". In: *arXiv preprint arXiv:2101.03091* (2021).

[146] Maja Rudolph, Francisco Ruiz, Susan Athey, and David Blei. "Structured embedding models for grouped data". In: *Advances in neural information processing systems* 30 (2017).

[147] Maja Rudolph, Francisco Ruiz, Stephan Mandt, and David Blei. "Exponential family embeddings". In: *Advances in Neural Information Processing Systems*. 2016, pp. 478–486.

[148] Günther Ruhe and Moshood Omolade Saliu. "The science and practice of software release planning". In: *IEEE Software* 2005 (2005), pp. 1–10.

[149] Omolade Saliu and Guenther Ruhe. "Supporting software release planning decisions for evolving systems". In: *29th Annual IEEE/NASA Software Engineering Workshop*. IEEE. 2005, pp. 14–26.

[150] Gerard Salton and Christopher Buckley. "Term-weighting approaches in automatic text retrieval". In: *Information processing & management* 24.5 (1988), pp. 513–523.

[151] Simone Scalabrino, Gabriele Bavota, Barbara Russo, Massimiliano Di Penta, and Rocco Oliveto. “Listening to the crowd for the release planning of mobile apps”. In: *IEEE Transactions on Software Engineering* 45.1 (2017), pp. 68–86.

[152] Adrienne Shaw. “What is video game culture? Cultural studies and game studies”. In: *Games and culture* 5.4 (2010), pp. 403–424.

[153] Yexuan Shi et al. “Federated topic discovery: A semantic consistent approach”. In: *IEEE Intelligent Systems* 36.5 (2020), pp. 96–103.

[154] Rafet Sifa, Anders Drachen, and Christian Bauckhage. “Profiling in games: Understanding behavior from telemetry”. In: *Social interactions in virtual worlds: An interdisciplinary perspective* (2018), pp. 337–374.

[155] M Sklar. “Fonctions de repartition an dimensions et leurs marges”. In: *Publ. inst. statist. univ. Paris* 8 (1959), pp. 229–231.

[156] Charles Spearman. “" General Intelligence" Objectively Determined and Measured.” In: (1961).

[157] Constance A Steinkuehler. “Why game (culture) studies now?” In: *Games and culture* 1.1 (2006), pp. 97–102.

[158] Brandon C Strubberg, Timothy J Elliott, Erin P Pumroy, and Angela E Shaffer. “Measuring Fun: A Case Study in Adapting to the Evolving Metrics of Player Experience”. In: *Loading: The Journal of the Canadian Game Studies Association* 13.21 (2020), pp. 1–19.

[159] Fan-Yun Sun, Meng Qu, Jordan Hoffmann, Chin-Wei Huang, and Jian Tang. “vgraph: A generative model for joint community detection and node representation learning”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[160] Mikael Svahnberg et al. “A systematic review on strategic release planning models”. In: *Information and software technology* 52.3 (2010), pp. 237–248.

[161] Jan Švelch. “Normalizing player surveillance through video game infographics”. In: *New Media & Society* (2022), p. 14614448221097889.

[162] Stefano Tardini and Lorenzo Cantoni. “A semiotic approach to online communities: Belonging, interest and identity in websites’ and videogames’ communities”. In: *Proceedings of the IADIS International Conference e-Society*. 2005, pp. 371–8.

[163] Leo Taslaman and Björn Nilsson. "A framework for regularized non-negative matrix factorization, with application to the analysis of gene expression data". In: *PloS one* 7.11 (2012), e46331.

[164] Kathryn Thompson. ""No One Cares, Apostolate" What Social Cheating Reveals". In: *Games and culture* 9.6 (2014), pp. 491–502.

[165] Michael E Tipping and Christopher M Bishop. "Probabilistic principal component analysis". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61.3 (1999), pp. 611–622.

[166] Jukka Vahlo and Juho Hamari. "Five-factor inventory of intrinsic motivations to gameplay (IMG)". In: *Proceedings of the 52nd Hawaii International Conference on System Sciences, Hawaii, USA, 2019*. HICSS. 2019.

[167] Laurens Van der Maaten and Geoffrey Hinton. "Visualizing data using t-SNE." In: *Journal of machine learning research* 9.11 (2008).

[168] Lorenzo Villarroel, Gabriele Bavota, Barbara Russo, Rocco Oliveto, and Massimiliano Di Penta. "Release planning of mobile apps based on user reviews". In: *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*. IEEE. 2016, pp. 14–24.

[169] Seppo Virtanen, Arto Klami, Suleiman Khan, and Samuel Kaski. "Bayesian group factor analysis". In: *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 1269–1277.

[170] Hanna Wallach, Shane Jensen, Lee Dicker, and Katherine Heller. "An alternative prior process for nonparametric Bayesian clustering". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 892–899.

[171] Isaac Waller and Ashton Anderson. "Quantifying social organization and political polarization in online platforms". In: *Nature* 600.7888 (2021), pp. 264–268.

[172] Binghui Wang, Jiayi Guo, Ang Li, Yiran Chen, and Hai Li. "Privacy-preserving representation learning on graphs: A mutual information perspective". In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2021, pp. 1667–1676.

[173] Karl Werder et al. “Data-driven, data-informed, data-augmented: How ubisoft’s ghost recon wildlands live unit uses data for continuous product innovation”. In: *California management review* 62.3 (2020), pp. 86–102.

[174] Stefan Werning. “Disrupting video game distribution”. In: *Nordic Journal of Media Studies* 1.1 (2019), pp. 103–124.

[175] Josef Wiemeyer, Lennart Nacke, Christiane Moser, and Florian ‘Floyd’Mueller. “Player experience”. In: *Serious games: Foundations, concepts and practice* (2016), pp. 243–271.

[176] Dmitri Williams. “Bridging the methodological divide in game research”. In: *Simulation & Gaming* 36.4 (2005), pp. 447–463.

[177] KT Wong. “The Data-Driven Myth and the Deceptive Futurity of “the World’s Fastest Growing Games Region”: Selling the Southeast Asian Games Market via Game Analytics”. In: *Games and Culture* (2022), p. 15554120221077731.

[178] Yingcai Wu et al. “OpinionSeer: interactive visualization of hotel customer feedback”. In: *IEEE transactions on visualization and computer graphics* 16.6 (2010), pp. 1109–1118.

[179] Tim Wulf, Frank M Schneider, and Stefan Beckert. “Watching players: An exploration of media enjoyment on Twitch”. In: *Games and culture* 15.3 (2020), pp. 328–346.

[180] Chonghuan Xu. “A novel recommendation method based on social network using matrix factorization technique”. In: *Information processing & management* 54.3 (2018), pp. 463–474.

[181] Weidi Xu, Haoze Sun, Chao Deng, and Ying Tan. “Variational autoencoder for semi-supervised text classification”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017.

[182] Zhiheng Xu, Long Ru, Liang Xiang, and Qing Yang. “Discovering user interest on twitter with a modified author-topic model”. In: *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*. Vol. 1. IEEE. 2011, pp. 422–429.

[183] Lina Yao, Quan Z Sheng, Anne HH Ngu, Helen Ashman, and Xue Li. "Exploring recommendations in internet of things". In: *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 2014, pp. 855–858.

[184] Nick Yee. "Motivations for play in online games". In: *CyberPsychology & behavior* 9.6 (2006), pp. 772–775.

[185] Nick Yee, Nicolas Ducheneaut, and Les Nelson. "Online gaming motivations scale: development and validation". In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2012, pp. 2803–2806.

[186] Yang Yu, Ba-Hung Nguyen, Fangyu Yu, and Van-Nam Huynh. "Esports Game Updates and Player Perception: Data Analysis of PUBG Steam Reviews". In: *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*. IEEE. 2021, pp. 1–6.

[187] José P Zagal, Amanda Ladd, and Terris Johnson. "Characterizing and understanding game reviews". In: *Proceedings of the 4th international Conference on Foundations of Digital Games*. 2009, pp. 215–222.

[188] José P Zagal, Noriko Tomuro, and Andriy Shepitsen. "Natural language processing in game studies research: An overview". In: *Simulation & Gaming* 43.3 (2012), pp. 356–373.

[189] Xiang Zhang, Hans-Frederick Brown, and Anil Shankar. "Data-driven personas: Constructing archetypal users with clickstreams and user telemetry". In: *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016, pp. 5350–5359.

[190] Zhao Zhang et al. "A survey on concept factorization: From shallow to deep representation learning". In: *Information Processing & Management* 58.3 (2021), p. 102534.

[191] Tianqing Zhu, Gang Li, Wanlei Zhou, Ping Xiong, and Cao Yuan. "Privacy-preserving topic model for tagging recommender systems". In: *Knowledge and information systems* 46 (2016), pp. 33–58.

PUBLICATIONS

PUBLICATION
I

A statistical analysis of Steam user profiles towards personalized gamification

Xiaozhou Li, Chien Lu, Jaakko Peltonen, and Zheying Zhang

In: *Proceedings of the 3rd International GamiFIN Conference, Levi, Finland, April 8-10, 2019.* Ed. by Jonna Koivisto and Juho Hamari. CEUR-WS.org, 2019, pp. 217–228

A statistical analysis of Steam user profiles towards personalized gamification

Xiaozhou Li, Chien Lu, Jaakko Peltonen, and Zheying Zhang

Tampere University
Kalevantie 4, 33100, Tampere, Finland
{xiaozhou.li, chien.lu, jaakko.peltonen, zheying.zhang}@tuni.fi

Abstract. Gamification is widely used as motivational design towards enhancing the engagement and performance of its users. Many commonly adopted game design elements have been verified to be effective in various domains. However, the designs of such elements in the majority of the target systems are similar. Due to inevitable differences between users, gamification systems can perform more effectively when users are provided with differently and personally designed features according to their preferences. Many studies have suggested such requirements towards personalizing gamified systems based on the users' preferences, with categorizing gamification users and identifying their preferences as the initial step. This study proposes a preliminary analysis of the factors that categorize user preference in a game community, based on the user profiles data of the Steam platform. It shall not only facilitate understanding of players' preferences in a game community but also lay the groundwork for the potential personalized gamification design.

Keywords: Gamification · Exploratory Factor Analysis · Steam · User Profile · Preference · Personalized Gamification.

1 Introduction

Gamification, commonly defined as the use of game design elements for non-game contexts [12], has been widely adopted as motivational design to support users motivation enhancement and performance improvement. Many game design elements, e.g., badges/achievements, points, leaderboard, progress, story, etc., have been adopted in various service domains and proven effective in many studies [14]. However, the majority of the gamification systems provide very limited alteration towards different users but adopt the one-size-for-all design approach instead [32]. Such rigid gameful designs are to a certain extent ineffective in persuading the users into positive behaviors. Many studies have shown that different users are likely to be motivated by different game elements and persuasive strategies [31, 32, 40]. Therefore, it is critical to understand different users' preferences when providing them the personalized gameful experiences.

The studies on the users' types and preferences regarding gamification systems are based on the similar studies on game players. A seminal study on the player types for multi-user dungeon (MUD) games is Bartle's player typology [2]. Meanwhile, a number of studies also contribute to extending the user typology framework by focusing

on psychographic and behavioral aspects [15]. Even though the direct connection is not addressed, such studies on player typology do facilitate the understanding of users preference of play style and their motivations of playing [15]. On the other hand, a gamification-specific user typology framework is developed by Marczewski [26], who proposes six gamification user types based on intrinsic or extrinsic motivational affordances [36] and their different degrees for the users. Furthermore, based on this particular framework, a 24-item survey response scale is presented to score users' preferences regarding the six different types of motivation toward a gameful system, which can therefore identify a users type and describe his/her preferences [42].

Despite the uniform well-defined player types and gamification user types, such a 'clear-cut' categorization approach can be questioned as a player may not belong to a certain type strictly [15, 21]. In addition, limitations of using survey data towards such categorization have also been recognized [42]. In this study, we focus on users of the Steam platform and their community-related behaviors presented on their profile pages. The users' Steam profiles provide various information, including the games they have, the game achievements, item trading, friends, groups, reviews, screenshots, profile customization options, and so on. The objective nature and large volume of such data shall has the potential to yield enhanced characterizations of users and their diferences. Herein, based on factor analysis of large user profile data, we identify the factors that characterize the differences between Steam users. Instead of a strict categorization of players, the study aims to answer what are the factors that distinguish Steam users from one another and determine their preferences, as well as how such distinguishing factors can be applied to facilitate personalized gamification design.

The paper is organized as follows. Section 2 introduces previous studies on game players and gamification user typologies and on analysis of the Steam platform and user data. Section 3 introduces our data collection and analysis methods, Sections 4 and 5 present results and discussion Section 6 concludes.

2 Related Work

2.1 Player Types and Gamification User Types

The aim of segmentation in marketing is to identify different customer groups so that they are served with products and services that match their unique needs. Studies on player types also serve this purpose. The majority of the prevailingly cited studies focus on the player segmentation in terms of the behavioral and psychographic attributes instead of geographic or demographic ones [15]; our focus is similar, since our Steam profiles did not contain demographic/geographic attributes and we focused on the available profile information reflecting player behavior. When available, our modeling principle could accommodate demographic/geographic attributes as covariates.

Bartle's seminal player typology — Achiever, Explorer, Socializer and Killer — is based on the things people enjoy about MUD in either an action or interaction dimension towards either players or the game world [2]. It is also criticized for being dichotomous and too simplifying, as well as focusing on only one game genre instead of a broad range [3, 15, 42]. Extending Bartle's typology model, many studies have proposed similar typology models for online game players with specialized focuses [43, 45]. Many

other studies present different ways of categorizing players based on their various motivation and behaviors when not fixating on online games [21, 39]. Such player typology models provide ways to detect the difference in players and their preference regarding motivations and behaviors in general. On the other hand, many studies also focus more specifically on players' preferences regarding game design elements [11, 19].

The studies on gamification user types also adapt the results from the player typology studies. Such studies are mostly supported by the research on behavior motivations and personalities [29,36]. Regarding the user typology in the gamification domain, Marczewskis gamification user type model is the most cited work [26]. Motivated by the intrinsic and extrinsic motivational factors of the users, which is defined by the Self Determination Theory (SDT) [35], Marczewski categorizes the users of gamification services into six types, including socializers, achievers, philanthropists, free spirits, players, and disrupters. Other studies also attempt to provide adapted typology frameworks regarding specific domains [1,44]. Meanwhile, adapting Marczewski's gamification user types model, Tondello et al. present and validate a standard scale to determine users' preference towards gamification systems regarding different motivation types [42]. Based on that, their subsequent works contribute to suggesting gameful design elements regarding user preferences, personalizing persuasive strategies, and creating a recommender system model for personalized gamification [32, 40, 41]. However, mentioned as their limitation, the data are self-reporting and subject heavily to participants' personal understanding of survey statements and preferences towards diverse game elements. Thus, relevant objective data with a larger sample volume can address such limitation and can also yield alternative results.

2.2 The Steam Platform and Users

Steam, a popular digital game distribution platforms, has drawn attention from the academia. Becker et al. analyze the role of games and groups in the Steam community and present the evolution of its network over time [5]. O'Neill et al. also investigate the Steam community but focus on the gamers' behaviors, in terms of their social connectivity, playtime, game ownership, genre affinity, and monetary expenditure [30], whereas Blackburn et al. focus more specifically on the cheating behavior [7]. Many other studies also investigate the various perspectives of players' behaviors on the Steam platform. For example, Sifa et al. investigate the players' engagement and cross-game behavior by analyzing their different playtime frequency distributions [37,38]. Baumann et al. focus on "hardcore" gamers' behavioral categories based on their Steam profiles [4]. Lim and Harrell examine players' social identity and the relation between their profile maintaining behaviors and their social network size [22]. Meanwhile, other scholars also study the other perspectives of Steam, such as, recommender systems for its content [6], early access mechanism [24], game updating strategies [23], game reviews [25], and so on. However, research on characterizing players based on their Steam profile data towards analyzing players' preference to different game design elements is still limited.

3 Method

3.1 Data Collection

A web crawler based on the Beautiful Soup Python module was created to collect data from public user profiles. The data collection proceeded in a "snowball" manner. The crawler started from one user's Steam profile URL which was selected at random from the top 10 Steam user leaderboard, and crawled the list of the user's friends profile URL. Iteratively, the list of users was grown via crawling the friends of each of the existing users on the list and appending the results to the end of the list. Although guaranteeing an unbiased sample from such a huge base is difficult and our gathered dataset is necessarily small, it can still achieve a good representativity. Duplicated profile URLs, as well as private ones from which no valid data can be obtained, were eliminated. To reduce crawling time while achieving reasonable coverage, only profile URLs were crawled, and from the initial data pool of 2561387 unique user profile URLs, we collected the profile information on a random subset of the URLs which includes 60267 users. The crawled features include *Levels*, *Showcases*, *Badges*, *Number of Games*, *Screenshots*, *Workshop Items*, *Videos*, *Reviews*, *Guides*, *Artworks*, *Groups*, *Friends*, *Items Owned*, *Trades Made*, *Market Transactions*, *Achievements*, *Perfect Games*, *Game Completion Rate*, and four binary *profile customization* related variables: *Avatar*, *Status*, *Background*, and *Favorite Badge* customization (customized or not). To summarize the binary variables per user, we define an aggregate value called *Profile Customization* whose value is the percent of 'customized' values: for example, if a particular user customized three of the four items mentioned above, his/her *Profile Customization* score will be assigned as 0.75. In addition, each user's active time span was also collected based on the time when the user last logged off and the time when the user created the account, using the SteamAPI. To take the user activity into account, we further computed the duration the profile had existed using the above-mentioned information and utilized it to normalize the profile variables, by simply dividing each variable by the profile duration.

3.2 Exploratory Factor Analysis

To uncover the underlying structures of the Steam user profiles, an exploratory factor analysis (EFA, [13]) is conducted. It enables us to reduce the complexity of the data, explain the observations with a smaller set of latent factors and discover the relations between variables. Unlike clustering which discovers groups of players, EFA discovers underlying axes characterizing players and their differences. In game culture studies, EFA has been widely used especially in studies related to user/player types and user motivations (e.g. [42, 43]). Extracted EFA factors can also be a basis for analysis such as clustering (player segmentation) or prediction in follow-up work; we focus on discovering underlying axes of variation in Steam user profiles through EFA and their applications in gamification.

One common issue in EFA is how to decide the number of factors. In this paper, the parallel analysis (PA) introduced by Horn [18] is adopted to solve the problem. It has been widely used and has given good results in recent research works (e.g. [33,

34]). Several comparative studies (e.g. [8, 46]) have shown that it is an effective way to determine the number of factors.

Table 1. Result of Parallel Analysis

Factor	Observed Eigenvalue	Simulated Eigenvalue
1	3.104	1.031
2	2.744	1.025
3	1.650	1.021
4	1.382	1.018
5	1.167	1.015
6	1.130	1.011
7	1.073	1.008
8	1.027	1.006
9	0.916	1.003

In PA, the Monte Carlo simulation technique is employed to simulate random samples consisting of uncorrelated variables that *parallel* the number of samples and variables in the observed data. From each such simulation, eigenvalues of the correlation matrix of the simulated data are extracted, and the eigenvalues are, as suggested in the original paper [18], averaged across several simulations. The eigenvalues extracted from the correlation matrix of the observed data, ordered by magnitude, are then compared to the average simulated eigenvalues, also ordered by magnitude. The decision criteria is that the factors with observed eigenvalues higher than the corresponding simulated eigenvalues are considered significant. Hereby, we conduct the parallel analysis task with 5000 simulations to determine the number of factors.

To simplify interpretation of the factor analysis result, the *varimax* rotation technique [20] which maximizes the variance of the each factor loading is employed. Results with an alternative rotation approach *promax* [17] were similar.

4 Result

4.1 Factor Analysis

The result of the parallel analysis is shown in Table 1. Based on the mentioned criteria, the turning point can be found easily by examining the differences between observed eigenvalues and simulated eigenvalues. Since the simulated eigenvalue becomes greater than the observed eigenvalue in the 9th factor (1.003 and 0.916 respectively), the first 8 factors are retained. The corresponding factor loadings can be found in Table 2. A cross-loading of the variable Profile.Customization was found on Factor 1 and 7, we further computed the Cronbach's alpha [9] on those two factors to evaluate their internal consistency and the values are found acceptable (0.87 and 0.71 respectively).

4.2 Factors Interpretation

Based on the result of EFA, we interpret each of the eight factors and summarize each of the unique preference attributes of Steam users.

Table 2. Loadings of the Extracted Factors

Variable	Factor 1	2	3	4	5	6	7	8
Level	**0.641**	-0.005	0.004	-0.002	0.008	-0.013	-0.263	0.002
Showcases	0.026	0.107	0.065	**0.828**	0.162	0.180	0.028	0.067
Badges	**0.954**	0.033	0.004	0.010	0.006	0.043	0.016	0.004
Games	0.019	**0.511**	0.020	0.016	0.108	0.365	0.030	0.088
Screenshots	-0.000	0.118	0.332	0.046	0.344	0.039	0.022	**0.490**
Workshop.Items	0.007	-0.045	0.042	0.127	**0.789**	-0.027	0.003	-0.082
Videos	0.002	-0.030	-0.066	0.046	-0.074	-0.022	-0.003	**0.901**
Reviews	0.002	0.232	0.039	0.044	**0.769**	0.039	0.018	0.113
Guides	0.002	0.024	**0.879**	-0.031	-0.090	-0.003	-0.001	-0.002
Artwork	0.004	-0.010	**0.836**	0.101	0.192	0.006	0.018	0.030
Groups	0.078	0.017	0.020	0.031	0.026	0.008	**0.951**	0.009
Friends	**0.947**	0.002	0.004	0.043	0.007	0.014	0.202	0.001
Items.Owned	0.004	0.048	0.005	0.049	-0.004	**0.733**	0.006	-0.022
Trades.Made	-0.003	-0.142	-0.002	0.281	-0.063	**0.551**	0.003	-0.061
Market.Transactions	0.017	0.116	0.001	-0.063	0.044	**0.645**	-0.007	0.049
Achievements	0.005	**0.865**	0.014	0.125	0.014	-0.010	-0.001	-0.011
Perfect.Games	0.003	**0.847**	0.006	0.210	0.105	-0.045	-0.002	-0.017
Game.Completion.Rate	0.008	0.274	0.013	**0.852**	0.054	-0.004	0.003	0.021
Profile.Customization	**0.808**	-0.007	-0.008	-0.019	-0.015	-0.016	**0.553**	-0.007

Factor 1: Elite (Level, Badge, Friends, and Profile Customization) Factor 1 indicates the users' tendency to become the elite of the Steam community. The *elite* users focus on their social comparison advantages over the others by enhancing their quantifiable social scores, such as, levels, badges, and friends numbers. According to Steam's unique mechanism, the users can upgrade their levels and earn more badges without the requirements of exerting more effort in actual gameplay. Therefore, the elite users tend to value their social achievement more than experiences in gameplay. In addition, they also prefer profile customization in order to present their unique social identity.

Factor 2: Achiever (Games, Achievement, and Perfect Games) Users' tendency in Factor 2 indicates their preference towards mastering the games. They focus on completing games thoroughly and obtaining as many in-game achievements as possible. They also tend to enlarge their game collection whenever possible. Compared to the *elite* users, the *achiever* users prefer to put their effort in games and less in social.

Factor 3: Provider (Guides and Artworks) Users with high attribute in Factor 3 love to provide facilitation to the others with gameplay guides and self-created unique game-related arts. Different from *elite* and *achiever* users who focus on their social presence or achievement, the *provider* users tend to be more altruistic and care about other users and their game playing.

Factor 4: Completer (Showcases and Game Completion Rate) Similar to the *achiever* users, the *completer* users also focus on gameplay but less on achievements. They prefer to finish the games that they start but have less intention of pursuing the full achievement by investing extra amount of hours. Meanwhile, they like to show their possessions, e.g., showcases, as much as possible, but put less effort on organizing compared with the *elite* users.

Factor 5: Improver (Workshop Items and Reviews) Users with high value on Factor 5 focus on game improvement. They make efforts to add unique experiences to games via workshop items and reviews. These encourage developers to improve the games and publish better games in the future. Similar to *provider* users, they are also altruistic but focus more on game quality.

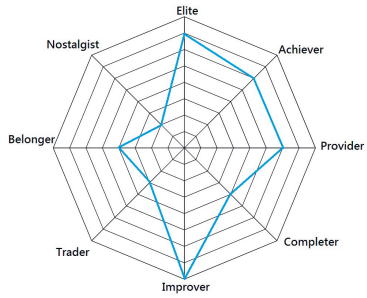

Fig. 1. An Example of User Preference Attributes Radar Chart

Factor 6: Trader (Item Owned, Trades Made, and Market Transaction) The *trader* users do not pay much attention to either games or social, but to buying and selling game related virtual items instead. According to Steam's mechanism, users neither have to own or play games to obtain items nor have to become friends with others or join groups to make trades. Thus, *trader* users tend to make the community a business playground, buying low and selling high.

Factor 7: Belonger (Groups and Profile Customization) Similar to the *elite* users, the *belonger* users also tend to focus more on social interaction than gameplay, when the difference is that the belonger users prefer the feeling of relatedness and belonging, rather than social comparison. Belonging to social groups is always their first priority. Having a proper customized profile is thus also necessary to fit them in the groups.

Factor 8: Nostalgist (Screenshots and Videos) Users with high *nostalgist* attribute have the tendency of restoring their gameplay memories by taking screenshots and recording videos. They also share their gameplay memories with others in the activity timeline, so that other players can enjoy the unique scenes and compare to their own gameplay too. Meanwhile, the "thumbs up" and appreciation from the others is their reward.

It is worth noting that the eight factors aim to explore the various attributes of Steam users instead of arbitrarily categorizing each user into a single type. Generally, each individual user shall contain certain scores in all given attributes while the attribute value distribution of different users shall differ. Meanwhile, each user may also contain high or low score in multiple attributes simultaneously. By reducing the variable dimensions to one for each attribute and normalizing the value, each individual user shall have a radar chart illustrating his/her salient attributes. Fig. 1 shows an example of a user who possesses a salient attribute of *improver* and is creative with workshop items and also loves to contribute in improving games by giving reviews. Meanwhile, this particular

user also possesses relevantly salient attributes of *elite*, *achiever*, and *provider*. It indicates that the user also favors gaining levels, badges, and achievements, and providing guides and artworks to the community.

Table 3. An Example Mapping between Preference Attributes and Motivation Types

Attributes	Steam Variables	Motivation Types [10, 36]	Gameful Elements [40]
Elite	Level	Mastery	Progression
	Badges	Mastery	Incentive
	Friends	Relatedness	Socialization
	Profile Customization	Autonomy	Customization
Achiever	Games	Mastery	Progression
	Achievements	Mastery	Incentive
	Perfect Games	Mastery	Incentive
Provider	Guides	Mastery, Purpose	Altruism
	Artwork	Autonomy	Altruism
Completer	Showcases	Autonomy, Mastery	Customization
	Game Completion Rate	Mastery	Progression
Improver	Workshop Items	Autonomy, Purpose	Altruism
	Reviews	Autonomy, Purpose	Altruism
Trader	Items Owned	Mastery	Incentive
	Trades Made	Relatedness	Socialization
	Market Transactions	Relatedness	Socialization
Belonger	Groups	Relatedness	Socialization
	Profile Customization	Autonomy	Customization
Nostalgist	Screenshots	Autonomy, Relatedness	Socialization
	Videos	Autonomy, Relatedness	Socialization

To apply such a preference framework in gamification design, based on the variables each attribute is related to, we could find connections between attributes and the established intrinsic motivation types or other similar gamification design models or frameworks. With different player motivation and design elements frameworks, the application towards personalized gamification design could differ. Table 3 is an example of connecting the obtained preference attributes with the SDT motivation types [10, 36] and the gameful design elements categories [40]. Ideally, each Steam variable can be mapped to a certain type of motivation and a particular gameful design element category. Subsequently, the motivation that drives the corresponding preference attributes and the related gameful design element set can be decided and weighted (e.g., based on relatedness of the variables to the attributes). However, such presumption of connecting attributes, motivation types, and design elements can be subjective, when the motivation of each user towards each individual Steam variable is unknown and hard to be dichotomized. For example, 'Level' is likely to be driven by the motivation of mastery, when, on the other hand, particularly in Steam, higher level means that the user will have more badges and showcases to customize. Therefore, the 'Level' variable

is driven by the motivation of autonomy, to some extent. Furthermore, a quantifiable value of 'Level', together with 'Badges' and 'Profile Customization', can be also seen as the tendency towards social comparison. Such equivocality shall be addressed with potential ordering or voting schemes.

5 Discussion

Compared with Lim and Harrell's study on players' social identity [22], we cover more perspectives of Steam users' social behaviors in the gamer community by extending the data collection to more features. However, different from Sifa et al.'s work [38] our data covers only the Steam users' profile information and not users' in-game behaviors. Thus, with the current dataset, mapping from the obtained user preferences towards the gameful design elements regarding heavily in-game behaviors, such as, immersion or risk/reward, is not possible [40]. Furthermore, based on the goal of this study to study users' preference regarding gamification design, the data limits generalization towards all gamification users instead of only gamers. Despite the above limitations, the data (similar to other product-oriented social media profiles, e.g. Amazon profiles) can be seen as more generalized rather than focusing on gamers from specific games or genres. Compared with previous studies on gamification user types [40,42], such data collected from user profiles can be more objective than self-reported survey data.

This study presents a data-driven approach to investigating users' preferences towards game design elements. The resulting axes of variation among players can be inspected and used in gamification. In future work the results can also be used as a basis for categorization of players; data-driven approaches [16] can improve efficiency and representativeness compared to manually designed categories. One follow-up direction is to build a collaborative filtering recommender system based on similarity of users' preference towards various game design elements, allowing a personalized gamification design based on the recommendation for each user [41]. Another future direction is to validate the user preference framework with empirical analysis. For example, the user preference scale of Tondello et al. [42] can be adopted as a reference, with Steam users as participants. Furthermore, the data volume can be enlarged with more users, e.g., by crawling from multiple seed users; our data could further be combined with additional data regarding, e.g., players' in-game behaviors, preference on game genres, and reviews on games. After validation, the proposed user preference framework can be applied to future data-driven player studies. Together with previous gamification design methods [27], the framework will facilitate gamification design and provides an efficient way to address key issues in the user analysis phase [28].

6 Conclusion

We presented an exploratory way of analyzing user presences towards game design elements using Steam user profile data. Using EFA, eight factors/attributes are gained, the value of which can be used to define each individual user's preference regarding behaviors in the Steam community. Together with the connection between such behaviors and the underlying motivation types and gameful design elements, each user's preference

regarding gamification systems can be also perceived. Due to the quantifiable and objective nature of the data, such estimation of the users' preference can be more precise. It will contribute to the future work of personalized gamification design and creation of recommender systems for personalized gamification in a data-driven manner.

Acknowledgments. This research was supported by the Academy of Finland project Centre of Excellence in Game Culture Studies (CoE-GameCult, 312395).

References

1. Barata, G., Gama, S., Jorge, J.A., Gonçalves, D.J.: Relating gaming habits with student performance in a gamified learning experience. In: Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play. pp. 17–25. ACM (2014)
2. Bartle, R.: Hearts, clubs, diamonds, spades: Players who suit muds. Journal of MUD research **1**(1), 19 (1996)
3. Bateman, C., Lowenhaupt, R., Nacke, L.: Player typology in theory and practice. In: DiGRA Conference (2011)
4. Baumann, F., Emmert, D., Baumgartl, H., Buettner, R.: Hardcore gamer profiling: Results from an unsupervised learning approach to playing behavior on the steam platform. Procedia Computer Science **126**, 1289–1297 (2018)
5. Becker, R., Chernihov, Y., Shavitt, Y., Zilberman, N.: An analysis of the steam community network evolution. In: Electrical & Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of. pp. 1–5. IEEE (2012)
6. Bertens, P., Guitart, A., Chen, P.P., Periáñez, Á.: A machine-learning item recommendation system for video games. arXiv preprint arXiv:1806.04900 (2018)
7. Blackburn, J., Simha, R., Kourtellis, N., Zuo, X., Long, C., Ripeanu, M., Skvoretz, J., Iamnitchi, A.: Cheaters in the steam community gaming social network. arXiv preprint arXiv:1112.4915 (2011)
8. Crawford, A.V., Green, S.B., Levy, R., Lo, W.J., Scott, L., Svetina, D., Thompson, M.S.: Evaluation of parallel analysis methods for determining the number of factors. Educational and Psychological Measurement **70**(6), 885–901 (2010)
9. Cronbach, L.J.: Coefficient alpha and the internal structure of tests. psychometrika **16**(3), 297–334 (1951)
10. Deci, E.L., Eghrari, H., Patrick, B.C., Leone, D.R.: Facilitating internalization: The self-determination theory perspective. Journal of personality **62**(1), 119–142 (1994)
11. Denisova, A., Cairns, P.: First person vs. third person perspective in digital games: Do player preferences affect immersion? In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. pp. 145–148. ACM (2015)
12. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: defining gamification. In: Proceedings of the 15th international academic MindTrek conference. pp. 9–15. ACM (2011)
13. Hair, J.F., Black, W.C., Babin, B.J., Anderson, R.E., Tatham, R.L., et al.: Multivariate data analysis (vol. 6) (2006)
14. Hamari, J., Koivisto, J., Sarsa, H.: Does gamification work?–a literature review of empirical studies on gamification. In: 2014 47th Hawaii international conference on system sciences (HICSS). pp. 3025–3034. IEEE (2014)
15. Hamari, J., Tuunanen, J.: Player types: A meta-synthesis. Transactions of the Digital Games Research Association **1**(2), 29 (2014)
16. Han, J., Pei, J., Kamber, M.: Data mining: concepts and techniques. Elsevier (2011)

17. Hendrickson, A.E., White, P.O.: Promax: A quick method for rotation to oblique simple structure. British journal of statistical psychology **17**(1), 65–70 (1964)
18. Horn, J.L.: A rationale and test for the number of factors in factor analysis. Psychometrika **30**(2), 179–185 (1965)
19. Hsu, S.H., Kao, C.H., Wu, M.C.: Factors influencing player preferences for heroic roles in role-playing games. CyberPsychology & Behavior **10**(2), 293–295 (2006)
20. Kaiser, H.F.: The varimax criterion for analytic rotation in factor analysis. Psychometrika **23**(3), 187–200 (1958)
21. Kallio, K.P., Mäyrä, F., Kaipainen, K.: At least nine ways to play: Approaching gamer mentalities. Games and Culture **6**(4), 327–353 (2011)
22. Lim, C.U., Harrell, D.F.: Developing social identity models of players from game telemetry data. In: AIIDE (2014)
23. Lin, D., Bezemer, C.P., Hassan, A.E.: Studying the urgent updates of popular games on the steam platform. Empirical Software Engineering **22**(4), 2095–2126 (2017)
24. Lin, D., Bezemer, C.P., Hassan, A.E.: An empirical study of early access games on the steam platform. Empirical Software Engineering **23**(2), 771–799 (2018)
25. Lin, D., Bezemer, C.P., Zou, Y., Hassan, A.E.: An empirical study of game reviews on the steam platform. Empirical Software Engineering pp. 1–38 (2018)
26. Marczewski, A.: Even ninja monkeys like to play: Gamification, game thinking & motivational design. Gamified UK (2015)
27. Mora, A., Riera, D., Gonzalez, C., Arnedo-Moreno, J.: A literature review of gamification design frameworks. In: 2015 7th International Conference on Games and Virtual Worlds for Serious Applications (VS-Games). pp. 1–8. IEEE (2015)
28. Morschheuser, B., Hassan, L., Werder, K., Hamari, J.: How to design gamification? a method for engineering gamified software. Information and Software Technology **95**, 219–237 (2018)
29. Nacke, L.E., Bateman, C., Mandryk, R.L.: Brainhex: preliminary results from a neurobiological gamer typology survey. In: International Conference on Entertainment Computing. pp. 288–293. Springer (2011)
30. O'Neill, M., Vaziripour, E., Wu, J., Zappala, D.: Condensing steam: Distilling the diversity of gamer behavior. In: Proceedings of the 2016 Internet Measurement Conference. pp. 81–95. ACM (2016)
31. Orji, R., Nacke, L.E., Di Marco, C.: Towards personality-driven persuasive health games and gamified systems. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 1015–1027. ACM (2017)
32. Orji, R., Tondello, G.F., Nacke, L.E.: Personalizing persuasive strategies in gameful systems to gamification user types. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. p. 435. ACM (2018)
33. Pontes, H.M., Griffiths, M.D.: Measuring dsm-5 internet gaming disorder: Development and validation of a short psychometric scale. Computers in Human Behavior **45**, 137–143 (2015)
34. Reilly, A., Eaves, R.C.: Factor analysis of the minnesota infant development inventory based on a hispanic migrant population. Educational and psychological measurement **60**(2), 271–285 (2000)
35. Ryan, R.M., Deci, E.L.: Intrinsic and extrinsic motivations: Classic definitions and new directions. Contemporary educational psychology **25**(1), 54–67 (2000)
36. Ryan, R.M., Deci, E.L.: Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. American psychologist **55**(1), 68 (2000)
37. Sifa, R., Bauckhage, C., Drachen, A.: The playtime principle: Large-scale cross-games interest modeling. In: CIG. pp. 1–8 (2014)
38. Sifa, R., Drachen, A., Bauckhage, C.: Large-scale cross-game player behavior analysis on steam. Borderlands **2**, 46–378 (2015)

39. Stewart, B.: Personality and play styles: A unified model. Gamasutra, September **1** (2011)
40. Tondello, G.F., Mora, A., Nacke, L.E.: Elements of gameful design emerging from user preferences. In: Proceedings of the Annual Symposium on Computer-Human Interaction in Play. pp. 129–142. ACM (2017)
41. Tondello, G.F., Orji, R., Nacke, L.E.: Recommender systems for personalized gamification. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization. pp. 425–430. ACM (2017)
42. Tondello, G.F., Wehbe, R.R., Diamond, L., Busch, M., Marczewski, A., Nacke, L.E.: The gamification user types hexad scale. In: Proceedings of the 2016 annual symposium on computer-human interaction in play. pp. 229–243. ACM (2016)
43. Tseng, F.C.: Segmenting online gamers by motivation. Expert Systems with Applications **38**(6), 7693–7697 (2011)
44. Xu, Y., Poole, E.S., Miller, A.D., Eiriksdottir, E., Kestranek, D., Catrambone, R., Mynatt, E.D.: This is not a one-horse race: understanding player types in multiplayer pervasive health games for youth. In: Proceedings of the ACM 2012 conference on computer supported cooperative work. pp. 843–852. ACM (2012)
45. Yee, N.: Motivations for play in online games. CyberPsychology & behavior **9**(6), 772–775 (2006)
46. Zwick, W.R., Velicer, W.F.: Comparison of five rules for determining the number of components to retain. Psychological bulletin **99**(3), 432 (1986)

PUBLICATION
II

Patches and Player Community Perceptions: Analysis of No Man's Sky Steam Reviews

Chien Lu, Xiaozhou Li, Timo Nummenmaa, Zheying Zhang, and Jaakko Peltonen

In: *DiGRA '20 - Proceedings of the 2020 DiGRA International Conference*. Ed. by Dale Leorke. 2020

Patches and Player Community Perceptions: Analysis of No Man's Sky Steam Reviews

Chien Lu, Xiaozhou Li, Timo Nummenmaa, Zheying Zhang, and Jaakko Peltonen
Tampere University
Kalevantie 4, 33100 Tampere, Finland
firstname.lastname@tuni.fi

ABSTRACT

Current game publishing typically involves an ongoing commitment to maintain and update games after initial release, and as a result the reception of games among players has the potential to evolve; it is then crucial to understand how players' concerns and perception of the game are affected by ongoing updates and by passage of time in general. We carry out a data-driven analysis of a prominent game release, No Man's Sky, using topic modeling based text mining of Steam reviews. Importantly, our approach treats player perception not as a single sentiment but identifies multiple topics of interest that evolve differently over time, and allows us to contrast patching of the game to evolution of the topics.

Keywords

No Man's Sky, Player Modeling, Topic Modeling

INTRODUCTION

Steam is one of the biggest digital game distribution platforms. In addition to being a platform for purchasing and playing, it is also a community where members have mutual discussions and game owners can write game reviews to share their opinions and game-play experiences. When writing a game review, the player can label the game as "recommended" or "not recommended". How much time (hours) the player has played the game when writing the review is also recorded. Steam reviews form an important view into a game's reception among players; an overall summary of the proportion of positive reviews is prominently reported on a game's Steam store page and is also often reported on third-party sites and in various news and social media discussion on a game. However, such an overall summary does not reveal the main concerns reviewers report on, or the change of their perceptions over time.

Steam allows for games to be easily updated once released. Developers release updates on their own schedule; on the players' side, available updates can be set to happen automatically, making it easy for users to keep their games up to date. The dynamic of such updates is very different compared to the earlier practice without a unified platform, where users needed to download patches from developer or publisher websites. Steam also supports the sale of downloadable content (DLC), allowing developers to esily add paid content to already published games. In addition to normal game releases, Steam has a programme called early access (*Steamworks Documentation: Early Access* Accessed 8 Dec 2019). It

Proceedings of DiGRA 2020

allows developers to already sell games that are still in development and not ready for a final release on the platform. The impact of such an environment, where game updates are commonplace, on player perception has not been explored in game research in a comprehensive quantitative way. In this work we do so in context of a particular game, No Man's Sky.

No Man's Sky is an action-adventure survival game first launched in August 2016. It received strongly critical responses from players due to lacking features that had been promised to be in the game. However, since launch the game has been supported by rapid updates; it has had 8 major updates so far, denoted by versions 1.00, 1.10, 1.20, 1.30, 1.50, 1.70, 1.75 and 2,00, released on 12 August 2016, 26 November 2016, 8 March 2017, 11 August 2017, 24 July 2018, 29 October 2018, 22 November 2018 and 14 August 2018, respectively[1].

The changes to No Man's Sky exemplify several typical types of change in digital games. Digital games, as other software products, evolve during their lifecycle. The evolution of games can be in the form of emerging change (designing a space for the players to mold their own game experiences), reactive change (changing the game by reacting to direct or indirect feedback from the players) or pre-planned change (content that is already designed, or in some cases already produced, before the launch of the game), and their evolution has similarities to how more utility focused software products evolve, but not all properties are equal (Nummenmaa et al. 2013). No Man's sky is a mixture of all three types of evolution. Emerging change is built into the system, even if lightly, as the game world is generated piece by piece when users access new worlds. Reactive change is prevalent, as new changes are implemented due to feedback in the form of patches. Pre-planned change is also present, as the developer has implemented features that have already been promised prior to release. As Newman (2012) has pointed out, due to two factors, ports (transferring to different operating systems or platforms) and patches (updating or adding new features), a game itself is an unstable object. Due to this nature of games, the player perceptions or experiences of a game are therefore dynamically changing over time.

Despite research on various aspects of game development, and other research on impact of game updating strategies, there is a lack of research coupling large-scale analytics of different aspects of players' reception of a game to aspects of the update strategy, and in this work we do that. Our research questions are: RQ1 - What are the main topics of discussion (e.g. themes of concern or appreciation) in players' reviews? RQ2 - How do the contents of reviews change over time, are some topics rising or falling over time, and at what rates? RQ3 - How does the presence of the topics differ in reviews recommending versus not recommending the game? RQ4 - How do updates carried out to the game coincide with changes in the topical content of the reviews?

To answer the research questions, in this research, a collection of more than 85 thousands user reviews across roughly 3 years of *No Man Sky* from steam were analyzed. We use a machine learning based text mining technique called topic modeling to analyze the collected data in a computational manner. The model we use both extracts topics as semantically meaningful themes in players' reviews, and also models the relationship between the presence of those themes, time of the review, players' playing hours, and players' attitude in the are distilled/extract their relationships between players' overall attitude towards the game in the sense of whether they recommend the game or not.

In the following we first discuss a selection of related work on challenges in game development, customer feedback and review analysis, update strategy planning, and analysis of updates and reviews on Steam. We then discuss the method from data collection to the text analysis. Next, we discuss the results first in terms of the extracted topics and in terms of their prevalence over time. Lastly, we provide discussion and conclusions.

RELATED WORK

Many studies have addressed the challenges and issues in computer game development practice from software engineering perspectives (Alves et al. 2007; Kanode and Haddad 2009). Kanode and Haddad (2009) list and specify several challenges in game development in terms of software engineering, including assets, scopes, process, publishing, management, team organization and third-party technology. Ampatzoglou and Stamelos (2010), by reviewing the literature, examine the use of software engineering theories, methods and tools in game development practice and find that game developers tend to fit traditional software engineering methods to game development with certain adjustments. The authors also indicate that the maintenance activities within game development are mainly corrective and maintenance and verification in game development are often neglected. However, the game products can be changed significantly due to the feedback from testing phase and the market (Kasurinen et al. 2014). Thus, enabled by the contemporary online distribution channels, updating games correctively and perfectively can improve them significantly towards enhanced customer satisfaction.

In order to improve products and services, Customer feedback is an important data source for companies to understand the market and the needs of their customers (Cho et al. 2002; Wu et al. 2010). In the software engineering domain, end user feedback is also critical for facilitating the evolution of software products and services (Burnett et al. 2004; Ko et al. 2011). The importance of end users as stakeholders is particularly enhanced for mobile applications, since they are commonly distributed through online platforms (Holzer and Ondrus 2011). The combination of collectable end user feedback and traceable software evolution allows further requirements analysis to be done effectively through statistical and data-driven methodologies, in order to plan future changes and to be aware of how the changes may impact user satisfaction (Palomba et al. 2015). Following advances in data mining, many studies have provided various approaches towards effective review analysis to uncover critical user needs (Fu et al. 2013; Chen et al. 2014; Guzman and Maalej 2014; Li et al. 2018). Hence, despite their differences from mobile app reviews, video game end user reviews can also provide valuable information that game developers can take into account in order to improve their game products (Lin et al. 2019).

The evolution of software products is considered important for maintaining their quality, when together with the widely applied incremental and agile development methods, users receive early releases of software products and are more likely to support their evolution with meaningful feedback from which requirements are elicited and prioritized effectively and continuously (Greer and Ruhe 2004). Thus, an effective planning for software release is highly required. Many studies have contributed to the practice of software release planning, in terms of the process, decision making, strategic models and tools (Ruhe and Saliu 2005; Saliu and Ruhe 2005; Svahnberg et al. 2010). In particular for mobile applications that are distributed at unified online platforms where updates are easily delivered and feedback are

instantly received, Nayebi et al. (2016) find that developers tend to follow predefined rational update strategies and mostly believe that frequent updates and the different release strategies shall affect the app quality and users' feedback. Hence, towards effective mobile application specific release planning strategies that ease the developers' efforts and respond swiftly to users' concerns and complaints, many studies have proposed approaches and tools facilitating mobile app release planning (Villarroel et al. 2016; Ciurumelea et al. 2017; Scalabrino et al. 2017). Comparatively, even though many players still choose to purchase physical copies, the video games online distribution platforms have grown rapidly with the advantages of easiness to find relevant games based on preferences, affordability, easiness of payment, and so on (Toivonen and Sotamaa 2010). The mechanisms of the platforms also enable developers to constantly listen to players' feedback and update accordingly, but also requires them to plan the updates properly, especially for "Early Release" games (Lin et al. 2018).

As one of the most popular digital game distribution platforms, Steam provides not only video game purchasing and downloading service but also online communities for the players to review games and for developers to respond. Due to the notable volume and dynamic of the data contained in Steam, it has been widely used for research purposes (Kang et al. 2017; Lin et al. 2019). Lim and Harrell examine the players' behaviors of profile and social network maintenance and analyze the differences in their player identities (Lim and Harrell 2014). Slivar et al. analyze the the impact of game types and video adaptation strategies on the quality of the experience (Slivar et al. 2015). To investigate player behaviors on Steam, Sifa et al. analyze the players' different playtime frequency distribution and investigate their engagement and cross-game behavior (Sifa et al. 2014; Sifa et al. 2015). Regarding game updates on Steam, Lin et al. conduct an empirical study of the urgent updates of the 50 most popular games and find that the choice of update strategy affects the proportion of compulsory urgent updates (Lin et al. 2017). Furthermore, regarding game reviews, Lin et al. perform an empirical study on the reviews of 6224 games on Steam and analyze the review content and the relation between players' play hours and their reviews (Lin et al. 2019).

METHOD

Data Collection

In this study, we use Scrapy[2], an open-source Python-written web crawling framework, to obtain the user review data from the Steam platform, specifically for the game No Man's Sky. Scrapy was first released in 2008 with its latest version 1.8.0 being compatible with Python 3.5 and later versions. Technically, in order to crawl structured review data from the community page of a particular game, we define a Spider with Scrapy and run it through the crawler engine. The crawling process starts with a request on the URL of the game community page and calls the default callback method, which loops through the elements (i.e., review items) with CSS selectors and yields a dictionary with the requested information. Notably, Scrapy is able to crawl the content loaded with Javascript via users' scrolling that cannot be obtained using BeautifulSoup[3], which is crucial for crawling Steam user review data.

As a result of the crawling, we obtain the 85805 unique user reviews on No Man's Sky from its release date, August 12th 2016, to October 5th 2019. The features of the data include

review publication date, review text, user ID, the recommend/not recommend flag for the review, the user's play hours, the number of products owned by the user, the number of people who rated the review as helpful and the number of people who rated the review as funny. Among the obtained reviews, over half (44335) were given within the first month of the game release, with 59.14% of the users not recommending the game. During the timespan of the data set the overall recommendation rate has increased to 53.03%.

Text Analysis

We employ a text analysis technique called Structural Topic Model (STM, Roberts et al. 2016) to analyze the collected review texts. STM has been applied to analyze gaming discussion on game development (Lu et al. 2019) and trophy hunting (Lu et al. 2020). Topic modeling represents document content as a mixture of underlying topics, each of which has a distribution of typical words; these underlying topics and their prevalences in each document in the collection are found by fitting the topic model to the data set. The resulting topics and their prevalences over the documents can then be analyzed. The resulting topics can describe subjects of discussion, but can also describe other elements such as tone of writing. Among a set of multiple topics, some may differ greatly from one another while others may be more similar, describing differences of emphasis within a common theme. Compared to methods such as Latent Dirichlet Allocation (LDA, Blei et al. 2003) and Dynamic Topic Model (DTM, Blei and Lafferty 2006), the STM technique that we use is a more advanced model which is able to take available document-level covariates into account when modeling the text. For Steam reviews, we take into account several covariates in the modeling: the user recommendation (recommend or not) indicates the general positive/negative evaluation of the game, thus it is taken as one of the covariates; we also take the posting time as a covariate in order to model the evolution of the review content over time; the user's playing hours which reflects their amount of experience with the game as a player is also included as a covariate in the model.

Before the model traing, stop words (e.g. 'is', 'this', 'etc') and rare words (words that only appeared once in the whole text corpus) were removed and all the words were then lemmatized. The lemmatization technique takes morphology into account and can find unified forms for more complicated cases, such as irregular verbs (e.g. 'drive', 'drove' and 'driven' are lemmatized to their common lemma 'drive').

The final model was decided based on the criterion called held-out likelihood. To compute the value of the criterion, a proportion (50%) of a small subset of the collected documents is considered unobserved ("held out") and is not used to build the topic distributions, and the STM models are evaluated by their likelihood on this held-out portion, representing the ability of the models to represent previously unseen text. In our large collection of reviews, it is possible to find a large number of underlying topics, and thus it we chose the number of topics by a careful search. We first searched among topic numbers $K = 10, 20, ...$ to 100, with an interval of 10. After finding that the model with $K = 50$ had the maximum value of the held-out likelihood criterion, the search was focused around $K = 50$, and a more detailed search for possible improved values with $K = 41$ to 49 and $K = 51$ to 59 was conducted. The model with $K = 55$ ultimately turned out the have the best value of the held-out likelihood after the search and was chosen for the model. Note that the criteria of held-out likelihood has received criticism by e.g. (Chang et al. 2009), however, it is still

a common practice to decide the number of topics with STM (Stamolampros et al. 2019).

After deciding the number of topics, the semantic coherence value (Mimno et al. 2011) was taken as a criterion to choose the best model from multiple runs with different initializations. The semantic coherence value measures how strongly the top words in each topic co-occur over documents, thus, it can be employed to evaluate the performance of topic models and to choose the best-performing model among several models. We built 10 models with 55 topics using the whole dataset, starting from different initializations, and the model with the best average semantic coherence value over topics was selected as the final model.

RESULT

We first discuss the themes found in the extracted topics, then we discuss the evolution of the topics' prevalence over time, and we further discuss the impact of the users' play hours on the prevalence of different topics in their reviews.

Extracted topics

The top 10 words of each topic are listed in Tables 1, 2, 3, and 4. In each table, we list for each topic its most common words, and also its overall proportion ("Pr (%)") representing how much of all review content arises from that topic. The topics are also given descriptive names by the authors by analyzing their top words as well as analyzing example reviews that prominently arise from the topic. Notably, the rich review content in Steam allowed us to extract a large variety of topics with clear semantic meaning; this both indicates that players have a rich variety of concerns relating to the game and its development, and shows the benefit of using text mining approaches for review analysis. We next discuss the found topics.

The topic **Evaluating Game-play** with terms such as *play, fun, get, ...* holds the highest prevalence. One example quote is

´´It's fun at first, but gets boring after a while and then weirdly, gets fun again It's a good game, though it has it's flaws 9/10, go buy the game goddamnit".

Followed by topics **Reaching Recommended Status** and **Appreciating Improvements**. They reflect players' positive perceptions after improvements of the game. Other similar topics include **Gradual Improvement** which emphasizes temporal aspects (e.g. with terms such as time, long and due); For example, one quote of the topic **Gradual Improvements** is

´´this has supprised me the last few times i have played. at first i was not overly happy with my purchase but as time has gone by i have become more happy. is it poerfect, no, but it is fun to play now."

A certain amount topics are related to updates, including **Updates and Added Content**, **Change of Game** and **Upgrades and Items**. One example quote from the topic **Updates and Added Content** is:

´´The Foundation Update has added a good amount of new features to the game, including

building bases and owning Freighters. This is just the foundation of future updates so you can see Hello Games are actively working to make the game better. Kudos to them."

Among other topics with higher prevalence, some of them are related to game purchase such as **Worth the Price**, **Pre-order** and **Refund**; some are directly related to disappointment to the promises, one example is the topic **Disappointment to Promise and Hype**, one example quote is :

´´I preordered that game thinking it would be magical after seeing the trailers. It was not what was promised then. I tried again after beyond because they said they made it a lot better. yes some things changed but this game is so empty and ugly compared to what they said it was. the only fun i had was building my base which is now broken after a patch. I don't even want to figure it out. Quests ? boring as hell and there is nothing rewarding in them. thank you hello games, i was naive, now i will never preorder anything again, I'll wait to see what kind of crap a game really is before jumping in...."

another similar topic is **Lies and Miss Promises**, one example quote is:

´´The trailers lie. Most of the stuff in the trailers don't exist in the game. They are selling this to us under false pretenses which is illegal."

Other topics that directly reflect to the disappointments include **Strong Dislike, waste of money**, **Feels Unfinished, Indie vs AAA (Quality Level)**, **Quick Disappointment**, **Lack of Content / Grind** and **Recurring Bad Game-play**. One example quote from the topic **Strong Dislike, waste of money**

´´♥♥♥♥ YOU ♥♥♥♥ THIS GAME. YOU SHOULD GO BANKRUPT . NO ♥♥♥♥ING WAY IM GONA BUY FROM YOUR ♥♥♥♥IN COMPANY GAME. ♥♥♥♥ ♥♥♥♥♥"

and example quotes from the topic **Feels Unfinished, Indie vs AAA (Quality Level)**

´´In it's current state it feels as if I purchased an early access game.", ´´AAA price. Indie gameplay. 11/10".

Despite the negative feelings, there are some topics related to the appriciations including **Appreciation**, **Enjoyment of Play Experience** and **Enjoyment Despite Flaws**. One example quote of the topic **Appreciation** is

´´Best game,a lot of oprtunities and no limits you always have something to do,they have redeamed theirselfs.".

There are also topics that are related to other reviews. **Reviews vs. Reality** with top words such as *review, read, see, okay, say...* is about players' reflections on other reviews or the comparison between the reviews the players have read their own experiences. One example quote of this topic is

´´I really do not understand why there is so uch negative comments and posts about this game. I totally enjoy it and..."

Another topic related to other players' reviews is **Not as Bad as People Say** with top words such as *people, say, review, give, think, everyone ....* Where are some example quotes of this topic that are trying to defend the game:

´´This game is amazing, I woud love to wright a proper review but I am speechles and I dont understand why the reviews are so negative, thats all I have to say, this game is amazing..."

and

´´Some people like it, some don't. Some people anticipated more, some didn't. Some people feel tricked, some don't. Some people love Trump, some don't. I personally think that if this game was named as "Early Acces Game"(which it technically is) then a lot of things would be different"

Technical issues are also discussed in the collected game reviews. Relevant topics include **Graphics Settings**, **Crashes** and **Bugs and Glitches**. For example, one quote from the topic **Crashes** is

´´i cant even start playing, the minute the game starts to load it crashes".

Some topics are related the to details of game-play, including **Moving and Looking**, **Base-building and Desired Content**, **Spaceship Travel and Combat**, **Repetitive Resource-collection Game-play**, **Exploration and Discovery** and **Material Collection**. One example quote from **Repetitive Resource-collection Game-play** is

´´Explore, collect resources so you can keep exploring, repeat..."

Besides, some topic are specifically associated to PCG content, e.g. **Procedurally Generated Universe and Exploration** and **Procedural Content Generation of Maps**.

One quote from the topic **Procedurally Generated Universe and Exploration** is

´´A very interesting game where you explore a procedurally generated universe. It is very chill and if you enjoy relaxedly exploring a universe it's a great time. If you're looking for a survival game or fps or flight sim its not really that, though it has elements of that. Good stuff if you like exploration though."

Topic Prevalence over Time

Figures 1, 2, and 3 show the temporal dynamics of the topic prevalence. The blue line presents the prevalence among players who recommend the game and the red line presents the prevalence among the players who do not recommend the game; dotted lines around them represent 95% confidence intervals of the topic prevalences. The date of updates are also marked with black and green bars; the black bars represent the starting date of each major update and green bars are smaller updates.

In the plots of topic prevalence over time, 20 out of 55 plots show that the prevalence of the topic is always high among players who do not recommend the game; note that Figures 1, 2, and 3 only show a selection of interesting plots; the 20 out of 55 ratio was verified for the

Table 1: Extracted topics, part 1

Topic	Pr (%)	Top Words
Evaluating Game-play	4.46	play, fun, get, buy, hour, pretty, first, bore, good, like, couple, look, day, soon, gameplay
Reaching Recommended Status	3.69	update, launch, still, next, since, improve, change, come, beyond, finally, recommend, original, long, recent, theyve
Appreciating Improvements	3.46	good, great, lot, amaze, awesome, nice, keep, ton, job, perfect, overall, game, work, friend, fun
Worth the Price	2.94	worth, price, sale, recommend, full, definitely, quite, pay, enjoyable, pick, say, tag, good, chill, like
Game Lifecycle, Work of the Developers	2.81	release, year, dev, game, work, late, continue, ago, free, developer, day, finally, week, dlc, month
Interaction with Other Players	2.73	want, try, play, can, give, friend, another, multiplayer, like, else, make, back, someone, think, time
Not as Bad as People Say	2.68	people, say, review, give, think, everyone, hate, like, negative, see, alot, hope, positive, good, personally
Graphics Settings	2.64	run, setting, gtx, graphic, ram, low, high, max, fine, spec, set, smooth, card, window, computer
Disapointment of Promise and Hype	2.62	promise, hype, wait, buy, deliver, disappoint, title, worth, live, hope, game, pay, trash, huge, preorder
Strong dislike, Waste of money	2.54	money, ♥♥♥♥, waste, suck, buy, crap, ♥♥♥♥ing, copy, garbage, paste, piece, back, scam, ever, ♥♥♥♥♥♥
Lies and Missed Promises	2.53	lie, feature, show, advertise, trailer, miss, promise, developer, video, false, product, many, multiplayer, sell, unfinished
Appreciation	2.30	love, time, play, start, hour, keep, cool, idea, absolutely, feel, always, beautiful, first, put, experience
Enjoyment of Play Experience	2.30	enjoy, explore, like, far, bite, thing, play, look, feel, may, relax, slow, find, although, kind
Feels Unfinished, Indie vs AAA (Quality Level)	2.20	feel, content, early, potential, access, current, amount, indie, aaa, simply, depth, extremely, high, shallow, market
Crashes	2.16	crash, load, start, screen, work, minute, play, try, computer, playable, unplayable, past, fix, open, min

Table 2: Extracted topics, part 2

Topic	Pr (%)	Top Words
Lack of Content / Grind	2.11	nothing, grind, anything, real, end, like, literally, reason, whole, thing, basically, way, empty, stay, youll
Comparing with Other Games	2.06	space, exploration, like, minecraft, world, survival, elite, dangerous, combat, deep, open, game, adventure, sandbox, simulator
Quick Disappointment	2.00	bad, spend, half, hell, like, hour, big, tech, look, demo, page, ever, forget, seriously, straight
Issues and Patches	1.98	issue, fix, patch, problem, support, performance, work, need, optimization, edit, technical, hopefully, state, experience, poor
Refunds	1.95	refund, steam, hour, ask, return, realize, alpha, buy, playtime, wish, attempt, hope, policy, game, store
Updates and Added Content	1.93	new, update, add, game, bring, content, community, future, major, forward, feature, stick, atla, foundation, improvement
Reviews vs Reality	1.92	review, read, see, okay, say, write, time, watch, decide, know, check, sure, think, post, need
Change of Game	1.80	stuff, good, make, need, like, decent, super, lot, big, easy, yet, little, still, take, slowly
Gradual Improvement	1.73	however, purchase, time, game, happy, long, due, developer, concept, recommend, effort, offer, regret, massive, point
Discussion of Game Versions	1.72	actually, ever, big, game, version, edit, make, call, good, disappointment, back, put, come, sorry, late
Reactions	1.70	know, please, thank, guy, hard, yes, damn, stop, kinda, wow, good, god, let, work, like
General Opinion Words	1.64	thing, like, game, want, think, enjoy, hear, know, type, say, anyone, believe, fantastic, person, follow
Pre-orders	1.58	never, everything, pre, order, every, almost, like, make, ever, first, sit, see, imagine, look, time
Moving and Looking	1.57	around, turn, take, away, look, see, like, move, head, walk, way, hit, one, figure, blow
Variation in Content	1.54	planet, different, look, animal, see, plant, every, rock, color, thing, similar, variation, type, like, generate

Table 3: Extracted topics, part 3

Topic	Pr (%)	Top Words
Performance	1.53	drop, run, terrible, rate, frame, bad, stutter, horrible, port, lag, optimize, poorly, constant, like, console
Basebuilding and Desired Content	1.53	build, base, player, multiplayer, add, story, make, single, character, good, vehicle, friend, new, still, able
Repetitive Resource-collection Game-play	1.50	resource, galaxy, planet, repetitive, center, hour, gather, find, collect, another, repeat, bore, become, first, reach
Procedurally Generated Universe and Exploration	1.45	universe, experience, story, generate, explore, action, procedurally, vast, discovery, infinite, unique, exploration, visual, wonder, truly
Bugs and Glitches	1.44	bug, save, break, progress, time, con, pro, fix, buggy, hour, glitches, play, start, many, file
Enjoyment Despite Flaws	1.43	many, despite, experience, mod, moment, see, along, yet, incredible, true, flaw, dream, consider, become, world
Hype and Expectation	1.41	expect, expectation, small, hype, game, exactly, team, train, review, think, sci, plenty, fan, people, gamers
Lack of Interest and Reward	1.40	feel, lack, make, interest, reward, simple, little, place, point, gameplay, thing, certain, kind, find, purpose
Minor Complaints	1.37	seem, like, use, right, thing, first, sure, able, otherwise, sad, complaint, minor, look, see, lucky
Developers and Studios	1.36	fact, gaming, game, matter, example, say, develop, value, total, never, history, compare, studio, trust, let
Acknowledging and Expecting Improvement	1.35	man, sky, game, become, wonderful, ever, say, truly, come, leap, good, upcoming, next, experience, freelancer
Exploration and Discovery	1.33	planet, find, alien, discover, learn, creature, system, name, new, explore, word, language, race, fauna, species
Control Difficulty	1.19	rather, less, time, mouse, step, hold, three, require, handle, appear, spore, play, impossible, avoid, button
Recurring Bad Gameplay	1.17	happen, still, make, time, constantly, mess, good, instead, box, thing, apparently, somewhere, need, manage, fill

Table 4: Extracted topics, part 4

Topic	Pr (%)	Top Words
Spaceship Travel and Combat	1.14	ship, mine, fly, space, land, planet, station, tool, fuel, sell, resource, attack, multi, weapon, combat
Death	1.13	lose, kill, back, die, leave, annoy, time, find, shoot, sometimes, spawn, try, inside, death, power
Material Collection	1.11	life, find, every, material, planet, farm, start, time, need, minute, thing, walk, take, tutorial, except
Falling Short of Expectations	1.10	tell, gameplay, graphic, short, fall, suggest, requirement, average, little, good, will, hard, level, discount, significant
Travel Between Star Systems	1.10	system, star, travel, jump, entire, end, light, leave, can, explain, set, life, make, find, take
Control Interfaces	1.08	control, option, menu, flight, hold, change, click, press, key, controller, texture, interface, hand, force, button
Upgrades and Items	1.06	ship, upgrade, inventory, item, slot, suit, fight, find, sentinel, trade, management, space, system, need, blueprint
Procedural Content Generation of Maps	0.98	map, design, generation, sound, procedural, limit, world, surface, element, variety, vary, effect, engine, system, encounter
Survival and Challenge	0.91	survival, mode, craft, mechanic, need, challenge, grind, easy, normal, progression, make, creative, use, satisfy, equipment
Interaction with Factions	0.84	large, planet, battle, see, interaction, faction, trade, giant, space, planetary, terrain, player, creature, war, close
Quests	0.83	quest, freighter, mission, pirate, ability, ship, fleet, space, system, base, use, trade, enemy, multiple, main

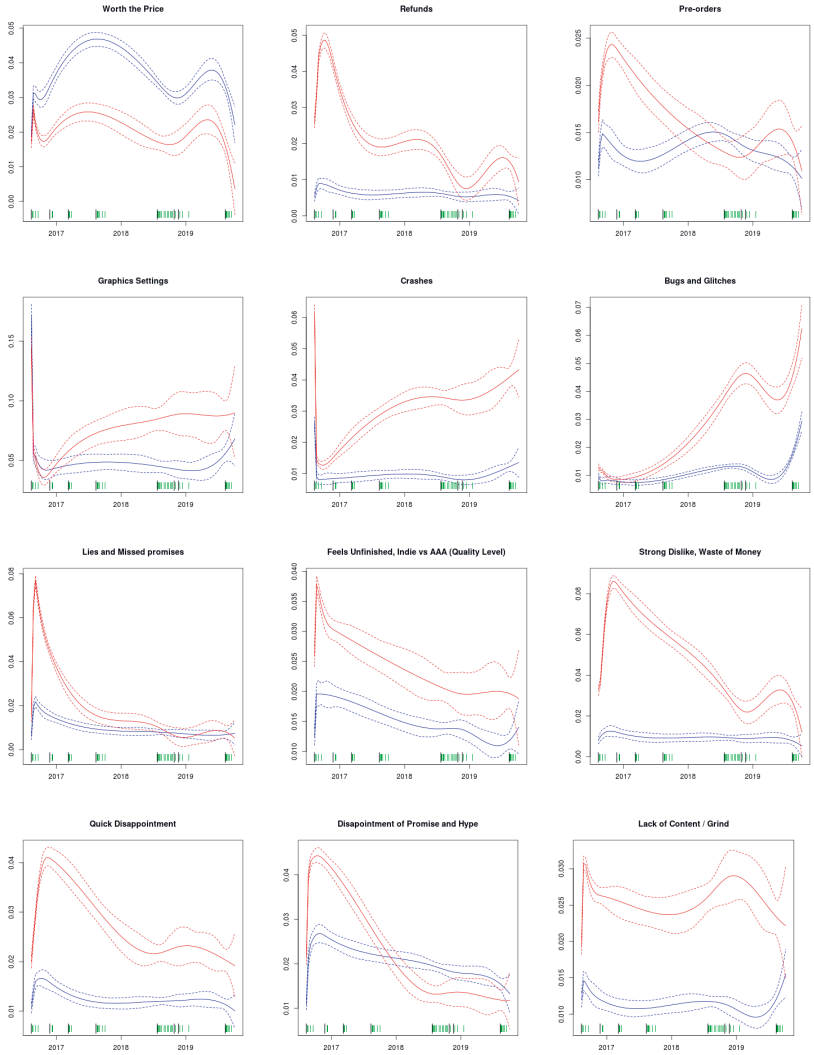

Figure 1: Selected topic prevalence over time. Part 1. Blue: Recommend; Red: Do not recommend; Dot lines: 95% interval.; Black Bar: Time of the updates

whole set of plots. One kind of such topics include players' negative perception and comments on the game, such as **Strong dislike, Waste of money**, **Lies and Missed promises**, **Feels Unfinished, Indie vs AAA (Quality Level)**, **Quick Disappointment**, **Refund**, etc.

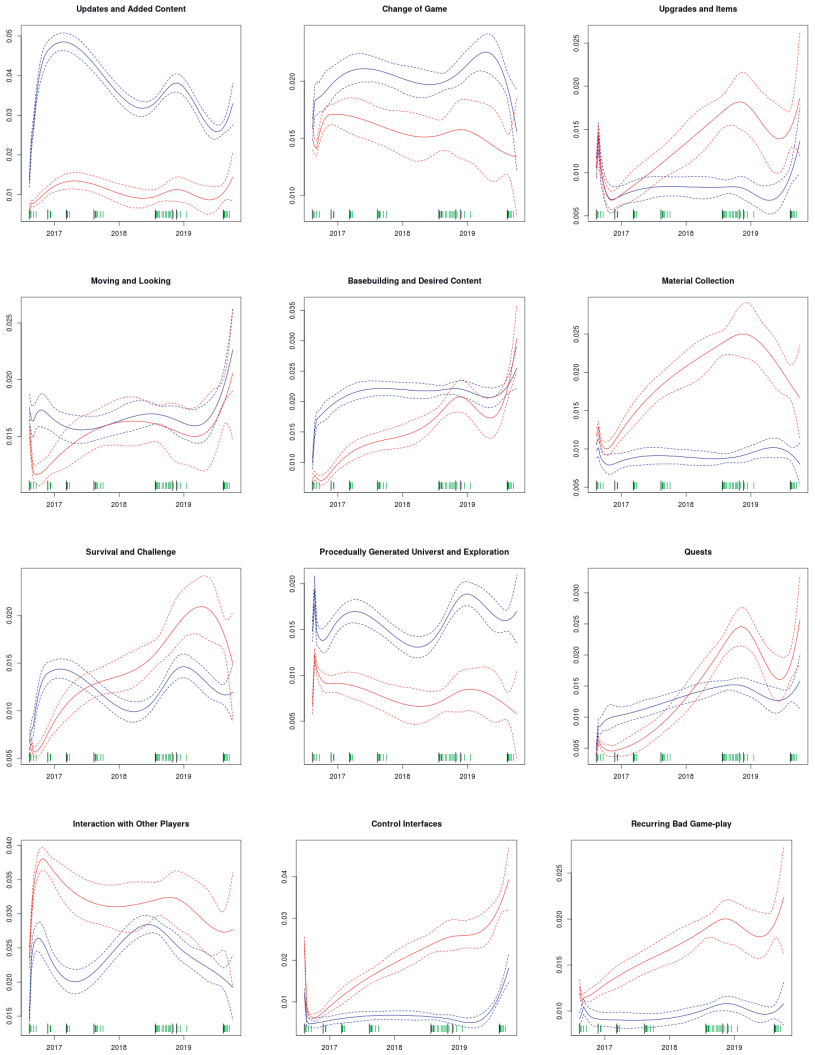

Figure 2: Selected topic prevalence over time. Part 2.

Specifically, the prevalence of these topics immediately reaches the peak after the game was released, and starts declining after the first major release called Foundation (update 1.10) on Nov. 26 2016, and continues declining over the follow-up major releases, which indicates that the game studio's effort on releases do relieve players' strong dissatisfaction. Besides, there are also topics of complaints of game contents or features, and they are **Material Col-**

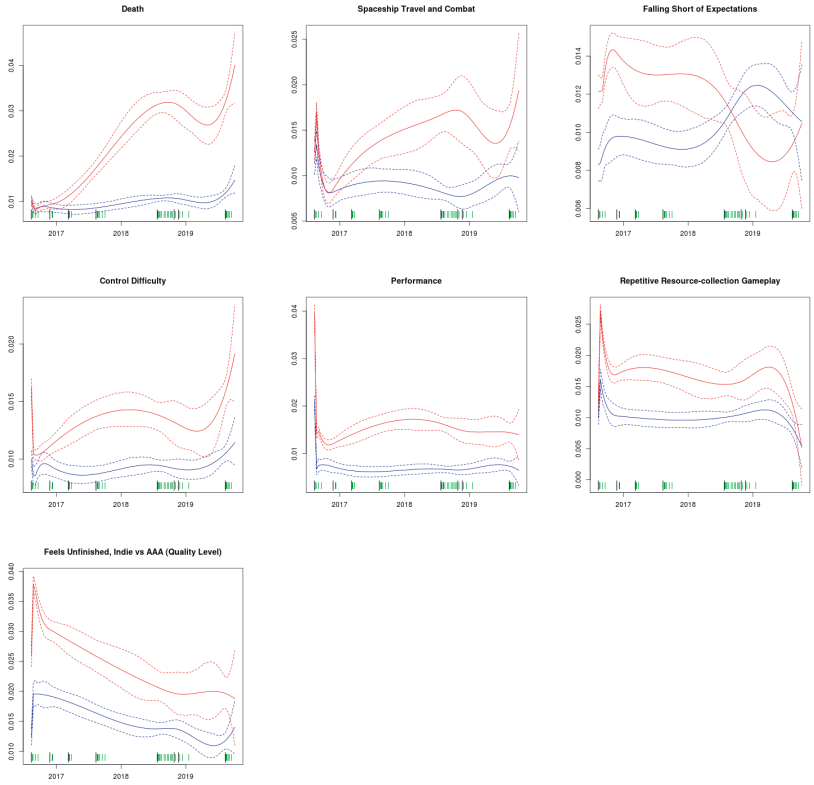

Figure 3: Selected topic prevalence over time. Part 3.

lection, **Control Interfaces**, **Graphics Settings**, **Crashes**, **Bugs and Glitches**, **Recurring Bad Gameplay**, **Spaceship Travel and Combat**, **Death**, etc. Their prevalence increases among players not recommending the game, indicating the bugs and issues remain or are introduced throughout the releases. Some of them might be mitigated in a specific release. For example, prevalence plots of topics such as **Material Collection**, **Bugs and Glitches**, **Recurring Bad Gameplay**, **Spaceship Travel and Combat**, **Death** have a small hump after two major updates in October, 2018 (1.70) and November, 2018 (1.75). Their prevalence declines for a certain period before rising again.

Despite the controversy, there are 19 topics showing more game recommendation since it was released. The topics are related to understanding, appreciation, and acknowledgement of the game and its continuous improvements despite the failure to present the promised features to meet some players' high expectations when the game came out. Some players also defend the game against the complaints and disappointment expressed in other reviews.These clearly implies the game studio's effort on keeping improving games, and

the improvements are appreciated. In addition, the prevalence plots such as **Falling Short of Expectations** and **Disappointment of Promise and Hype** show a decline among players not recommending the game and a rise among players recommending the game. Both support the observation of a gradual increase in players' satisfaction, along with the game updates.

Among game purchase topics, for the topic **Worth the Price**, there is a clear distinction between players who recommend and do not recommend the game, on the other hand, the topics **Refunds** and **Pre-orders** have reached the peak in the beginning and the prevalence of those topics in general has a going-down trend after the peak. The topic **Refunds** might have been driven by the different kinds of early disappointment and reached the peak, One the other hand, **Pre-orders** is a timely topic so the discussion capacity has been done after certain amount of time.

The temporal pattern of the topics that reflect the disappointment varies. **Lies and Miss Promises**, **Feels Unfinished, Indie vs AAA (Quality Level)**, **Strong Dislike, Waste of Money**, **Quick Disappointment** and **Disappointment of Promises and Hype** are frequently discusses among players who do not recommend the game only in the early stage, the prevalence dropped in different time. One the other hand, it seems that **Lack of Content / Grind** is a constantly lasting issue especially among players who don not recommend the game.

When it comes to technical related topics (**Graphics Settings**, **Crashes** and **Bugs and Glitches**), the trend of the prevalence in general has been growing especially among players don't recommend the game. One potential reason is that, due to more and more new features added into the game over time, there is a higher possibility for players to encounter technical difficulties, especially bugs and glitches and result in negative perceptions.

Temporal trend of topics related to game updates (in Figure 2) show different trends in terms of temporal dynamics. The topic **Updates and Added Content** and **Changes of Game** show a overall positive perception and the topic **Upgrades and Items** shows a growing negative perceptions.

Topics related to specific game-play experiences, including **Moving and Looking**, **Basebuilding and Desired Content**, **Survival and Challenge**, **Procedurally Generated Universe and Exploration**, **Quests**, and **Material Collection**, each shows a different pattern. The topic **Moving and Looking** shows a mixture of reviews both from players recommend and do not recommend the game. **Basebuilding and Desired Content** was more prevalent in the beginning among players who recommend the game but the prevalence has grown among players who do not recommend the game and in the end prevalence is roughly the same in both kinds of players. The topic **Material collection** has a growing prevalence among players who do no recommend the game and it reached the peak around the beginning of 2019. The topic **Survival and Challenge** was more prevalent in the beginning among player who recommend the game and in the end it has become more popular in players who who do not recommend the game. The topic **Procedurally Generated Universe and Exploration** has been always popular among players who recommend the game and the topic **Quests** has grown in both kinds of players but the growing changing was more

obvious among players who do not recommend the game.

Examples of Relationships between Updates and Review Content

As already mentioned in the previous subsection, the temporal trends in Figure 2 for topics directly related to game updates already show clear changes over time, with the **Updates and Added Content** and **Changes of Game** topics showing a rising presence in positive (recommending) reviews near major updates, and the topic **Upgrades and Items** showing a growing presence in negative (not recommending) reviews near the updates towards the end of 2018; and several other topics had changes of prevalence associated with times of the updates as discussed above. We next discuss the influence of updates on the reviews in more detail.

The players opinions can be potentially affected by the updates. For example, updates 1.50 (24 July 2018) have added more missions including Real time missions, Scheduled missions, New mission types including freighter combat so on. However, it turns out such content-adding updates aroused complaints. This can be seen in the growing prevalence of the topic **Quest** among players who do not recommend the game. For example, in a review written on 30 July 2018, a user wrote

"I really wanted to like this game, but after 12 hours of game-play and about 10 attempts at the freighter mission. I'm still unable to obtain the freighter. When doing the missions I either recieve wanted level from stray shots at the freighter. Or when I do complete it and recieve the ships transmission to land. I land and then the game hard crashes when I press the button accept the freighter. I spent about 4 hours redoing the mission with the same results on mulitple occasions."

Another example is related to the topic **Survival and Challenge**. Updates 1.10 (26 November 2016) had aroused positive reviews, for example one review written on the next day (27 November 2016) said:

"The new 1.1 update and survival mode add alot of promise to the game, i had uninstalled it but playing survival mode is actually alot more fun than the original game mode !"

And another example review written on the same day said

"New update made this a playable game. If it was too easy before try survival mode, it's pretty brutal."

One example of updates that brought both positive and negative reviews is seen in reviews strongly featuring the topic **Interact with Other Players**, one review written on 29 July 2018 said

"Honestly, this gmae is MIND BLOWING now. Playing it alone or with a friend is so much fun. Give it a chance."

yet another review written on 12 December 2018 said:

"Multiplayer update !! Ok Let's give it a chance, let's buy it. It can't be that bad. Except

that it's not multiplayer. It's observer mode.I was in the middle of a sandstorm walking back toward my ship, while my shields were failing rapidly. My friend was a few steps away, no sandstom, his shields are fine. Can't share anything. This is not multiplayer. Don't call this multiplayer. If this is multiplayer then watching a game on Twitch is multiplayer."

Topic Prevalence and Play Hours

Figure 4 shows the interaction between topic prevalence and play hours; the STM topic model which uses the play hours as a covariate allows us to extract this influence from the model and plot it. The horizontal axis displays how much the topic prevalence of a review increases or decreases when the writer of the review has played one hour; the dot shows the mean increase and the bar shows the 95% confidence interval. Thus, compared to an average review, an increase of one playing hour tends to happen for reviews having around .001 % more content of **Procedual Content Generation of Maps** and .001 % less content of **Worth the Price** and correspondingly for the other topics.

In general, topics related to game-play details such as **Exploration and Discovery**, **Spaceship Travel and Combat**, **Survival and Challenge**, and **Procedurally Generated Universe and Exploration** are positively associated with the play hours. Some other topics such as **Moving and Looking** and **Material Collection**. Although lean to in average higher playing hours, the associations are not significant

When it comes to disappointments, the associations with play hours vary. **Lack of Interest and Reward** leans to higher play hours whereas **Strong Dislike, Waste of Money**, **Quick Disappointment**, and **Disappointment of Promise and Hype** is associated with lower playing hours. Other topics such as **Lack of Content/Grind**, **Lies and Missed Promises**, **Falling Short of Expectations**, and **Feels Unfinished, Indie vs AAA (Quality Level)** do not show either significant positive or negative association with the playing hours.

The variation can be also found in topics related to positive feelings. **Enjoyment Despite Flaws** and **Appreciation** are significantly associated with playing hours, on the other hand, **Appreciating Improvements** leans to lower playing hours.

The topics related to updates also show both directions of association with the playing hours. Topic prevalence of **Upgrades and Items** and **Updates and Added Content** are associated with higher playing hours whereas **Change of Games** and **Gradual Improvements** are negatively associated with the playing hours.

DISCUSSION

One worth mentioning phenomena is that the players' perceptions are indeed changing over time and the change can potentially affected by updates and of the game. There are extracted topics such as **Updates and Added Content**, **Changes of Game** and **Upgrades and Items** that are directly related to the game updates. Besides, the corresponding difference of topic prevalence between players who recommend the game and player who do not recommend the game shows that the overall evaluation can be either positive of negative once the player has experienced the updates.

Another phenomenon is how the playing hours affect the playing experiences that are reflected in the game reviews. The positive association of topics related to game-play details

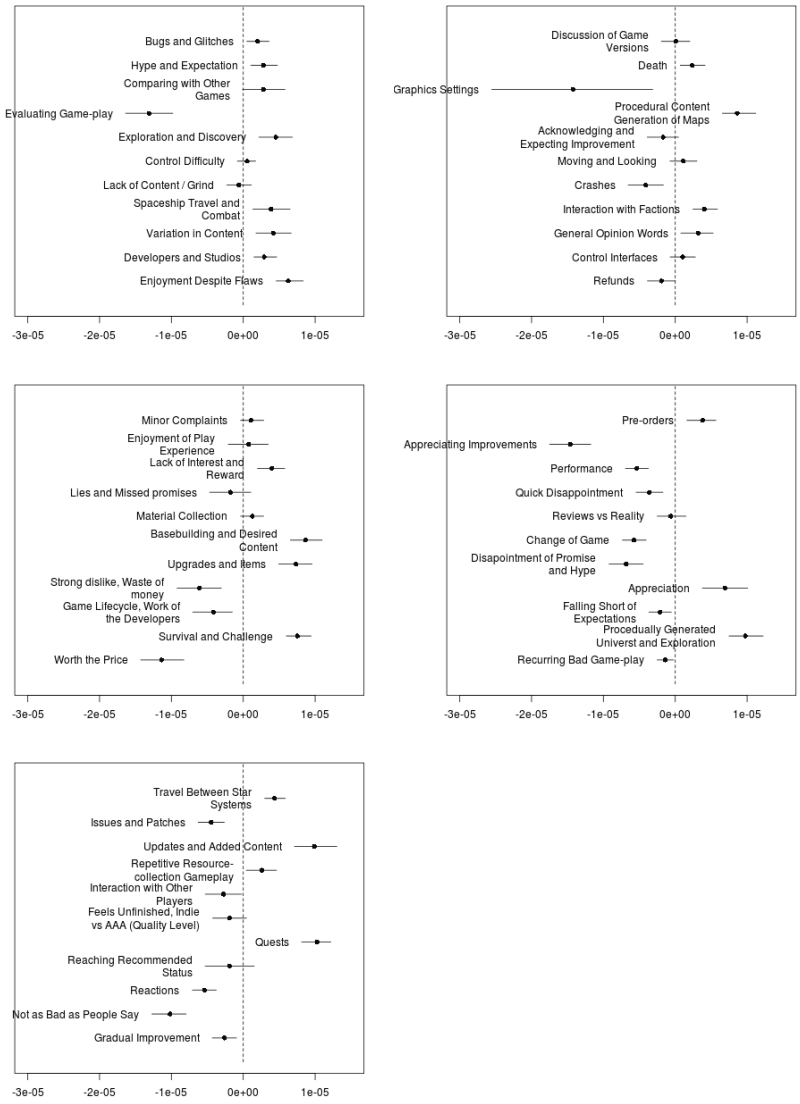

Figure 4: Influence of play hours on topical content of reviews.

with the play hours shows that, compared to other topics, it requires players to spend enough time on playing the game so that the game-play details can become a part of their experience and can be written in the reviews. The association of the topic **Lack of Interest and**

Reward with higher playing hours can potentially reflect the opinions from the players who were not satisfied even after updates. The association of the topics **Strong Dislike, Waste of Money**, **Quick Disappointment**, and **Disappointment of Promise and Hype** with lower playing hours can result from the players who were disappointed in the beginning when the game was launched and chose to complain about the game in their reviews or even gave up to play the game.

The updates have played an important role in the life-cycle of *No Man's Sky* especially after the game was launched. Some of them did potentially affect the opinions of players (e.g. topic **Survival and Challenge**) in a positive way. However, our analysis showed that the updates do not always bring positive feedback. Apparently the players had differnt perceptions to topics **Upgrades and Items**, **Updates and Added Content**, **Change of Games**, and **Change of Gradual Improvements**. Besides, the update related to the topic **Interact with Other Players** especially the multiplayer feature is one example. Some players felt even ´´MIND BLOWING" (see the quote in Section) but some players were not satisfied and left a negative review.

CONCLUSIONS, LIMITATIONS AND OPPORTUNITIES

In this research, an analysis of a large collection of over 85000 game reviews of the game *No Man's Sky* is conducted. The results reveal a large variety of topics that were discussed by players, answering RQ1; the results also reveal clear temporal dynamics of topic prevalence over time, answering RQ2; the results also revealed differences of temporal dynamics between reviews that recommended the game and reviews that did not, answering RQ3; and the results further revealed how such temporal changes coincided with updates to the game, with concrete examples how the updates can potentially affect the discussions, answering RQ4.

The reviews were collected from the Steam platform. Despite the large amount of reviews we were able to gather, reviews from this platform can only reflect the opinions of PC players. Some findings of this research might be applicable directly to players in other platforms such as PlayStation. However, some platform-specific topics, especially technical related topics, might not be appropriate to be imposed on players in other platforms.

This work can be beneficial not only for researchers to study and model players' expectations of game content and reactions to game releases and updates, but also for game industry practitioners when it comes to maintaining players' perceptions; game companies can draw insights from the issues and reactions of players found in this work if the release of another game leads to a similar situation as in *No Man's Sky*.

ACKNOWLEDGMENTS

This work was supported by Academy of Finland decisions 312395, 313748 and 327352.

ENDNOTES

1 https://nomanssky.gamepedia.com/Patch_notes

2 https://scrapy.org/

3 https://www.crummy.com/software/BeautifulSoup/

BIBLIOGRAPHY

Alves, Carina, Geber Ramalho, and Alexandre Damasceno. 2007. "Challenges in requirements engineering for mobile games development: The meantime case study." In *15th IEEE International Requirements Engineering Conference (RE 2007),* 275–280. IEEE.

Ampatzoglou, Apostolos, and Ioannis Stamelos. 2010. "Software engineering research for computer games: A systematic review." *Information and Software Technology* 52 (9): 888–901.

Blei, David M, and John D Lafferty. 2006. "Dynamic topic models." In *Proceedings of the 23rd international conference on Machine learning,* 113–120. ACM.

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. "Latent dirichlet allocation." *Journal of machine Learning research* 3 (Jan): 993–1022.

Burnett, Margaret, Curtis Cook, and Gregg Rothermel. 2004. "End-user software engineering." *Communications of the ACM* 47 (9): 53–58.

Chang, Jonathan, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber, and David M Blei. 2009. "Reading tea leaves: How humans interpret topic models." In *Advances in neural information processing systems,* 288–296.

Chen, Ning, Jialiu Lin, Steven CH Hoi, Xiaokui Xiao, and Boshen Zhang. 2014. "AR-miner: mining informative reviews for developers from mobile app marketplace." In *Proceedings of the 36th International Conference on Software Engineering,* 767–778. ACM.

Cho, Yooncheong, Il Im, Roxanne Hiltz, and Jerry Fjermestad. 2002. "An analysis of online customer complaints: implications for web complaint management." In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences,* 2308–2317. IEEE.

Ciurumelea, Adelina, Andreas Schaufelbühl, Sebastiano Panichella, and Harald C Gall. 2017. "Analyzing reviews and code of mobile apps for better release planning." In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER),* 91–102. IEEE.

Fu, Bin, Jialiu Lin, Lei Li, Christos Faloutsos, Jason Hong, and Norman Sadeh. 2013. "Why people hate your app: Making sense of user feedback in a mobile app store." In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining,* 1276–1284. ACM.

Greer, Des, and Guenther Ruhe. 2004. "Software release planning: an evolutionary and iterative approach." *Information and software technology* 46 (4): 243–253.

Guzman, Emitza, and Walid Maalej. 2014. "How do users like this feature? a fine grained sentiment analysis of app reviews." In *2014 IEEE 22nd international requirements engineering conference (RE),* 153–162. IEEE.

Hello Games. 2016. *No Man's Sky.* [PlayStation 4, Microsoft Windows, Xbox One] Hello Games, Guildford, UK.

Holzer, Adrian, and Jan Ondrus. 2011. "Mobile application market: A developer's perspective." *Telematics and informatics* 28 (1): 22–31.

Kang, Ha-Na, Hye-Ryeon Yong, and Hyun-Seok Hwang. 2017. "A Study of Factors Influencing Helpfulness of Game Reviews: Analyzing STEAM Game Review Data." *Journal of Korea Game Society* 17 (3): 33–44.

Kanode, Christopher M, and Hisham M Haddad. 2009. "Software engineering challenges in game development." In *2009 Sixth International Conference on Information Technology: New Generations,* 260–265. IEEE.

Kasurinen, Jussi, Andrey Maglyas, and Kari Smolander. 2014. "Is requirements engineering useless in game development?" In *International Working Conference on Requirements Engineering: Foundation for Software Quality,* 1–16. Springer.

Ko, Andrew J, Robin Abraham, Laura Beckwith, Alan Blackwell, Margaret Burnett, Martin Erwig, Chris Scaffidi, Joseph Lawrance, Henry Lieberman, Brad Myers, et al. 2011. "The state of the art in end-user software engineering." *ACM Computing Surveys (CSUR)* 43 (3): 21.

Li, Xiaozhou, Zheying Zhang, and Kostas Stefanidis. 2018. "Mobile App Evolution Analysis Based on User Reviews." In *The 17th International Conference on Intelligent Software Methodologies, Tools, and Techniques,* 773–786.

Lim, Chong-U, and D Fox Harrell. 2014. "Developing Social Identity Models of Players from Game Telemetry Data." In *AIIDE.*

Lin, Dayi, Cor-Paul Bezemer, and Ahmed E Hassan. 2017. "Studying the urgent updates of popular games on the steam platform." *Empirical Software Engineering* 22 (4): 2095–2126.

———. 2018. "An empirical study of early access games on the Steam platform." *Empirical Software Engineering* 23 (2): 771–799.

Lin, Dayi, Cor-Paul Bezemer, Ying Zou, and Ahmed E Hassan. 2019. "An empirical study of game reviews on the Steam platform." *Empirical Software Engineering* 24 (1): 170–207.

Lu, Chien, Jaakko Peltonen, and Timo Nummenmaa. 2019. "Game postmortems vs. developer Reddit AMAs: computational analysis of developer communication." In *Proceedings of the 14th International Conference on the Foundations of Digital Games,* 1–7.

Lu, Chien, Jaakko Peltonen, Timo Nummenmaa, Xiaozhou Li, and Zheying Zhang. 2020. "What Makes a Trophy Hunter? An Empirical Analysis of Reddit Discussions." In *International GamiFIN conference.* In press.

Mimno, David, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. 2011. "Optimizing semantic coherence in topic models." In *Proceedings of the conference on empirical methods in natural language processing,* 262–272. Association for Computational Linguistics.

Nayebi, Maleknaz, Bram Adams, and Guenther Ruhe. 2016. “Release Practices for Mobile Apps–What do Users and Developers Think?” In *2016 ieee 23rd international conference on software analysis, evolution, and reengineering (saner),* 1:552–562. IEEE.

Newman, James. 2012. “Ports and patches: Digital games as unstable objects.” *Convergence* 18 (2): 135–142.

Nummenmaa, Timo, Annakaisa Kultima, Kati Alha, and Tommi Mikkonen. 2013. “Applying Lehman’s Laws to Game Evolution.” In *Proceedings of the 2013 International Workshop on Principles of Software Evolution,* 11–17. IWPSE 2013. Saint Petersburg, Russia: ACM. ISBN: 978-1-4503-2311-6. `http://doi.acm.org/10.1145/2501543.2501546`.

Palomba, Fabio, Mario Linares-Vasquez, Gabriele Bavota, Rocco Oliveto, Massimiliano Di Penta, Denys Poshyvanyk, and Andrea De Lucia. 2015. “User reviews matter! tracking crowdsourced reviews to support evolution of successful apps.” In *2015 IEEE international conference on software maintenance and evolution (ICSME),* 291–300. IEEE.

Roberts, Margaret E, Brandon M Stewart, and Edoardo M Airoldi. 2016. “A model of text for experimentation in the social sciences.” *Journal of the American Statistical Association* 111 (515): 988–1003.

Ruhe, Günther, and Moshood Omolade Saliu. 2005. “The science and practice of software release planning.” *IEEE Software* 2005:1–10.

Saliu, Omolade, and Guenther Ruhe. 2005. “Supporting software release planning decisions for evolving systems.” In *29th Annual IEEE/NASA Software Engineering Workshop,* 14–26. IEEE.

Scalabrino, Simone, Gabriele Bavota, Barbara Russo, Massimiliano Di Penta, and Rocco Oliveto. 2017. “Listening to the crowd for the release planning of mobile apps.” *IEEE Transactions on Software Engineering* 45 (1): 68–86.

Sifa, Rafet, Christian Bauckhage, and Anders Drachen. 2014. “The Playtime Principle: Large-scale cross-games interest modeling.” In *CIG,* 1–8.

Sifa, Rafet, Anders Drachen, and Christian Bauckhage. 2015. “Large-scale cross-game player behavior analysis on steam.” *Borderlands* 2:46–378.

Slivar, Ivan, Mirko Suznjevic, and Lea Skorin-Kapov. 2015. “The impact of video encoding parameters and game type on QoE for cloud gaming: A case study using the steam platform.” In *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX),* 1–6. IEEE.

Stamolampros, Panagiotis, Nikolaos Korfiatis, Panos Kourouthanassis, and Efthymia Symitsi. 2019. “Flying to quality: Cultural influences on online reviews.” *Journal of Travel Research* 58 (3): 496–511.

Steamworks Documentation: Early Access. Accessed 8 Dec 2019. `https://partner.steamgames.com/doc/store/earlyaccess`.

Svahnberg, Mikael, Tony Gorschek, Robert Feldt, Richard Torkar, Saad Bin Saleem, and Muhammad Usman Shafique. 2010. “A systematic review on strategic release planning models.” *Information and software technology* 52 (3): 237–248.

Toivonen, Saara, and Olli Sotamaa. 2010. “Digital distribution of games: the players’ perspective.” In *Proceedings of the International Academic Conference on the Future of Game Design and Technology,* 199–206. ACM.

Villarroel, Lorenzo, Gabriele Bavota, Barbara Russo, Rocco Oliveto, and Massimiliano Di Penta. 2016. “Release planning of mobile apps based on user reviews.” In *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE),* 14–24. IEEE.

Wu, Yingcai, Furu Wei, Shixia Liu, Norman Au, Weiwei Cui, Hong Zhou, and Huamin Qu. 2010. “OpinionSeer: interactive visualization of hotel customer feedback.” *IEEE transactions on visualization and computer graphics* 16 (6): 1109–1118.

PUBLICATION III

Probabilistic Dynamic Non-negative Group Factor Model for Multi-source Text Mining

Chien Lu, Jaakko Peltonen, Jyrki Nummenmaa, and Kalervo Järvelin

In: *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*. Ed. by Mathieu d'Aquin et al. ACM, 2020, pp. 1035–1043

Probabilistic Dynamic Non-negative Group Factor Model for Multi-source Text Mining

Chien Lu
Tampere University
Tampere, Finland
chien.lu@tuni.fi

Jaakko Peltonen
Tampere University
Tampere, Finland
jaakko.peltonen@tuni.fi

Jyrki Nummenmaa
Tampere University
Tampere, Finland
jyrki.nummenmaa@tuni.fi

Kalervo Järvelin
Tampere University
Tampere, Finland
kalervo.jarvelin@tuni.fi

ABSTRACT

Nonnegative matrix factorization (NMF) is a popular approach to model data, however, most models are unable to flexibly take into account multiple matrices across sources and time or apply only to integer-valued data. We introduce a probabilistic, Gaussian Process based, more inclusive NMF-based model which jointly analyzes nonnegative data such as text data word content from multiple sources in a temporal dynamic manner. The model collectively models observed matrix data, source-wise latent variables and their dependencies and temporal evolution with a full-fledged hierarchical approach including flexible nonparametric temporal dynamics. Experiments on simulated data and real data show the model outperforms comparable models. A case study on social media and news demonstrates the model discovers semantically meaningful topical factors and their evolution.

CCS CONCEPTS

• **Computing methodologies** → **Non-negative matrix factorization**; **Natural language processing**.

KEYWORDS

Nonnegative Matrix Factorization; Gaussian Process; Multiple Sources

ACM Reference Format:
Chien Lu, Jaakko Peltonen, Jyrki Nummenmaa, and Kalervo Järvelin. 2020. Probabilistic Dynamic Non-negative Group Factor Model for Multi-source Text Mining. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3340531.3411956

1 INTRODUCTION

Factor analysis is a popular approach to extract latent components describing variable relationships within data sets, and non-negative matrix factorization (NMF) [14, 18] in particular has become a prominent solution for data sets in matrix form, applicable in numerous settings where measurements and their latent factors are expected to be nonnegative, such as in several text analytics and bioinformatics settings. However, much factor analysis work has focused on factorization of individual matrices.

Analyzing data from multiple sources has attracted increasing attention in the machine learning community [11]. For instance text data such as online discussions or news articles from a single source may not provide a sufficiently thorough understanding of the underlying phenomena. Analyzing the factors underlying data matrices from multiple sources jointly is a promising approach to infer improved models that better represent the phenomena, have better predictive performance, and allow discovery of relations and interactions between different sources.

In addition to multiple sources, modeling temporal variation of the phenomena from data collected over time is also often desired. Models including flexible generative approaches such as Gaussian Processes (GPs) [19] and their extensions have been proposed to model temporal dynamics. Temporal analysis should ideally reveal both variation of the underlying factor prevalences and variation of the factors' contents over time.

Although NMF has been widely accepted as a classical approach when analyzing text data, to our knowledge there are only few probabilistic matrix-factorization models that address the multiple sources aspect or the temporal aspect, and none that address both.

We introduce a novel probabilistic non-negative matrix factorization model, suitable for analysis of multiple data matrices across sources and time, applicable to any series of nonnegative real-valued matrices. The proposed method models the matrix data, the underlying source-wise parameters of factor prevalence and content, and inter-source parameters of factor relationships across sources. Temporal dynamics of topic prevalence, topic content and source-source interaction are modeled with a flexible (Hierarchical) Gaussian Process Latent Variable Model (GPLVM) [12, 13, 15] based approach. Modeling temporal dynamics with GP priors can model smooth temporal changes without fixing a rigid parametric form [8]. We carry out variational inference for the model.

The model has superior performance in experiments in predicting held-out data. We demonstrate the model both on simulated data and a case study on news and social media. We use a text analytics case for simplicity of illustrating results, but the model

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10.
https://doi.org/10.1145/3340531.3411956

applies to all similar domains with non-negative data and is not restricted e.g. to integer-valued count data, unlike some text analysis solutions.

The rest of the paper is organized as follows. Next, Section 2 describes preliminaries and related work. Sections 3 and 4 present the basic structure of the proposed model and its variational inference. Sections 5 and 6 describe the experiments with simulated and real data. Section 7 provides conclusions and discussion.

2 RELATED BACKGROUND

Non-negative Matrix Factorization. NMF is a widely used data analysis approach in domains such as bioinformatics [23], image processing [14], and text mining [16]. In short, NMF finds an approximate decomposition of a $N \times D$ matrix $\mathbf{X}$ containing only nonnegative element values into a product of two lower-rank matrices $\mathbf{X} \approx \mathbf{Z}\mathbf{W}^\top$ where $\mathbf{Z}$ is a $N \times K$ matrix, $\mathbf{W}$ is a $D \times K$ matrix, and K is the number of latent factors, where $\mathbf{Z}$ and $\mathbf{W}$ also contain only nonnegative values. For example in text analytics $\mathbf{X}$ may be a term-document matrix of N terms and D documents, $\mathbf{W}$ can be interpreted as a topic loading matrix of K topics of D documents, so that each row $\mathbf{w}_d$ contains the topic loadings for document d, and $\mathbf{Z}$ can be interpreted as a topic content matrix of N terms across the K topics, each column $\mathbf{z}_{.k}$ is a discrete distribution over terms for topic k. Different NMF variants use different divergences to measure difference between $\mathbf{X}$ and its approximation $\mathbf{Z}\mathbf{W}^\top$ and regularize $\mathbf{Z}$ and $\mathbf{W}$ by different penalties. We adopt the form where the model is specified by a particular noise model between $\mathbf{Z}\mathbf{W}^\top$ and the observed $\mathbf{X}$ and particular priors for $\mathbf{Z}$ and $\mathbf{W}$; the latter incorporate a hierarchical model for cross-sources and temporal dynamics.

Related Work. Some NMF based methods have been proposed to model temporal dynamics [21, 24] of text data or data from multiple sources [4, 7, 22]; most of these are not hierarchical approaches or deal with only one of the two aspects (multi-source or temporal). For example, in the Joint Past-Present Decomposition Model (JPP; [24]) at each time slice the term-document matrix is explained by both current topics and topics at the previous time slice.

A noteworthy example of Matrix factorization approaches is Bayesian Group Factor Analysis (GFA) [10, 25] which analyzes data from multiple sources (groups). GFA considers the joint data set $\mathbf{Y} = \{\mathbf{X}_1, \ldots, \mathbf{X}_M\}$ of matrices $\mathbf{X}_1 \in \mathbb{R}^{N \times D_1}, \ldots, \mathbf{X}_M \in \mathbb{R}^{N \times D_M}$. GFA factorizes $\mathbf{Y}$ into matrices $\mathbf{Z}$ and $\mathbf{W}$ as $\mathbf{Y} \approx \mathbf{Z}\mathbf{W}^\top$ where $\mathbf{W} = [\mathbf{W}_1^\top \ldots \mathbf{W}_M^\top]^\top$, $\mathbf{W}_m \in \mathbb{R}^{D_m \times K}$ and each element $w_{m,k}(d)$ in $\mathbf{W}_m$ is normally distributed with zero mean and a group-wise precision parameter $\alpha_{m,k}$ as $w_{m,k}(d) \sim N(0, \alpha_{m,k}^{-1})$. The precision parameter $\alpha_{m,k}$ enables GFA to model shared underlying features between groups. However, GFA has no model for temporal dynamics. Moreover, GFA is not designed to model non-negative factorization and hence it can yield negative-valued factors even for nonnegative-valued data, making it unsuitable to be directly applied in cases when factors are required to be nonnegative e.g. for interpretability, such as loadings and contents of topics in text data. We use GFA as a comparison both as is and with a simple correction for nonnegativity.

One similar work [9] tries to model the temporal dynamics but only takes the dynamics of the left-hand side matrix $\boldsymbol{Z}$ into account, the loading matrix $\boldsymbol{W}$ is considered static.

Another group of approaches are the models based on Poisson factor analysis (PFA) [1, 6, 17, 28]. However, since the Poisson distribution only models positive integers, the approaches only model positive-integer-valued matrices but not positive real-valued matrices; the latter occur in many domains including text mining, e.g. real-valued term weighting such as TF-IDF is often crucial for document representation. This paper focuses on methods applicable to positive real-valued matrices.

3 PROPOSED MODEL

We now present the proposed dynamic non-negative Bayesian group factor (DNBGFA) model. For clarity we use text data terminology (documents, terms, topics) but the model is general. DNBGFA considers a temporal sequence of T term-document matrices $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots, \mathbf{X}^{(T)}$, sharing the same vocabulary of N terms (words). For each time slice t, $\mathbf{X}^{(t)} = [\mathbf{X}_1^{(t)}, \ldots \mathbf{X}_M^{(t)}]$ is a combined matrix of M text sources, each $\mathbf{X}_m^{(t)}$ contains N terms and $D_m^{(t)}$ documents, and the total document count at time t is $D^{(t)} = \sum_m D_m^{(t)}$.

For each time slice t, the task is to approximately factorize the $N \times D^{(t)}$ term-document matrix $\mathbf{X}^{(t)}$ as

$$\mathbf{X}^{(t)} \approx \mathbf{Z}^{(t)} \mathbf{W}^{(t)\top} \tag{1}$$

where $\mathbf{Z}^{(t)}$ is a $N \times K$ matrix which represents the topic content and $\mathbf{W}^{(t)}$, a $D^{(t)} \times K$ matrix, represents the topic prevalence, and both matrices are nonnegative. The setup is illustrated in Figure 1. We infer the factorization as part of a hierarchical generative model for the data.

The graphical plate model representation of the model is shown in Figure 2. We assume a truncated-Gaussian likelihood where each Gaussian is truncated from below at 0 as is appropriate for nonnegative data, so that

$$p(\mathbf{X}^{(t)} | \mathbf{Z}^{(t)}, \mathbf{W}^{(t)}) = \prod_{n,d} N^+ \left(x_{n,d}^{(t)} | \mathbf{z}_n^{(t)\top} \mathbf{w}_d^{(t)}, \sigma^2 \right) \tag{2}$$

where $\mathbf{w}_d^{(t)}$ denotes the dth column of $\mathbf{W}^{(t)}$ representing the topic prevalence in document d, $\mathbf{z}_n^{(t)}$ denotes the nth row of $\mathbf{Z}^{(t)}$ representing the weight of the nth vocabulary word across the topics The σ^2 controls the noisiness of the observations. We set it equal for every document in the following implementations but one can also take the advantage of the flexibility to make a more sophisticated model if needed. For example, a more detailed document-specific variance σ_d^2 representing a source-specific noise parameter can be assigned as $\sigma_d = \sigma_{m(d)}$ where $m(d)$ denotes the group that document d belongs to.

The key idea is to generate the factor matrices in a way that flexibly ties them over time, topics, and sources, without restricting the time dependency to a pre-given form; we generate the dependencies (covariance matrices) as functions of latent variables that are draws from flexible nonparametric time series models, as detailed in Sections 3.1 and 3.2.

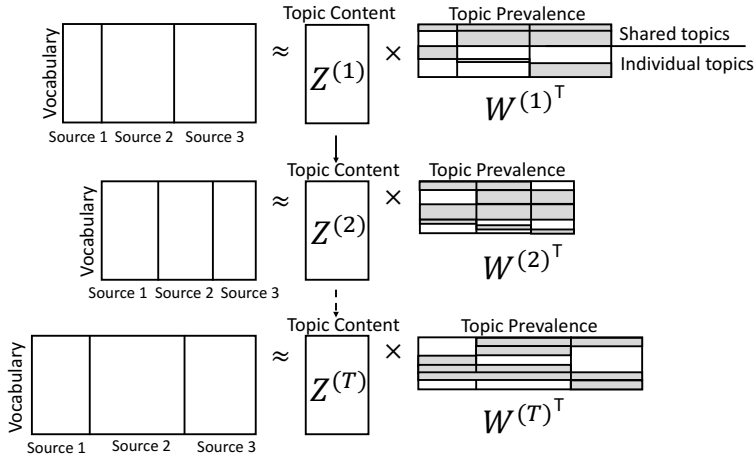

Figure 1: Illustration of the DNBGFA model. A sequence of non-negative matrices $\mathbf{X}^{(1)}, ..., \mathbf{X}^{(T)}$ is factorized into $\mathbf{Z}^{(1)}, ..., \mathbf{Z}^{(T)}$ and $\mathbf{W}^{(1)}, ..., \mathbf{W}^{(T)}$ while modeling temporal dependencies of factors.

3.1 Topic Content

To enforce non-negativity of the topic content matrix, each element $z_{k,n}^{(t)}$ of $\mathbf{Z}^{(t)}$ is parameterized by a softmax transformation

$$z_{k,n}^{(t)} = \frac{\exp(\eta_{k,n}^{(t)})}{\sum_{n'=1}^{N} \exp(\eta_{k,n'}^{(t)})} \tag{3}$$

which ensures the summation of word proportions of each topic $\sum_{n'=1}^{N} z_{k,n'}^{(t)}$ is equal to 1. Note that we will model magnitude of numbers in observed matrices by the loading matrices, hence we can without loss of generality fix the sums as above. Similar transformations are often used in text mining models [3, 20].

GPLVM based model. For each term n, the variable $\boldsymbol{\eta}_n = [\eta_{1,n}^{(1)} \cdots \eta_{K,n}^{(1)} \cdots \eta_{1,n}^{(T)} \cdots \eta_{K,n}^{(T)}]^\top$ controls topic content and the dependencies between its elements represent dependencies across sources and time. We model them in a nonparametric approach by a Gaussian process latent variable model (GPLVM) which lets us model temporal dynamics in a flexible way. In a GPLVM, the parameters of a Gaussian distribution are constructed by a draw from another GP :

$$\left[\eta_{1,n}^{(1)} \eta_{1,n}^{(2)} \cdots \eta_{1,n}^{(T)} \cdots \eta_{K,n}^{(1)} \cdots \eta_{K,n}^{(T)}\right]^\top \sim \mathbf{N}\left(\mathbf{0}, \Sigma_\eta\right) \tag{4}$$

where

$$\Sigma_{\boldsymbol{\eta}} = \mathcal{K}_{\boldsymbol{\eta}} + \epsilon_{\boldsymbol{\eta}} \mathbf{I} \tag{5}$$

and $\mathcal{K}_{\boldsymbol{\eta}}$ consists of elements

$$\mathcal{K}_{k,l}^{(\eta)}(t_i, t_j) = k_0^{(\eta)}(t_i, t_j)\delta_{k,l} + k_{k,l}^{(\eta)}(t_i, t_j) \tag{6}$$

where k, l are topic indices and $k_0^{(\eta)}(t_i, t_j)\delta_{k,l}$ is a kernel function which governs the within topic consistency over time, $k_{k,l}^{(\eta)}(t_i, t_j)$ governs the topic-topic interaction, and $\epsilon_{\boldsymbol{\eta}}$ controls noisiness.

The kernel $k_0^{(\eta)}$ can be formed by an arbitrary kernel function of time slices t_i, $i = 1, \ldots, T$. In this paper, we use RBF kernel which is defined as

$$rbf_{(\xi,\iota)}(t_i, t_j) = \iota^2 \times e^{\frac{-||t_i - t_j||^2}{\xi^2}} \tag{7}$$

where hyperparameters ξ and ι control dependencies over time. This is a nonparametric time series model for the changing of the term n over time in topic k. As in GPs, no specific functional form is assumed for behavior over time, only that values at similar time points are correlated as described.

$k_{k,l}^{(\eta)}(t_i, t_j)$ controls topic-topic interactions not only within a time-slice but also across two different time-slices. We construct it as

$$k_{k,l}^{(\eta)}(t_i, t_j) = e^{-\lambda_\eta |t_i - t_j|} r_k^{(t_i)} r_l^{(t_j)} \tag{8}$$

which consists of an exponential time decay term $\lambda_\eta \sim Gamma(a, b)$ and products $r_k^{(t_i)} r_l^{(t_j)}$ that control topic-topic interactions of topics k and l across time in a more flexible way: for each topic the vector $\mathbf{r}_k = [r_k^{(t_1)}, \ldots, r_k^{(t_T)}]^\top$ is drawn as a realization of a GP as

$$\mathbf{r}_k \sim GP(\mathbf{0}, \Sigma_{\mathbf{r}}) \quad , \quad \Sigma_{\mathbf{r}} = \mathcal{K}_{\mathbf{r}} + \epsilon_r \mathbf{I} \tag{9}$$

where $\mathcal{K}_{\mathbf{r}}$ consists of elements

$$\mathcal{K}^{(r)}(t_i, t_j) = k_0^{(r)}(t_i, t_j) \tag{10}$$

and ϵ_r controls noisiness. Large values of the product $r_k^{(t_i)} r_l^{(t_j)}$ strengthen the dependency $k_{k,l}^{(\eta)}(t_i, t_j)$ between two time slices

whereas small values of the product decrease the dependency, allowing new topic content to emerge.

Like $k_0^{(\eta)}$, the kernel $k_0^{(r)}$ can be computed given time slices t_i, $i = 1, \ldots, T$ with an RBF kernel shown in equation (7). Noisiness variables ϵ_η and ϵ_r could be given priors or be used as hyperparameters, we did the latter for simplicity.

The kernel $k_0^{(\eta)}$ is identical in different topics, hence it acts as a regularizing term controlling word (dis)similarities within topics over time whereas $k_{k,l}^{(\eta)}$ models flexibility of topic-topic interactions.

3.2 Topic Prevalence

Similar to the topic content model, to enforce non-negativity, each $w(d)_{m,k}^{(t)}$ of $\mathbf{W}^{(t)}$ is sampled from a truncated normal distribution with mean 0 and a source-wise variance $e^{\alpha_{m,k}^{(t)}}$:

$$w(d)_{m,k}^{(t)} \sim N^{+}(0, e^{\alpha_{m,k}^{(t)}}) \ . \tag{11}$$

The source-wise latent variables $\alpha_{m,k}^{(t)}$ which control the sparsity of topic in data sources m and time slices t are again a realization of a GPLVM

$$\left[\alpha_{1,k}^{(1)} \ldots \alpha_{1,k}^{(T)}, \ldots, \alpha_{M,k}^{(1)} \ldots \alpha_{M,k}^{(T)}\right]^{\top} \sim \mathbf{N}(\mathbf{0}, \Sigma_{\boldsymbol{\alpha}}) \tag{12}$$

where

$$\Sigma_{\boldsymbol{\alpha}} = \mathcal{K}_{\boldsymbol{\alpha}} + \epsilon_{\boldsymbol{\alpha}}\mathbf{I} \tag{13}$$

and $\mathcal{K}_{\boldsymbol{\alpha}}$ consists of elements

$$\mathcal{K}_{m,n}^{(\alpha)}(t_i, t_j) = k_0^{(\alpha)}(t_i, t_j)\delta_{m,n} + k_{m,n}^{(\alpha)}(t_i, t_j) \tag{14}$$

where $k_0^{(\alpha)}(t_i, t_j)\delta_{m,n}$ is a kernel function governs the within source consistency of topic prevalence over time and $k_{m,n}^{(\alpha)}(t_i, t_j)$ governs the cross-source interactions.

The cross-source interactions $k_{m,n}^{(\alpha)}(t_i, t_j)$ are constructed as

$$k_{m,n}^{(\alpha)}(t_i, t_j) = e^{-\lambda_\alpha |t_i - t_j|} s_m^{(t_i)} s_n^{(t_j)} \tag{15}$$

where ϵ_α controls noisiness and the matrix is otherwise again composed of products of two terms, an exponential time decay term with decay variable $\lambda_\alpha \sim Gamma(c, g)$ and the products $s_m^{(t_i)} s_n^{(t_j)}$ that control correlation sources across time in a flexible manner, by generating for each source m the vector $\mathbf{s}_m = [s_m^{(1)}, \ldots, s_m^{(T)}]^{\top}$ from an independent GP as

$$\mathbf{s}_m \sim \mathbf{N}(\mathbf{0}, \Sigma_{\mathbf{s}}) \ , \ \ \Sigma_{\mathbf{s}} = \mathcal{K}_s + \epsilon_s\mathbf{I} \tag{16}$$

where $\mathcal{K}_{\mathbf{s}}$ consists of elements

$$\mathcal{K}^{(s)}(t_i, t_j) = k_0^{(s)}(t_i, t_j) \tag{17}$$

and ϵ_s controls noisiness.

As before, covariances $k_0^{(\alpha)}$ and $k_0^{(s)}$ are obtained by RBF kernel, whose hyperparameters control time depencency; we used RBF. For the noisiness parameters ϵ_α, and ϵ_s could again be given their own priors but for simplicity we kept them as hyperparameters.

The models of topic content and prevalence in the previous section and this section are highly analogous just like matrices $\mathbf{Z}^{(t)}$ and $\mathbf{W}^{(t)}$ have highly analogous roles. The differences are the different way to enforce non-negativity, and the different role of topics and sources: in the previous section correlations were modeled by GPLVMs for each term across topics and time slices, here correlations are modeled by GPLVMs for each topic across sources and time slices. This establishes a flexible framework for factorization of matrices related across sources and time. The factorizations at each time slice (i.e., the parameter posteriors) are learned based on both the hierarchical prior and the likelihood.

The hierarchical prior lets the model handle cases where at some time slices no documents belonging to a source exist; we test this in an experiment in Section 5. If a source is known to be inactive (not just missing) at some time slices, such as birth/death of sources, it can be specified into the priors e.g. by larger ϵ_α , if such expert knowledge is available.

Algorithm 1 Variational EM Procedure

Require:
$\mathbf{X}^{(1)} \ldots \mathbf{X}^{(T)}$: Observed matrices
K: number of topics
σ_d: Hyper-parameters (likelihood)
$k_0^{(\eta)}, k_0^{(r)}, \epsilon_r, \epsilon_\eta, a, b$: Hyper-parameters (content)
$k_0^{(\alpha)}, k_0^{(s)}, \epsilon_s, \epsilon_\alpha, c, g$: Hyper-parameters (prevalence)
Ensure:
for iter ← 1 to maxit **do**
  E-step: update $\boldsymbol{\eta}, \boldsymbol{\alpha}, \mathbf{W}$
  M-step: update r, s, $\lambda_\eta, \lambda_\alpha$
end for
return $\boldsymbol{\eta}, \boldsymbol{\alpha}, \mathbf{W}, \mathbf{r}, \mathbf{s}, \lambda_\alpha$ =0

4 VARIATIONAL INFERENCE

To deliver time-efficient inference, we derive variational inference algorithms. Approaches such as Gibbs sampling are possible, here we focus on the variational approach. The inference constructs a variational posterior distribution q for each parameter of interest; update rules for parameters of the q distributions are given below. We update the parameters in an EM manner, as shown in Algorithm 1. Inference algorithms are further described.

4.1 Topic Content Variables $\boldsymbol{\eta}$ and Topic Sparsity Sariables $\boldsymbol{\alpha}$

A Laplace's method based inference [26] is used. The variational distribution $q(\boldsymbol{\eta}_n^{(t)}) = \mathbf{N}(\boldsymbol{\eta}_n^{(t)} | \mathbf{m}_{\boldsymbol{\eta}_n^{(t)}}, -\nabla^2 f(\mathbf{m}_{\boldsymbol{\eta}_n^{(t)}})^{-1})$ where the mean $\mathbf{m}_{\boldsymbol{\eta}_n^{(t)}}$ is set to the value of the MAP solution which maximizes the joint log-probability f defined as

$$f(\boldsymbol{\eta}_n) = \sum_{(t)} E_{q(\mathbf{w})}\left[\log p(\mathbf{x}_n^{(t)} | \mathbf{z}_n^{(t)}, \mathbf{w}^{(t)})\right] + E_{q(\mathbf{r})}\left[\log p(\boldsymbol{\eta}_n^{(t)} | \mathbf{r})\right] \ . \tag{18}$$

In this work, we obtained the $\mathbf{m}_{\boldsymbol{\eta}_n^{(t)}} = \arg\max_{\boldsymbol{\eta}_n^{(t)}} f(\boldsymbol{\eta}_n^{(t)})$ using an optimizer called simulated annealing (SANN) [2]. The covariance matrix $\nabla^2 f(\mathbf{m}_{\boldsymbol{\eta}_n^{(t)}})$ is the Hessian matrix of f evaluated at the point $\mathbf{m}_{\boldsymbol{\eta}_n^{(t)}}$.

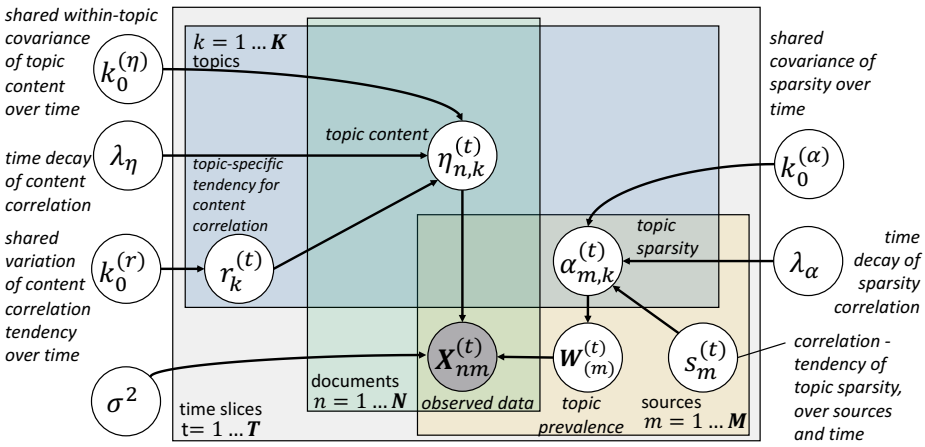

Figure 2: Graphical representation of the DNBGFA model. Noisiness parameters ϵ_η, ϵ_r, ϵ_s, ϵ_α not shown for clarity.

Inference of $\boldsymbol{\alpha}$ is similar, the variational distribution $q(\boldsymbol{\alpha}_k)$ is $\mathrm{N}(\boldsymbol{\alpha}_k|\mathbf{m}_{\boldsymbol{\alpha}_k}, -\nabla^2 f(\mathbf{m}_{\boldsymbol{\alpha}_k})^{-1})$ and the corresponding objective function $f(\boldsymbol{\alpha}_k)$ is

$$f(\boldsymbol{\alpha}_k) = \sum_{(t)} E_{q\left(\mathbf{w}_k^{(t)}\right)}\left[\log p(\mathbf{w}_k^{(t)}|\boldsymbol{\alpha}_k)\right] + E_{q(\mathbf{s})}\left[\log p(\boldsymbol{\alpha}_n^{(t)}|\mathbf{s})\right] \quad (19)$$

4.2 Topic Content Correlation r and Sparsity Correlation Tendencies s

To carry out the posterior inference of the variables $\mathbf{r}$ and $\mathbf{s}$ describing the topic-specific content and source-wise sparsity correlation tendencies, we adapted a recently developed framework proposed by Damianou et al. [5] which is able to capture the complexity of the interactions between latent variables. In the framework, auxiliary variables $\mathbf{u}^{(r)}$ and $\mathbf{r}_u$ are induced. The joint probability related to $\mathbf{r}$ is then expanded, written as

$$\prod_{n=1}^{N} p(\boldsymbol{\eta}_n|\mathbf{u}_n^{(r)}, \mathbf{r}, \mathbf{r}_u) p(\mathbf{u}_n^{(r)}|\mathbf{r}_u) p(\mathbf{r}) \quad (20)$$

where $p(\boldsymbol{\eta}_n|\mathbf{u}_n^{(r)}, \mathbf{r}, \mathbf{r}_u) = \mathrm{N}(\boldsymbol{\eta}_n|\mathbf{a}_\mathbf{n}, \Sigma_\eta^*)$ with $\mathbf{a}_\mathbf{n} = \mathcal{K}_{\eta u}{\mathcal{K}_u^{(r)}}^{-1}\mathbf{u}_n$ and $\Sigma_\eta^* = \Sigma_\eta - \mathcal{K}_{\eta u}{\mathcal{K}_u^{(r)}}^{-1}\mathcal{K}_{u\eta}$. The pseudo-inputs $\mathbf{r}_u = [r_u^{(1)}, \ldots, r_u^{(T)}]^\top$ are the constructing variables of $\mathbf{u}^{(r)}$, that is,

$$p(\mathbf{u}_n^{(r)}|\mathbf{r}_u) = \mathrm{N}(\mathbf{u}_n^{(r)}|\mathbf{0}, \mathcal{K}_u^{(r)}), \quad (21)$$

where $\mathcal{K}_u^{(r)}$ consists of elements

$$\mathcal{K}_u^{(r)}(t_i, t_j) = e^{-\lambda_\eta|t_i - t_j|} r_u^{(t_i)} r_u^{(t_j)} + k_0^{(\eta)}(t_i, t_j)\,. \quad (22)$$

The posterior is then approximated with

$$\prod_{n=1}^{N} p(\boldsymbol{\eta}_n|\mathbf{u}_n^{(r)}, \mathbf{r}', \mathbf{r}_u) q(\mathbf{u}_n^{(r)}) q(\mathbf{r}') \quad (23)$$

where $\mathbf{r}' = [r_1^{(1)} \ldots r_k^{(1)} \ldots r_1^{(T)} \ldots r_k^{(T)}]^\top$; $q(\mathbf{r}')$ is a Gaussian distribution $q(\mathbf{r}') = \mathrm{N}(\mathbf{r}'|\mathbf{m}_{\mathbf{r}'}, \mathrm{S}_{\mathbf{r}'})$ where the variational mean vector $\mathbf{m}_{\mathbf{r}'}$ and covariance matrix $\mathrm{S}_{\mathbf{r}'}$ are obtained via maximizing an objective function $\hat{\mathcal{F}}(\mathbf{r}') - KL\left(q(\mathbf{r}')||p(\mathbf{r}')\right)$ which is a Jensen's lower bound of the marginal likelihood, with respect to $\mathbf{m}_{\mathbf{r}'}$ and $\mathrm{S}_{\mathbf{r}'}$ together with $\mathbf{r}_u$.

We have

$$\hat{\mathcal{F}}(\mathbf{r}') = \frac{\sum_{n=1}^{N} \boldsymbol{\eta}_n^\top \mathbf{W}^{(r)} \boldsymbol{\eta}_n}{-2} + N \log\left(\frac{\epsilon_\eta^{-(K\times T)} |\mathcal{K}_u^{(r)}|^{\frac{1}{2}}}{(2\pi)^{\frac{(K\times T)}{2}} |\epsilon_\eta^{-2}\Psi_2^{(r)} + \mathcal{K}_u^{(r)}|^{\frac{1}{2}}}\right) + \frac{tr\left({\mathcal{K}_u^{(r)}}^{-1}\Psi_2^{(r)}\right) - \psi_0^{(r)}}{2\epsilon_\eta^2/N} \quad (24)$$

where the matrices involved are computed as

$$\mathbf{W}^{(r)} = \epsilon_\eta^{-2}\mathbf{I}_{(K\times T)} - \epsilon_\eta^{-4}\Psi_1^{(r)}\left(\epsilon_\eta^{-2}\Psi_2^{(r)} + \mathcal{K}_u^{(r)}\right)^{-1}{\Psi_1^{(r)}}^\top, \quad (25)$$

$$\psi_0^{(r)} = \mathbf{m}_{\mathbf{r}'}^\top \mathbf{m}_{\mathbf{r}'} + tr(\mathrm{S}_{\mathbf{r}'}), \quad (26)$$

$$\Psi_1^{(r)} = \mathbf{r}_u \mathbf{m}_{\mathbf{r}'}^\top \circ \mathbf{D}^{(\eta u)}, \quad (27)$$

$$\Psi_2^{(r)} = \mathbf{D}^{(u\eta)} \circ \mathbf{r}_u \left(\mathbf{m}_{\mathbf{r}'}\mathbf{m}_{\mathbf{r}'}^\top + Tr(\mathrm{S}_{\mathbf{r}'})\right)\mathbf{r}_u^\top \circ \mathbf{D}^{(\eta u)}, \quad (28)$$

$$\mathrm{S}_{\mathbf{r}'} = \left({\Sigma_{\mathbf{r}'}}^{-1} + diag(\xi_{\mathbf{r}'})\right)^{-1}, \quad (29)$$

where

$$\mathbf{D}^{(\eta u)} = \mathbf{1}_K \otimes \begin{bmatrix} 1 & \dots & e^{-|1-T|\lambda_\eta} \\ & \ddots & \\ e^{-|T-1|\lambda_\eta} & \dots & 1 \end{bmatrix} \tag{30}$$

and $\mathbf{D}^{(u\eta)} = \mathbf{D}^{(\eta u)\top}$. Note that $\circ$ denotes Hadamard product and $\otimes$ denotes Kronecker product.

For the parameters $\mathbf{s}$ which define the tendency of the topics' sparsity to correlate, the inference is done in a similar manner by imposing $\mathbf{u}^{(s)}$ and $\mathbf{s}_u$. The variational distribution $q(\mathbf{s}')$ related parameters $\{\mathbf{m}_{\mathbf{s}'}, \boldsymbol{\xi}_{\mathbf{s}'}, \mathbf{s}_u\}$ are obtained via optimizing the objective function $\hat{\mathcal{F}}(\mathbf{s}') - KL\,(q(\mathbf{s}')||p(\mathbf{s}'))$. The computation of $\hat{\mathcal{F}}(\mathbf{s}')$ is similar to the computation of (r') via replacing corresponding variables.

4.3 Time Decay Parameters λ_η and λ_α

Here we obtain the point estimates of λ_η and λ_α by optimizing the following objective functions:

$$f(\lambda_\eta) = E_{q(\boldsymbol{\eta})q(\mathbf{r})}\left[\sum_n \log p(\boldsymbol{\eta}_n|\mathbf{r}, \lambda_\eta)\right] + \log p(\lambda_\eta|a, b) \tag{31}$$

and

$$f(\lambda_\alpha) = E_{q(\boldsymbol{\alpha})q(\mathbf{s})}\left[\sum_k \log p(\boldsymbol{\alpha}_k|\mathbf{s}, \lambda_\alpha)\right] + \log p(\lambda_\alpha|c, g) \tag{32}$$

which can be done by standard optimizers, here we again use the SANN optimizer.

4.4 Topic Prevalence W

The truncated normal distribution preserves the Gaussian-Gaussian conjugacy, therefore, the variational distribution can can be obtained analytically:

$$q\left(\mathbf{w}_d^{(t)}\right) = \mathbf{N}^+(\mathbf{w}_d^{(t)}|\mathbf{m}_{\mathbf{w}_d}, \sigma^2\mathbf{S}_{\mathbf{w}_d}) \tag{33}$$

where we have

$$\mathbf{m}_{\mathbf{w}_d} = \mathbf{S}_{\mathbf{w}_d} E_q[\mathbf{Z}^{(t)\top}]\mathbf{x}_d^{(t)} \tag{34}$$

and

$$\mathbf{S}_{\mathbf{w}_d} = \left(E_q\left[\mathbf{Z}^{(t)\top}\mathbf{Z}^{(t)}\right]^{-1} + \Sigma^{-1}_{\alpha_d^{(t)}}\right)^{-1}. \tag{35}$$

5 SIMULATION EXPERIMENTS

We evaluate the proposed model both on simulated and on real data. We focus on cases where individual matrices are relatively small, so that good modeling assumptions become crucial for strong predictive performance. In this section we first compare the model with other approaches using artificial data in the same range as our collected data, simulated from an underlying DNBGFA model with $t = 1, ..., 10$, $N = 200$, each $D_m^{(t)} = 20$ and hyper-parameters: $k_0^{(\eta)} = k_0^{(\alpha)} = rbf_{(0.1,100)}$, $k_0^{(r)} = k_0^{(s)} = rbf_{(1,0.1)}$, $\epsilon_r = \epsilon_2 = 1$, $\epsilon_\eta = \epsilon_\alpha = 0.1$, $a = c = 1$, $b = g = 10$, $\sigma = 0.01$. The above RBF kernel parameters emphasize time dependency in the simulated data.

We compare the proposed method DNBGFA to six other methods: NMF, GFA and its variant denoted NGFA, JPP, and an integer-based method denoted DTM, as described below.

In these experiments as well as the case studies, the data are real-valued and we focus on comparing methods that are applicable to such real-valued data; therefore, NMF, GFA and JPP are selected as comparison methods designed for real-valued data. In contrast to the above methods, methods that are restricted to integers [6, 27, 28] are not readily applicable to real-valued data. We will compare to one such method, Dynamic Topic Model (DTM) [3] as a prominent example of integer-restricted dynamic methods; due to its restriction to integer data, DTM's model building is here based on integer-rounded observations. We compare performance of the methods in two scenarios below.

Partial Article. In this scenario, we simulate a situation where only partial content of articles are observed and we aim to predict the rest. A model built from the observed document parts is used to predict left-out content of the same documents. This scenario corresponds e.g. to using news RSS feed snippets to predict the news content, or using abstracts to predict the content of full-text research articles.

We simulate the scenario by leaving a randomly selected 10% of the content of each document vector $\mathbf{x}_d^{(t)}$ in the training data set. In detail, each column of the training term-document matrix $\mathbf{X}_{train}^{(t)}$ is generated by a multinomial draw. For each document vector $\mathbf{x}_d^{(t)}$ column (document) $\mathbf{x} = [x_1, \dots, x_N]^\top$ of the original matrix $\mathbf{X}^{(t)}$, denote the total term occurrence by $||\mathbf{x}||_1$ and the vector of term occurrence proportions by $\mathbf{x}/||\mathbf{x}||_1$; we fill the corresponding column of $X_{train}^{(t)}$ as the count vector of $0.10 \cdot ||\mathbf{x}||_1$ trials from the distribution $\mathbf{x}/||\mathbf{x}||_1$. The resulting training matrix contains 10% as many term occurrences as the original.

After training a model (DNBGFA, NMF, GFA, NGFA, DTM and JPP) to obtain the underlying topic content and topic prevalence matrices, the left-out term-document matrices of complete articles $X^{(t)}$ are then estimated by $\tilde{X^{(t)}} \approx Z_{train}^{(t)} W_{train}^{(t)\top} \times 10$, where the multiplier scales the prediction to the size of left-out data.

Interpolating Missing Data. In this scenario, we leave out the entire term-document matrix out of the 10 time slices, and we repeat the scenario 10 times leaving out a different time slice each time. The task is to estimate the missing slice given its time index and number of documents. For NMF, GFA and NGFA, the missing matrix is estimated using the result of the previous time slice, where topic loadings of an unseen document are estimated by average loadings. We have also tried to use the result from the next time slice and the performance is very similar. As DTM does not directly allow missing time slices we train it with the missing slice omitted and predict using the result from the previous time slice.

For DNBGFA, the matrices of the left-out time slice $Z^{(t)}$ and $W^{(t)}$ are directly estimated from the hierarchical model based on the time index of the held-out slice, thus the missing matrix $X^{(t)}$ can be directly estimated.

Results. For both scenarios, we repeat the process 20 times to account for stochasticity in data generation and in training methods, the root mean square error (RMSE) between predicted matrix content and true left-out content is employed as the performance

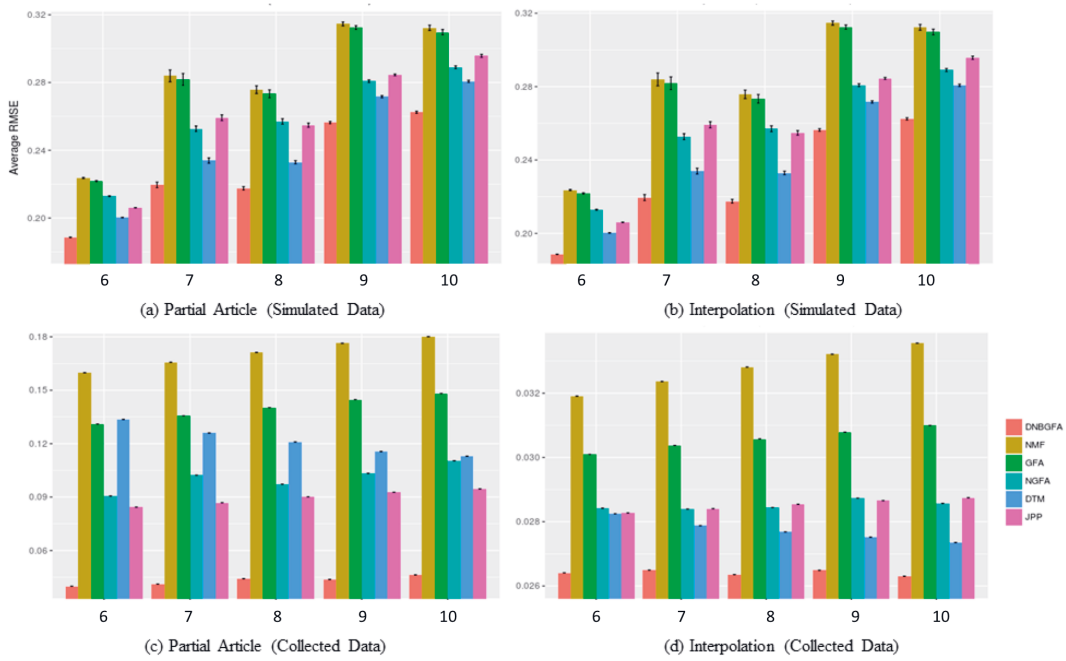

Figure 3: Performances are compared with averaged RMSE. Error bars are the variances of the mean value. DNBGFA attains the lowest average RMSE and outperforms other approaches in all four experiments (a)-(d), and for all cases over K (number of topic on the horizontal axis).

measure and pairwise t-tests between DNBGFA and other methods are then conducted to verify if the differences are statistically significant. Results can be found in Figure 3. In all cases DNBGFA achieves clearly smaller prediction error than other methods, and the differences between DNBGFA and other methods are statistically significant ($p < 0.01$).

6 CASE STUDY: FINNISH NEWS AND SOCIAL MEDIA

We apply the model to data from three text sources in 12 time slices (months) from September 2011 to August 2012, including *Helsingin Sanomat* (a Finnish newspaper), *Finnish Twitter Census* (www.finnishtwitter.com) and *Suomi24* (Finnish online forum; we take text from sections Talous (Economics) and Yhteiskunta (Society)). We remove stop-words and rare terms, lemmatize the text, then form TF-IDF weighted term-document matrices from the processed text.

Comparative Study. A comparison study is presented here, analogous to the two scenarios in Section 5 but with the above-mentioned data. For each experiment, we randomly sample 20 documents from each source and each time slice. The hyper-parameters are set as in the section 5 Results are shown in Figure 3. DNBGFA again outperforms other methods and differences are statistically significant.

Case Study: Exploratory Analysis of Topic Evolution. We further apply the proposed model to a subset of the above-mentioned dataset which contains the 150 longest documents from each time slice and each text source, yielding 5400 documents in total and 1286 terms after removing rare words and stop words. Figure 4 displays two example topics of the posterior analysis, showing their prevalence and topic content across time slices. The topic content evolution is extracted from the posterior of η (terms with highest loadings for each time slice are shown) and the prevalence is extracted from posterior of α (controlling ability of the topic to appear in documents; higher value yields higher chance to appear). Both of these topics start from low prevalences in September and October 2011, rise rapidly in November 2011, and continue with greater prevalences thereafter. Both topics have roughly equal prevalence across the sources (Suomi24 social media, Helsingin Sanomat news and Twitter), but the prevalences have differing time behavior. Prevalence in Twitter attains a peak fastest for both topics; for the Media topic Twitter prevalence has only one broad peak whereas for the Education topic there are three peaks. Prevalence in Helsingin sanomat shows two peaks for the Media topic,

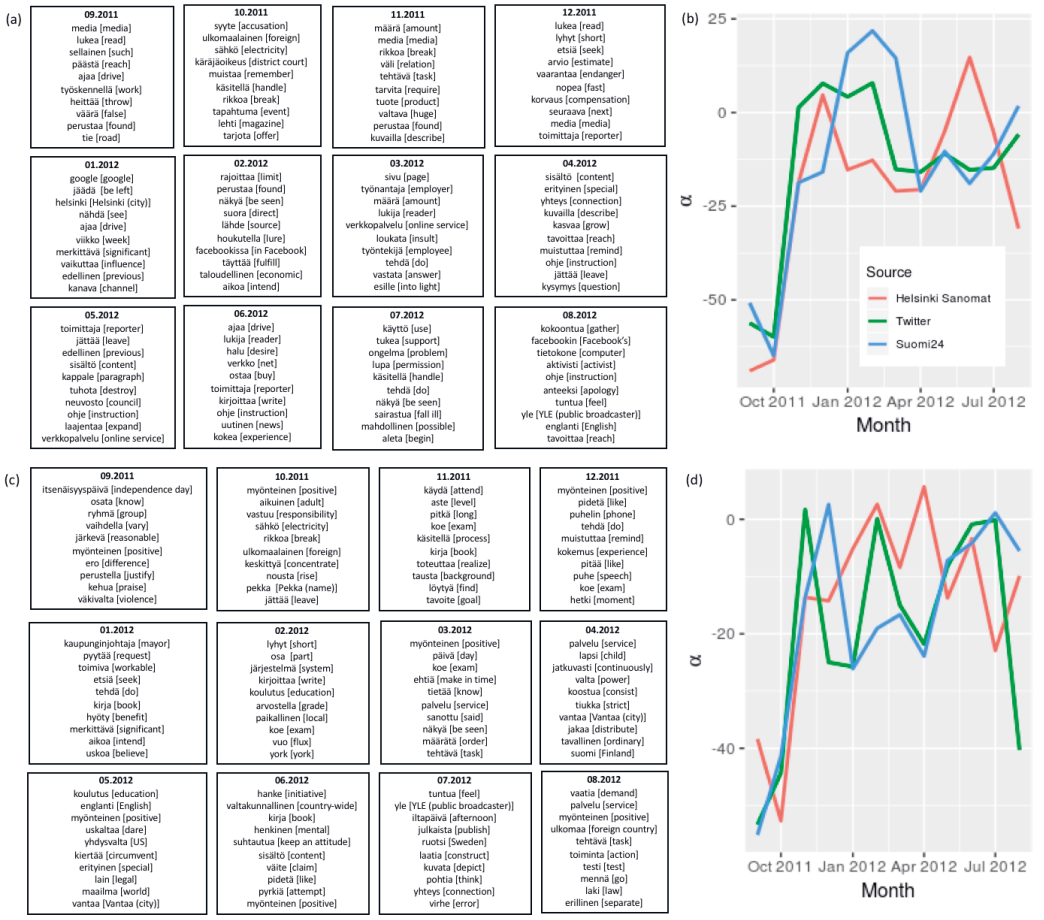

Figure 4: Evolution over time of topic content and topic sparsity (prevalence) in different sources: (a) evolution of the content of topic 'Media' in Finnish news and social media, (b) evolution of topic sparsity for the topic 'Media', (c) evolution of the content of topic 'Education' in Finnish news and social media, (d) evolution of topic sparsity for the topic 'Education'.

and noisy behavior for the Education topic. Prevalence in Suomi24 has a single peak for the Media topic in February 2012, and two peaks in December 2011 and July 2012 for the Education topic. Both topics are sensible in terms of their content and experience reasonable variation of prevalence and content over time. For example, in Figure 4 (a) the top words are all relevant to media but each time slice emphasizes a different aspect of media. It seems that time slices 09.2011, 21.2011, and 05.2012 focus more on news (contain words 'read', 'reporter' and 'paragraph') and time slices 02.2012 and 08.2012 focus more on social media (containing words 'Facebook', 'source' and 'computer'). Similarly, in Figure 4 (c) the top words refer to education with different time slices emphasizing different aspects, for example the time slice 03.2012 04.2012 emphasizes performance evaluation (containing words 'positive', 'exam' and 'task') whereas 04.2012 focuses more on education as a public service (containing words 'service', 'child' and 'city'). Our approach allows smooth changing of topic content, for example in Figure 4 (a) the word 'media' appears in adjacent time slices 11.2011 and 12.2011 of the Media topic, but with less prevalence in the latter.

7 CONCLUSIONS AND DISCUSSION

We introduced DNBGFA, a probabilistic NMF-based model that enables flexible modeling of temporal dynamics using multiple sources of data across data sources (domains) and time slices. Novelties include a Softmax+GP prior and overall structure of the hierarchical model; the model is a novel solution to address temporal dynamics and multiple sources at the same time. The hierarchical structure lets the model incorporate prior knowledge, especially underlying structure of source-source interactions and temporal dynamics, to inference, in addition to the data. The model achieved better generalization ability (ability to predict left-out data) than comparable models in realistic scenarios.The case study showed the model enables discovery of topic evolution and interactions. The model is applicable beyond text data to nonnegative matrices with multiple sources and temporal dynamics.

Our contributions are 1. Hierarchical modeling of topics shared across sources and time and topics unique to sources or time slices; 2. Discovering temporal dynamics of both topic content and prevalence; 3. Comparative studies using both simulated data and real-world data; 4. A real-world demonstration using data from three Finnish text sources.

ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland, decision numbers 312395, 313748, 295694, and 327352.

REFERENCES

[1] Ayan Acharya, Joydeep Ghosh, and Mingyuan Zhou. 2015. Nonparametric Bayesian factor analysis for dynamic count matrices. *arXiv preprint arXiv:1512.08996* (2015).
[2] Claude JP Bélisle. 1992. Convergence theorems for a class of simulated annealing algorithms on d. *Journal of Applied Probability* 29, 4 (1992), 885–895.
[3] David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*. ACM, 113–120.
[4] Yong Chen, Hui Zhang, Junjie Wu, Xingguang Wang, Rui Liu, and Mengxiang Lin. 2015. Modeling emerging, evolving and fading topics using dynamic soft orthogonal nmf with sparse representation. In *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE, 61–70.
[5] Andreas C Damianou, Michalis K Titsias, and Neil D Lawrence. 2016. Variational inference for latent variables and uncertain inputs in Gaussian processes. *The Journal of Machine Learning Research* 17, 1 (2016), 1425–1486.
[6] Zhe Gan, Changyou Chen, Ricardo Henao, David Carlson, and Lawrence Carin. 2015. Scalable deep Poisson factor analysis for topic modeling. In *International Conference on Machine Learning*. 1823–1832.
[7] Sunil Gupta, Dinh Phung, Brett Adams, and Svetha Venkatesh. 2011. A matrix factorization framework for jointly analyzing multiple nonnegative data. In *Proceedings of the Ninth Workshop on Text Mining-Eleventh SIAM International Conference on Data Mining*. Omnipress.
[8] Patrick Jähnichen, Florian Wenzel, Marius Kloft, and Stephan Mandt. 2018. Scalable generalized dynamic topic models. *arXiv preprint arXiv:1803.07868* (2018).
[9] Bin Ju, Yuntao Qian, Minchao Ye, Rong Ni, and Chenxi Zhu. 2015. Using dynamic multi-task non-negative matrix factorization to detect the evolution of user preferences in collaborative filtering. *PloS one* 10, 8 (2015).
[10] Arto Klami, Seppo Virtanen, Eemeli Leppäaho, and Samuel Kaski. 2015. Group factor analysis. *IEEE transactions on neural networks and learning systems* 26, 9 (2015), 2136–2147.
[11] Dana Lahat, Tülay Adali, and Christian Jutten. 2015. Multimodal data fusion: an overview of methods, challenges, and prospects. *Proc. IEEE* 103, 9 (2015), 1449–1477.
[12] Neil D Lawrence. 2004. Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in neural information processing systems*. 329–336.
[13] Neil D Lawrence and Andrew J Moore. 2007. Hierarchical Gaussian process latent variable models. In *Proceedings of the 24th international conference on Machine learning*. ACM, 481–488.
[14] Daniel D Lee and H Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (1999), 788.
[15] Ping Li and Songcan Chen. 2016. A review on gaussian process latent variable models. *CAAI Transactions on Intelligence Technology* 1, 4 (2016), 366–376.
[16] Minnan Luo, Feiping Nie, Xiaojun Chang, Yi Yang, Alexander Hauptmann, and Qinghua Zheng. 2017. Probabilistic non-negative matrix factorization and its robust extensions for topic modeling. In *Thirty-first AAAI conference on artificial intelligence*.
[17] John W Paisley, David M Blei, and Michael I Jordan. 2014. Bayesian Nonnegative Matrix Factorization with Stochastic Variational Inference.
[18] V Paul Pauca, Farial Shahnaz, Michael W Berry, and Robert J Plemmons. 2004. Text mining using non-negative matrix factorizations. In *Proceedings of the 2004 SIAM International Conference on Data Mining*. SIAM, 452–456.
[19] Carl Edward Rasmussen and Christopher KI Williams. 2006. *Gaussian process for machine learning*. MIT press.
[20] Margaret E Roberts, Brandon M Stewart, and Edoardo M Airoldi. 2016. A model of text for experimentation in the social sciences. *J. Amer. Statist. Assoc.* 111, 515 (2016), 988–1003.
[21] Ankan Saha and Vikas Sindhwani. 2010. Dynamic nmfs with temporal regularization for online analysis of streaming text. In *NIPS Workshop on Machine Learning for Social Computing, pp. 1C8*.
[22] Ankan Saha and Vikas Sindhwani. 2012. Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 693–702.
[23] Leo Taslaman and Björn Nilsson. 2012. A framework for regularized non-negative matrix factorization, with application to the analysis of gene expression data. *PloS one* 7, 11 (2012), e46331.
[24] Carmen K Vaca, Amin Mantrach, Alejandro Jaimes, and Marco Saerens. 2014. A time-based collective factorization for topic discovery and monitoring in news. In *Proceedings of the 23rd international conference on World wide web*. ACM, 527–538.
[25] Seppo Virtanen, Arto Klami, Suleiman Khan, and Samuel Kaski. 2012. Bayesian group factor analysis. In *Artificial Intelligence and Statistics*. 1269–1277.
[26] Chong Wang and David M Blei. 2013. Variational inference in nonconjugate models. *Journal of Machine Learning Research* 14, Apr (2013), 1005–1031.
[27] Xuerui Wang and Andrew McCallum. 2006. Topics over time: a non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 424–433.
[28] Mingyuan Zhou, Lauren A Hannah, David B Dunson, and Lawrence Carin. 2012. Beta-negative binomial process and Poisson factor analysis. *Journal of Machine Learning Research* (2012).

PUBLICATION
IV

Cross-structural Factor-topic Model: Document Analysis with Sophisticated Covariates

Chien Lu, Jaakko Peltonen, Timo Nummenmaa, Jyrki Nummenmaa, and Kalervo Järvelin

In: *Asian Conference on Machine Learning, ACML 2021, 17-19 November 2021, Virtual Event*. Ed. by Vineeth N. Balasubramanian and Ivor W. Tsang. PMLR, 2021, pp. 1129–1144

Proceedings of Machine Learning Research 157, 2021 ACML 2021

Cross-structural Factor-topic Model: Document Analysis with Sophisticated Covariates

Chien Lu CHIEN.LU@TUNI.FI
Jaakko Peltonen JAAKKO.PELTONEN@TUNI.FI
Timo Nummenmaa TIMO.NUMMENMAA@TUNI.FI
Jyrki Nummenmaa JYRKI.NUMMENMAA@TUNI.FI
Kalervo Järvelin KALERVO.JARVELIN@TUNI.FI
Faculty of Information Technology and Communication Sciences, Tampere University, Finland

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

Modern text data is increasingly gathered in situations where it is paired with a high-dimensional collection of covariates: then both the text, the covariates, and their relationships are of interest to analyze. Despite the growing amount of such data, current topic models are unable to take into account large amounts of covariates successfully: they fail to model structure among covariates and distort findings of both text and covariates. This paper presents a solution: a novel factor-topic model that enables researchers to analyze latent structure in both text and sophisticated document-level covariates collectively. The key innovation is that besides learning the underlying topical structure, the model also learns the underlying factorial structure from the covariates and the interactions between the two structures. A set of tailored variational inference algorithms for efficient computation are provided. Experiments on three different datasets show the model outperforms comparable topic models in the ability to predict held-out document content. Two case studies focusing on Finnish parliamentary election candidates and game players on Steam demonstrate the model discovers semantically meaningful topics, factors, and their interactions. The model both outperforms state-of-the-art models in predictive accuracy and offers new factor-topic insights beyond other topic models.

Keywords: Probabilistic Modeling, Natural Language Processing, Topic Modeling

1. Introduction

In multiple domains, textual data is paired with accompanying numerical covariates. Examples include questionnaires where free-choice text fields are paired with a set of numerical (continuous or discrete-choice) answers to different questions (often on a Likert scale); political discussion where statements of public figures are paired with their voting record; product reviews where review text is paired with covariates either describing the reviewer along different attributes or scoring the product by multiple criteria; and many others. Such datasets contain structure both within the text content, often described as underlying topics; structure within the set of covariates; and structure linking the text and the covariates. The structures of the covariates and how they interplay with the text content play crucial roles and offer valuable insights. However, current generative probabilistic models do not work well in this setting: the models have overemphasized the text structure only with

little attention to modeling structure in covariates. Available topic models either ignore covariates or simplistically model only direct influence of individual covariates, which yields poor overfitted performance when covariates are high-dimensional. Besides poor predictive performance, such models are also unable to provide insight into the structure in covariates and its relationship to topics. In this paper, we present a solution.

We introduce the *Cross-structural Factor Topic Model* (CFTM), a novel generative probabilistic model which can model the structure of both the text and its high-dimensional numerical covariates. We describe the generative structure of the model, and a parallelizable inference algorithm based on variational approximation. We show in experiments on several data sets that the method yields good performance in modeling held-out document content and yields meaningful insights about structures of covariates and text content.

The rest of the paper is structured as follows. Section 2 discusses related work. Sections 3 and 4 present the proposed method: Section 3 describes the generative model and Section 4 presents the inference approach. Empirical analysis including quantitative and qualitative evaluation is presented in Section 5. Conclusions are given in Section 6.

2. Related Work

For modeling text content of documents alone, topic models of multiple kinds have been proposed. Among them, Latent Dirichlet Allocation (LDA, Blei et al. 2003) is the classical method, which models document content as a bag of words whose word counts arise out of a mixture of latent topics, each of which has its own multinomial word distribution. Nonparametric topic models have been proposed, including Hierarchical Dirichlet Processes (Teh et al. 2006) which aim to learn the number of topics from data. Nonparametric modeling is a direction of future extension for our work.

The Entity topic model (ETM, Kim et al. 2012) models the influence of entities on word content by generating entity mentions from topics and then words from entity-describing word distributions. However, entity mentions are part of text content, no covariates are considered. An Author Topic Model (Rosen-Zvi et al. 2004) was introduced to model relationships between authors, documents, topics and words; however, such models only consider author identity and do not consider author attributes as covariates.

Supervised LDA (sLDA, Mcauliffe and Blei 2008) was developed to model labeled documents. An extended approach called Dirichlet-multinomial regression (DMR, Mimno and McCallum 2008) introduces regression model on topic mixture over covariates. The Sparse additive generative text model (SAGE, Eisenstein et al. 2011) allows topic content to fluctuate by the covariates. However, none of these models allows covariates to affect both topic prevalence and content; our proposed model addresses this.

MetaLDA (Zhao et al. 2017) and Structural topic model (STM, Roberts et al. 2016) both allow covariates to influence topic prevalence and content. However, MetaLDA does not provide a generative model of covariates, and only takes into account simple binary label covariates in modeling topics. STM was recently developed based on SAGE. It is an integrated solution to model covariates (both categorical and continuous) and text. However, the covariates that affect topic content have a limitation, as they allow discrete values only. Thus STM cannot handle sophisticated covariates. We will show in our experiments that STM performance is drastically worsened when the dimension of covariates is high.

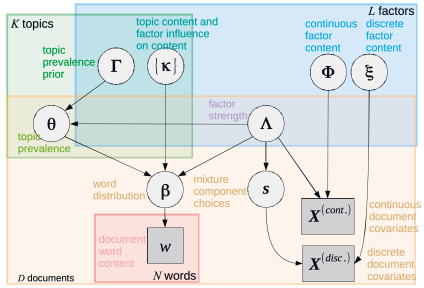

Figure 1: Proposed model. Plates denote topics, factors, documents, and words. The top row of latent variables (circles) describe topics, factors, and their interaction; the 2nd row ($\boldsymbol{\theta}$ and $\boldsymbol{\Lambda}$) describe prevalence of topics and factors in a document; the bottom part describes document content (gray boxes). Factor-loading prior parameter $\boldsymbol{\alpha}$ and noise variances σ, σ_γ, Σ_η, Σ_τ, σ_ϕ, σ_β omitted for clarity.

Distributed Multinomial Regression (Taddy 2015) is an alternative approach which directly models the relationship between the word occurrences and the covariates, but it does not model any structure among the covariates.

Another group of works focuses on combining neural models and topic modeling (Srivastava and Sutton 2017; Card et al. 2018; Gui et al. 2019; Wang and Yang 2020). Among them, SCHOLAR (Card et al. 2018) can be seen as a similar work to STM which incorporates covariates with a variational autoencoder (Kingma and Welling 2014). However, despite their flexibility these models do not generate structure within covariates, they only use covariate values as additional inputs in document content generation. Besides topic models, Non-negative Matrix Factorization (NMF) models are also used for text analysis. In general, a Poisson likelihood is employed to model the observed text whereas multinomial distributed likelihood are typically used by topic models. Many NMF-style works have been proposed (Hu et al. 2016; Acharya et al. 2015; da Silva et al. 2017; Zhao et al. 2018); among such works the most relevant is CTPF (Gopalan et al. 2014) which incorporates a multivariate user-rating matrix into account as covariates, and we will compare to it.

3. Proposed Method

We model a collection of documents indexed by $d \in \{1, \ldots, D\}$ with text content and covariates jointly by a probabilistic model. Word content is distributed over a vocabulary of V unique words indexed by $v \in \{1, \ldots, V\}$ and covariates are indexed by $p \in \{1, \ldots, P\}$. Word content arises from K underlying latent topics indexed by $k \in \{1, \ldots, K\}$, and covariates from $L < P$ underlying latent factors indexed by $l \in \{1, \ldots, L\}$. Topics and factors interact: the strength of the latent factors affects the prevalence of topics and content (word distribution) in each topic. Figure 1 shows the plate model representation of the overall model. We next describe the generative model of the covariates and text content.

3.1. Document-level Latent Variables

Factor Loadings. Each document d is attached with a loading vector over L factors,

$$\boldsymbol{\Lambda}_d = [\lambda_{d,1}, \ldots \lambda_{d,L}]^\top \sim Dir(\boldsymbol{\alpha}) \; . \tag{1}$$

Interaction Coefficients. For each topic $k \in \{1, ...K-1\}$ a L-length coefficient vector is generated as

$$\mathbf{\Gamma}_k \sim \boldsymbol{N}(0, \sigma_\gamma^2 \mathbf{I}_L) \tag{2}$$

to model the relation between the factors and the prevalence of the topic. Note that coefficient vectors for the first $K-1$ topics suffice since topic prevalences sum to 1.

Topic Prevalence. For each document d, the topic prevalence vector $\boldsymbol{\theta}_d = [\theta_{d,1}, \ldots, \theta_{d,K}]$ is generated as $\boldsymbol{\theta}_d = softmax(\boldsymbol{\eta}_d)$ where the auxiliary variables are generated as

$$\boldsymbol{\eta}_{d,1:(K-1)} \sim \boldsymbol{N}(\mathbf{\Gamma}^\top \mathbf{\Lambda}_d, \mathbf{\Sigma}_\eta) \tag{3}$$

and the $\eta_{d,K}$ is fixed to 0.

3.2. Structure of the Covariates

We assume the text content in each document is paired with a set of covariates. Different covariates in the set may require different model types to properly model their structure. Let $x_d^{(p)}$ denote the p:th covariate of document d. We model covariates with two kinds of structure: mixture model and factorization model. The former is suitable especially for discrete covariates, such as multiple-choice values, and the latter for continuous covariates. In both cases the covariate generation depends on a vector $\mathbf{\Lambda}_d$ of L latent parameters. We describe both types of covariate generation next.

Mixture Model. In this structure the p:th covariate is generated from a mixture. The mixture component membership label of the p:th covariate in document d is first generated from a categorical distribution

$$s_d^{(p)} \sim Cat(\mathbf{\Lambda}_d) \tag{4}$$

and the covariate $x_d^{(p)}$ corresponding to the label $s_d^{(p)}$ is then generated as $x_d^{(p)} \sim p(x_d^{(p)} | \boldsymbol{\xi}^{(p)}_{s_d^{(p)}})$ with parameter $\boldsymbol{\xi}^{(p)}_{s_d^{(p)}}$. We model the distribution in each mixture component as a Poisson distribution for covariates that are a count of rare events and as a multinomial distribution for categorical covariates.

Factorization Model. The covariate is directly generated from an exponential family distribution as

$$x_d^{(p)} | \mathbf{\Lambda}_d, \boldsymbol{\phi^{(p)}} \sim \mathbf{ExpFam}\left(\zeta\left(\mathbf{\Lambda}_d, \boldsymbol{\phi^{(p)}}\right), T\left(x_d^{(p)}\right)\right) \tag{5}$$

in which the natural parameter ζ is a weighted average of factor-wise parameters $\phi_l^{(p)} \sim N(0, \sigma_\phi^2)$ weighted by the document-specific factor loadings $\mathbf{\Lambda}_d$, so that

$$\zeta\left(\mathbf{\Lambda}_d, \boldsymbol{\phi^{(p)}}\right) = g^{(p)}\left(\sum_{l=1}^{L} \phi_l^{(p)} \lambda_{d,l}\right) \tag{6}$$

where g is the link function of the exponential-family model. For example, if a Gaussian with a known variance σ^2 is taken as the distribution, we have $x_d^{(p)} \sim N(\sum_{l=1}^{L} \phi_l^{(p)}, \sigma^2)$.

3.3. Structure of Text

Topic Content. We model the word generation process with a SAGE-inspired structure in which each document is attached with a latent vector $\boldsymbol{\beta}_d$ of length V. The v:th element of the latent vector is generated as

$$\beta_{d,v} = \kappa_v^{(w)} + \sum_k \theta_{d,k}\kappa_{v,k}^{(t)} + \sum_l \lambda_{d,l}\kappa_{v,l}^{(f)} + \sum_k \sum_l \theta_{d,k}\lambda_{d,l}\kappa_{v,l,k}^{(i)} + \epsilon_\beta \tag{7}$$

where $\epsilon_\beta \sim N(0, \sigma_\beta^2)$. The $\boldsymbol{\kappa}^{(w)}$ is a vector of length V controlling the overall word prevalence. The overall topic content $\boldsymbol{\kappa}^{(t)}$ is a $V \times K$ matrix, factor influence $\boldsymbol{\kappa}^{(f)}$ is a $V \times L$ matrix, and $\boldsymbol{\kappa}^{(i)}$ is a $V \times L \times K$ array which governs factor-topic interactions on the topic content level, that is, the value of $\kappa_{v,l,k}^{(i)}$ reflects the strength of how much the factor l alters the word probability of v in topic k.

To generate the observed words in the document, for the nth word in document d, the word $w_n^{(d)}$ is sampled from a multinomial distribution

$$w_n^{(d)} \sim MN\left(softmax\left(\boldsymbol{\beta}_d\right)\right) \ . \tag{8}$$

This model design allows the latent factors and topics to interact on both topic prevalence and topic content levels.

4. Variational Inference

We carry out variational inference for the model; variational inference aims to approximate the posterior distribution of model parameters by a factorized distribution q whose components are from known families. Unlike point estimate methods such as maximum a posteriori (MAP), variational inference is able to model a full distribution for parameters based on observations. The parameters of the factorized distribution are optimized by minimizing Kullback-Leibler divergence from the factorized distribution to the true parameter posterior, which becomes equivalent to maximizing the Evidence Lower Bound (ELBO). Iterative optimization optimizes each component distribution given the others; depending on the form of the observation probability and parameter priors, the optimum is obtained analytically for some parameters and by optimization techniques for others. In particular it turns out a crucial part, inference of the topic content, is nontrivial to do computationally efficiently–naive inference is slow; we solve this by a distributed multinomial regression approach with a kernel trick.

Topic Prevalence. Using Laplace Variational Inference (Braun and McAuliffe 2010; Wang and Blei 2013), the variational distribution of η_d is obtained as

$$q(\boldsymbol{\eta}_d) \approx \mathbf{N}(\hat{\boldsymbol{\eta}}_d, -\nabla^2\mathcal{L}(\hat{\boldsymbol{\eta}}_d)^{-1}) \tag{9}$$

where the mean $\hat{\boldsymbol{\eta}}_d$ is the MAP solution, i.e., optimum of

$$\mathcal{L}(\boldsymbol{\eta}_d) \propto -\frac{1}{2}\boldsymbol{\eta}_d^\top \boldsymbol{\Sigma}_\eta^{-1}\boldsymbol{\eta}_d + \boldsymbol{\eta}_d^\top \boldsymbol{\Sigma}_\eta^{-1}\boldsymbol{\Gamma}^\top \boldsymbol{\Lambda}_d + \sum_v c_{d,v} \log \sum_k u_{d,k,v} \exp(\eta_{d,k}) - W_d \log \sum_k \exp(\eta_{d,k}) \tag{10}$$

and $u_{d,k,v}$ is an auxiliary variable

$$u_{d,k,v} = \frac{\exp\left(\kappa_v^{(w)} + \kappa_{v,k}^{(t)} + E[\mathbf{\Lambda}_d]^\top \boldsymbol{\kappa}_v^{(f)} + E[\mathbf{\Lambda}_d]^\top \boldsymbol{\kappa}_{v,k}^{(i)}\right)}{\sum_v \exp\left(\kappa_v^{(w)} + \kappa_{v,k}^{(t)} + E[\mathbf{\Lambda}_d]^\top \boldsymbol{\kappa}_v^{(f)} + E[\mathbf{\Lambda}_d]^\top \boldsymbol{\kappa}_{v,k}^{(i)}\right)} \,. \tag{11}$$

The $\nabla^2 \mathcal{L}(\hat{\boldsymbol{\eta}}_d)$ is a Hessian matrix of $\mathcal{L}(\boldsymbol{\eta}_d)$ at $\hat{\boldsymbol{\eta}}_d$. We find $\hat{\boldsymbol{\eta}}_d$ with the "L-BFGS" optimizer.

Mixture Covariates Model. We infer the component parameters for each membership label $l = 1, \ldots, L$. We present separately the cases for count data and for categorical data. We also infer the distribution of the membership labels.

When the p:th covariate is Count Data (with a Poisson model), we consider the Poisson parameter $\xi_l^{(p)}$ for each membership label $l = 1, \ldots, L$. The optimum of the variational distribution has an analytical form and becomes

$$q(\xi_l^{(p)}) = Gamma(a^{(p)} + E[\mathbf{\Lambda}_l]^\top \boldsymbol{X}_d^{(p)}, b^{(p)} + \sum_d (E[\mathbf{\Lambda}_l])) \tag{12}$$

When the p:th covariate is Categorical Data (with a Multinomial model), we consider for each membership label $l = 1, \ldots, L$ the multinomial parameter $\xi_l^{(p)}$, i.e., the vector of category probabilities. The solution has an analytical form $q(\boldsymbol{\xi}_l^{(p)}) = Dir(\boldsymbol{a}^{(p)} + \sum_d E[s_d^{(p)} = l]\boldsymbol{X}_d^{(p)})$ where $s_d^{(p)}$ is the current membership label for covariate p of document d.

Membership Labels. The variational distribution of the mixture membership label $s_d^{(p)}$ for covariate p of document d is multinomial and the optimum has an analytical form $\log q(s_d^{(p)} = l) \propto \log E[\lambda_{d,l}] + \sum_p \log E[p(x_d^{(p)} | \boldsymbol{\xi}_l^{(p)})]$.

Factorization Covariates Model. Taking advantage of conjugacy, the variational posterior of the factor-wise natural parameters $\boldsymbol{\phi}_p^{(A)}$ is $q(\boldsymbol{\phi}^{(p)}) = \mathbf{N}(\hat{\boldsymbol{\mu}}_{\boldsymbol{\phi}^{(p)}}, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}^{(p)}})$ where the covariance matrix $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}}$ and the mean $\hat{\boldsymbol{\mu}}_{\boldsymbol{\phi}^{(p)}}$ are

$$\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}} = \left({\Sigma_\sigma}^{-1} + \frac{1}{\sigma_\phi^2} \sum_d E_q\left[\mathbf{\Lambda}_d \mathbf{\Lambda}_d^\top\right] \right)^{-1} \,, \quad \hat{\boldsymbol{\mu}}_{\boldsymbol{\phi}^{(p)}} = \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}} \frac{\sum_d E_q\left[\mathbf{\Lambda}_d\right] x_{d,p}^{(A)}}{{\sigma_\phi}^2} \tag{13}$$

where $\Sigma_\sigma = Diag(\sigma^2, \ldots, \sigma^2)$.

Factor Loading. The variational posterior of $\mathbf{\Lambda}_d$ is a Dirichlet distribution parameterized by pseudocount vector $\boldsymbol{\alpha}_{\mathbf{\Lambda}_d}$. To derive the variational posterior (i.e. find the $\boldsymbol{\alpha}_{\mathbf{\Lambda}_d}$), we set up an objective function proportional to the ELBO; the objective function is

$$E_q[(\mathbf{a} + s_d - 1)^\top \log \mathbf{\Lambda}_d + \frac{1}{2}(2\mathbf{b}^\top \mathbf{\Lambda}_d - {\mathbf{\Lambda}_d}^\top \mathbf{A} \mathbf{\Lambda}_d)] - H(\mathbf{\Lambda}_d) \tag{14}$$

where we have

$$\mathbf{b} = E_q\left[{\boldsymbol{\eta}_d}^\top \boldsymbol{\Sigma}_\eta^{-1} \Gamma^\top + \sum_k \theta_{d,k} ({\mathbf{w}_d}^\top \boldsymbol{\kappa}_k^{(i)}) \right] + E_q\left[{\mathbf{X}_d^{(A)}}^\top \boldsymbol{\Sigma}_{(A)}^{-1} \boldsymbol{\phi}^\top \right] \text{ and} \tag{15}$$

$$\mathbf{A} = E_q\left[\Gamma \boldsymbol{\Sigma}_\eta^{-1} \Gamma^\top + \boldsymbol{\phi} \boldsymbol{\Sigma}_{(A)}^{-1} \boldsymbol{\phi}^\top \right] \,, \tag{16}$$

and $H(\mathbf{\Lambda}_d)$ is the entropy of the Dirichlet distribution. Using a Taylor approximation to simplify the computation, denote

$$f(\mathbf{\Lambda}_d) = (\mathbf{a} + s_d - 1)^\top \log \mathbf{\Lambda}_d + \frac{1}{2}(2\mathbf{b}\mathbf{\Lambda}_d - \mathbf{\Lambda}_d\mathbf{A}\mathbf{\Lambda}_d) \ . \tag{17}$$

Then the objective function becomes

$$E_q[f(\mathbf{\Lambda}_d)] - H(\mathbf{\Lambda}_d) \approx f(\hat{\mathbf{\Lambda}}_d) + \frac{1}{2}tr(\nabla^2 f(\hat{\mathbf{\Lambda}}_d) Cov_q(\hat{\mathbf{\Lambda}_d})) - H(\mathbf{\Lambda}_d) \tag{18}$$

where $\hat{\mathbf{\Lambda}}_d = \frac{\boldsymbol{\alpha}_{\mathbf{\Lambda}_d}}{\sum_l \alpha_{\mathbf{\Lambda}_{dl}}}$ is the mean of $\mathbf{\Lambda}_d$ and $\nabla^2 f(\hat{\mathbf{\Lambda}}_d) = Diag(\frac{(1-\mathbf{a}-s_d)}{\hat{\mathbf{\Lambda}}_d^{-2}}) - \mathbf{A}$ is the Hessian matrix. The L-BFGS optimizer is used to optimize (18) with respect to $\boldsymbol{\alpha}_{\mathbf{\Lambda}_d}$.

Topic-Factor Interaction. For $k \in \{1, \ldots, K-1\}$, we derive the variational posterior of the interaction coefficient vector $\mathbf{\Gamma}_k$ which defines the effect of factor loadings on the topic prevalence. As the prior of the coefficients and the likelihood are both normal, taking the advantage of the conjugacy we have the analytical posterior $q(\mathbf{\Gamma}_k) = \mathbf{N}(\hat{\boldsymbol{\mu}}_{\mathbf{\Gamma}_k}, \hat{\mathbf{\Sigma}}_{\mathbf{\Gamma}_k})$ where the covariance matrix $\hat{\mathbf{\Sigma}}_{\mathbf{\Gamma}_k}$ and mean $\hat{\boldsymbol{\mu}}_{\mathbf{\Gamma}_k}$ are

$$\hat{\mathbf{\Sigma}}_{\mathbf{\Gamma}_k} = \left({\mathbf{\Sigma}_\eta}^{-1} \sum_d E_q\left[\mathbf{\Lambda}_d \mathbf{\Lambda}_d^\top\right] + {\Sigma_{\boldsymbol{\gamma}}}^{-1} \right)^{-1} \quad \text{and} \tag{19}$$

$$\hat{\boldsymbol{\mu}}_{\mathbf{\Gamma}_k} = \left(E_q\left[\mathbf{\Lambda}\right] E_q\left[\mathbf{\Lambda}\right]^\top + {\mathbf{\Sigma}_\gamma}^{-1} \right)^{-1} E_q\left[\mathbf{\Lambda}_d\right]^\top E_q\left[\boldsymbol{\eta}\right] \ . \tag{20}$$

Topic Content. The complexity of topic content $\boldsymbol{\beta}$ and $\boldsymbol{\kappa}$ leads to challenges of efficiency and accuracy. A naive derivation of a variational posterior would yield computationally inefficient and non-scalable equations involving inverses of huge matrices and other expensive computations. Instead, we develop a set of tailored inference algorithms based on distributed multinomial regression (Taddy 2015) and a kernel trick (Agrawal et al. 2019), as described next in Proposition 1, Proposition 2, and Theorem 1. The propositions and theorem show how the text structure inference algorithm can be implemented with parallel computation (each vocabulary term can be run in parallel) to enhance efficiency.

Proposition 1 (Distributed Multinomial Regression) *The inference of the $\boldsymbol{\beta}$ in (8) can be performed through conducting inference on independent Poisson models for each word, where each word v has the following generative model:*

$$\boldsymbol{\kappa}_v \sim N(0, \Sigma_\tau) \ , \quad \beta_{d,v} = {\boldsymbol{\kappa}_v}^\top \mathbf{\Psi}_d + \epsilon_\beta \ , \quad w_{d,v} \sim Poisson\left(e^{\beta_{d,v} + \kappa_v^{(w)} + \log m_d}\right) \tag{21}$$

where $\epsilon_\beta \sim N(0, \sigma_\beta^2)$ is the random noise. The notation $\boldsymbol{\kappa}_v = [\kappa_{v,1}^{(t)}, \ldots \kappa_{v,K}^{(t)}, \kappa_{v,1,1}^{(i)}, \ldots, \kappa_{v,L,K}^{(i)}]$ joins together the topic and topic-factor interaction coefficients affecting word v. Correspondingly, $\mathbf{\Psi}_d$ is a mapping function that represents the combined influence terms of both topic prevalences and factor loadings and is defined as

$$\mathbf{\Psi}_d \triangleq \mathbf{\Psi}\left(\boldsymbol{\theta}_d, \mathbf{\Lambda}_d\right) := [\theta_{d,1}, \ldots, \theta_{d,K}, \Lambda_{d,1}, \ldots, \Lambda_{d,L}, \Lambda_{d,1}\theta_{d,1}, \ldots, \Lambda_{d,L}\theta_{d,K}] \tag{22}$$

where the first K elements in the vector are the topic prevalence, the positions are corresponding to the $\boldsymbol{\kappa}_v^{(t)}$ and the rest are topic-factor interactions, their position are corresponding to $\text{vec}(\boldsymbol{\kappa}_v^{(i)})$. In the following, for simplicity we abuse the notation, using $\mathbf{z}_d$ to denote the collection of $\{\boldsymbol{\theta}_d, \boldsymbol{\Lambda}_d\}$. The logarithm of the document length $\log m_d$ is plugged in to serve as the fixed effect (exposure) in the Poisson model.

This framework was proposed by Taddy (2015) to transform the multinomial logistic model into a collection of independent Poisson models to circumvent expensive computations resulting from softmax transformation. Moreover, since the Poisson models are independent of each other, one can easily introduce parallel computation techniques (e.g. map-reduce, Dean and Ghemawat 2008) to speed up the computation. STM also adopted this approach; we apply the framework in a novel factor-topic modeling context. By adapting the framework, the likelihood model (8) is factorized into V independent term-wise Poisson models with a plug-in fixed effect (exposure) shared across terms.

Proposition 2 (Gaussian Process Reparametrization) *The generative model in Proposition 1 can be reparameterized as*

$$g_v \sim GP(0, k_\tau)\ , \quad \beta_{d,v} = g_v(\boldsymbol{z}_d)\ , \quad w_{d,v} \sim Poisson\left(e^{\beta_{d,v} + \kappa_v^{(w)} + \log m_d}\right) \tag{23}$$

where the equation of $\beta_{d,v}$ in (21) is seen as a function with inputs θ_d and λ_d and is then presented as the equation of $\beta_{d,v}$ in (23), and with a Gaussian process prior .

Combining the propositions 1 and 2, taking the weight-space view (see Rasmussen and Williams 2006), the prior of $\boldsymbol{\beta_v}$ becomes

$$\boldsymbol{\beta_v} \sim N(0, K_\tau + \sigma_\beta^2 \mathbf{I}_D) \tag{24}$$

where K_τ is a $D \times D$ matrix with $k_\tau(\mathbf{z}_d, \mathbf{z}_{d'}) \triangleq \boldsymbol{\Psi}_d^\top \Sigma_\tau \boldsymbol{\Psi}_{d'}$. We first find the point estimate $\boldsymbol{\beta_v}^* \triangleq \underset{\boldsymbol{\beta}}{\text{argmax}} f(\boldsymbol{\beta})$ with the objective function

$$f(\boldsymbol{\beta}) = \sum_d \log p(w_{d,v} | \beta_d, m_d) - \log \boldsymbol{\beta}^\top R_\tau \boldsymbol{\beta} \tag{25}$$

where $R_\tau = \left(K_\tau + \sigma_\beta^2 \mathbf{I}_D\right)^{-1}$. We use "L-BFGS" to get the fixed $\kappa_v^{(w)}$ value by $\kappa_v^{(w)} = \frac{1}{D}\sum_d \beta_{d,v}^*$, the margin $\boldsymbol{\beta}^{(m)} = \left[\boldsymbol{\beta}_1^* - \kappa_1^{(w)} \ldots, \boldsymbol{\beta}_V^* - \kappa_V^{(w)}\right]$ is then the posterior mode of $\boldsymbol{\beta}$. These equations infer the posterior of the word distribution parameters $\boldsymbol{\beta}$ which are combinations of topic and factor influences. Next we infer the influence variables $\boldsymbol{\kappa}_v^{(t)}$ of topics to each word and $\boldsymbol{\kappa}_{v,,k}^{(i)}$ of factors to each word and topic, with the following theorem.

Theorem 1 (Kappa Recovery) *Let $\boldsymbol{\theta}_k$ be a k-th unit vector with length K, $\boldsymbol{\Lambda}_l$ be a l-th unit vector with length L, $\boldsymbol{z}_k$ denote the collection $\{\boldsymbol{\theta}_k, \boldsymbol{\Lambda}_0\}$, $\boldsymbol{z}_l$ denote the collection $\{\boldsymbol{\theta}_0, \boldsymbol{\Lambda}_l\}$, and $\boldsymbol{z}_{k,l}$ denote the collection $\{\boldsymbol{\theta}_k, \boldsymbol{\Lambda}_l\}$. Then the posterior of $\kappa_{v,k}^{(t)}$ is $N(\mu_{\kappa_{v,k}^{(t)}}, \sigma^2_{\kappa_{v,k}^{(t)}})$, and the posterior of $\kappa_{v,l}^{(f)}$ is $N(\mu_{\kappa_{v,l}^{(f)}}, \sigma^2_{\kappa_{v,l}^{(f)}})$ where*

$$\mu_{\kappa_{v,k}^{(t)}} = K_\tau\left(\boldsymbol{z}_k, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau \boldsymbol{\beta}_v^{(m)}\ , \quad \mu_{\kappa_{v,l}^{(f)}} = K_\tau\left(\boldsymbol{z}_l, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau \boldsymbol{\beta}_v^{(m)}\ , \tag{26}$$

$$\sigma^2_{\kappa^{(t)}_{v,k}} = k_\tau\left(\boldsymbol{z}_k, \boldsymbol{z}_k\right) + K_\tau\left(\boldsymbol{z}_k, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\boldsymbol{z}_d\}_{d=1}^D, \boldsymbol{z}_k\right) \ , \tag{27}$$

$$\sigma^2_{\kappa^{(f)}_{v,l}} = k_\tau\left(\boldsymbol{z}_l, \boldsymbol{z}_l\right) + K_\tau\left(\boldsymbol{z}_l, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\boldsymbol{z}_d\}_{d=1}^D, \boldsymbol{z}_l\right) \tag{28}$$

and K_τ and R_τ are defined in Proposition 2. The posterior of $\kappa^{(i)}_{v,k,l}$ is $N(\mu_{\kappa^{(i)}_{v,k,l}}, \sigma_{\kappa^{(i)}_{v,k,l}})$ with

$$\mu_{\kappa^{(i)}_{v,k,l}} = [-1,-1,1]\, K_\tau\left(\{\boldsymbol{z}_k, \boldsymbol{z}_l, \boldsymbol{z}_{k,l}\}, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau \boldsymbol{\beta}_v^{(m)} \tag{29}$$

where $[-1,1,1]$ is simply the 1×3 matrix with elements 1 and -1, and the variance is

$$\begin{aligned}\sigma^2_{\kappa^{(i)}_{v,k,l}} = {} & k_\tau\left(\boldsymbol{z}_k, \boldsymbol{z}_k\right) + k_\tau\left(\boldsymbol{z}_l, \boldsymbol{z}_l\right) + k_\tau\left(\boldsymbol{z}_{k,l}, \boldsymbol{z}_{k,l}\right) + K_\tau\left(\boldsymbol{z}_k, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\boldsymbol{z}_d\}_{d=1}^D, \boldsymbol{z}_k\right) + \\ & K_\tau\left(\boldsymbol{z}_l, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\boldsymbol{z}_d\}_{d=1}^D, \boldsymbol{z}_l\right) + K_\tau\left(\boldsymbol{z}_{k,l}, \{\boldsymbol{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\boldsymbol{z}_d\}_{d=1}^D, \boldsymbol{z}_{k,l}\right) \ . \end{aligned} \tag{30}$$

Proof By Proposition 2, $g(\mathbf{z}_k) = \kappa^{(t)}_{v,k}$, thus, given the multivariate normal distribution

$$\begin{bmatrix} \boldsymbol{\beta}_v^{(m)} \\ g(\mathbf{z}_k) \end{bmatrix} \sim N\left(\mathbf{0}, \begin{bmatrix} K_\tau + \sigma_\beta^2 \mathbf{I}_D & K_\tau\left(\{\mathbf{z}_d\}_{d=1}^D, \mathbf{z}_k\right) \\ K_\tau\left(\mathbf{z}_k, \{\mathbf{z}_d\}_{d=1}^D\right) & k_\tau\left(\mathbf{z}_k, \mathbf{z}_k\right) \end{bmatrix}\right) \tag{31}$$

the posterior mean and variance of $\kappa^{(t)}_{v,k}$ can be obtained as

$$\mu_{\kappa^{(t)}_{v,k}} = E\left[g(\mathbf{z}_k) | \{\mathbf{z}_d\}_{d=1}^D, \boldsymbol{\beta}_v^*\right] = K_\tau\left(\mathbf{z}_k, \{\mathbf{z}_d\}_{d=1}^D\right) R_\tau \boldsymbol{\beta}_v^{(m)} \ \text{ and} \tag{32}$$

$$\sigma^2_{\kappa^{(t)}_{v,k}} = Var\left(g(\mathbf{z}_k) | \{\mathbf{z}_d\}_{d=1}^D, \boldsymbol{\beta}_v^*\right) = k_\tau\left(\mathbf{z}_k, \mathbf{z}_k\right) + K_\tau\left(\mathbf{z}_k, \{\mathbf{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\mathbf{z}_d\}_{d=1}^D, \mathbf{z}_k\right) \ . \tag{33}$$

Similarly, the posterior mean and variance of $\kappa^{(f)}_{v,l}$ are

$$\mu_{\kappa^{(t)}_{v,k}} = K_\tau\left(\mathbf{z}_k, \{\mathbf{z}_d\}_{d=1}^D\right) R_\tau \boldsymbol{\beta}_v^{(m)}, \ \sigma^2_{\kappa^{(f)}_{l,k}} = k_\tau\left(\mathbf{z}_l, \mathbf{z}_l\right) + K_\tau\left(\mathbf{z}_l, \{\mathbf{z}_d\}_{d=1}^D\right) R_\tau K_\tau\left(\{\mathbf{z}_d\}_{d=1}^D, \mathbf{z}_l\right) \ . \tag{34}$$

Since we have $g(\mathbf{z}_{k,l}) = \kappa^{(t)}_{v,k} + \kappa^{(f)}_{v,l} + \kappa^{(i)}_{v,k,l}$, given the multivariate normal distribution

$$\begin{bmatrix} \boldsymbol{\beta}_v^{(m)} \\ g(\mathbf{z}_{k,l}) \end{bmatrix} \sim N\left(\mathbf{0}, \begin{bmatrix} K_\tau + \sigma_\beta^2 \mathbf{I}_D & K_\tau\left(\{\mathbf{z}_d\}_{d=1}^D, \mathbf{z}_{k,l}\right) \\ K_\tau\left(\{\mathbf{z}_{k,l}, \mathbf{z}_d\}_{d=1}^D\right) & k_\tau\left(\mathbf{z}_{k,l}, \mathbf{z}_{k,l}\right) \end{bmatrix}\right) \tag{35}$$

the posterior mean and variance of $\kappa^{(i)}_{v,k,l}$ can be obtained accordingly via

$$\mu_{\kappa^{(i)}_{v,k,l}} = E[g(\mathbf{z}_{k,l}) - g(\mathbf{z}_k) - g(\mathbf{z}_l) | \{\mathbf{z}_d\}_{d=1}^D, \boldsymbol{\beta}_v^{(m)}] \ \text{ and} \tag{36}$$

$$\sigma^2_{\kappa^{(t)}_{v,k}} = Var\left(g(\mathbf{z}_{k,l}) - g(\mathbf{z}_k) - g(\mathbf{z}_l) | \{\mathbf{z}_d\}_{d=1}^D, \boldsymbol{\beta}_v^{(m)}\right) \ . \tag{37}$$

■

The process for the inference of text structure is summarized in Algorithm 1 and the entire inference process is shown in Algorithm 2, where the update steps correspond to the equations described in this section for each parameter.

Algorithm 1: Text Structure Inference	**Algorithm 2:** Variational Inference
Data: Term-document Matrix **Hyper-parameters:** σ_β, Σ_τ **Result:** $\{\boldsymbol{\beta}^{(m)}, \boldsymbol{\kappa}^{(w)}, \boldsymbol{\kappa}^{(t)}, \boldsymbol{\kappa}^{(i)}\}$ **for** v in $1, \ldots, V$ **do** Obtain $\boldsymbol{\beta}_v^*$ with (25) Obtain $\kappa_v^{(w)}$ with (4) Obtain $\boldsymbol{\beta}_v^{(m)} = \boldsymbol{\beta}_v^* - \kappa_v^{(w)}$ Recover $\boldsymbol{\kappa}_v$ with Theorem 1 **end**	**Data:** Term-document Matrix $\mathbf{W}$, Covariates $\mathbf{X}$ **Model Setting:** K, L **Hyper-parameters:** σ_β, Σ_τ, Σ_η, σ_γ, σ_ϕ **Result:** $\{\boldsymbol{\beta}, \boldsymbol{\kappa}, \boldsymbol{\eta}, \boldsymbol{\Gamma}, \boldsymbol{\Lambda}, \text{s}, \boldsymbol{\phi}, \boldsymbol{\xi}\}$ **for** t in $1, \ldots, maxit$ **do** Update $\boldsymbol{\beta}$, $\boldsymbol{\kappa}$ (Text Structure) Update $\boldsymbol{\eta}$, $\boldsymbol{\Gamma}$, $\boldsymbol{\Lambda}$, s (Local Variables) Update $\boldsymbol{\phi}$, $\boldsymbol{\xi}$ (Covariate Structure) **end**

5. Empirical Study

The empirical study comprises two parts. In the first part we compare our model quantitatively with other state-of-the-art approaches. We will show that it outperforms the other methods with regard to predictive performance on held-out data. The second part contains qualitative evaluations on case studies which demonstrate the usability of CFTM for gaining insight into text data and their covariates. The fitted CFTM model is used to extract underlying topics, structure among the covariates, and their interactions.

5.1. Datasets

We perform the empirical study using three real-world datasets.

Yle Election Compass 2019 is a survey directed to candidates for Finnish parliamentary elections with results open to the public.[1] It collects each candidate's basic information and agreement with different statements about ideological viewpoints, societal issues and policies, measured by 29 Likert scale questions (score 1-5). Candidates can elaborate their answers to advertise or communicate to voters; We take the written content of each candidate as the text document, the Likert scale questions as continuous variables, and gender and native languages as categorical variables. Text is lemmatized, numbers, punctuation and stop-words are removed. Texts with more than 40 words are taken for analysis, the final dataset contains 1937 documents and 1764 vocabulary terms. The original text is in Finnish, in the case study shown in Section 5.3 we provide an English translation.

Doom Eternal Game Reviews were collected from Steam[2], a popular gaming platform with an abundance of player-written game reviews. We focused on a first-person shooter game "Doom Eternal". Review texts and corresponding metadata were collected via SteamAPI and profile data was crawled from public profile pages linked with collected Steam IDs. The positivity/negativity (if the reviewer recommends the game or not) is taken

1. https://vaalikone.yle.fi/eduskuntavaali2019 ; https://yle.fi/uutiset/3-1072538.
2. https://store.steampowered.com/

as a categorical variable. The number of submissions and guides of users are rare events so they are considered count variables. We identified 22 continuous variables such as number of achievements, and played time, etc. For text content, numbers, punctuations and stop words were removed and the text was lemmatized. Reviews with more than 40 words after processing were kept. Finally, a collection of 1144 reviews and 2377 terms remained.

Airport Lounge User Reviews were collected from Skytrax[3], in which customers can give numerical ratings (score 1-5, including aspects such as comfort and staff) to the airport lounges together with written reviews. Again, we keep reviews with more than 40 words, numbers, punctuations and stop words were removed, and texts are lemmatized. The processed data set contains 1311 reviews with 2799 vocabulary terms, paired with 8 numerical ratings, and 2 categorical ratings (recommend or not).

5.2. Quantitative Evaluation

We compare our model with four state-of-the-art models: LDA, STM, MetaLDA, and SCHOLAR. The performance comparison focuses on held-out prediction using the above-mentioned datasets. Details are described as follows.

Evaluation Metric. The held-out likelihood is used to evaluate model performance. The text content is randomly divided into training and held-out sets, each containing 50% of the original content [4]. The training set is used to fit the models. The fitted model is then used to predict the held-out text content and the held-out likelihood values are computed. Note that another typical metric, perplexity on the test set, is an exponential transformation of the held-out likelihood: higher held-out likelihood means lower perplexity.

Experimental Settings. CFTM is run with simple unoptimized prior settings $\alpha = 10 \cdot \mathbf{1}$, $\Sigma_\gamma = \Sigma_\tau = 10 \cdot \mathbf{I}$, $\sigma_\eta = \sigma_\phi = 0.1$, $\sigma_\beta = 0.01$. Other methods are run with their default values. We evaluate the model performance on different settings (combinations of the number of topics $K \in \{5, 10, 15, 20\}$ and number of factors $L \in \{5, 10\}$). To assess robustness of the methods to limited data, we run experiments both on the full data sets and on a random draw of 500 documents. The document subset sampling (in the limited-data case), train-test division, and model fitting are repeated 10 times for each setting.

Running time. We implemented our algorithms in R [5]. Using the parallel implementation, on average our model takes around 8 minutes and 13 minutes to converge using 8 and 4 cores respectively. In contrast, the R implementation of STM (a method also having covariates) takes 18 minutes to converge, clearly longer than our model.

Results. The result is shown in Figure 2. In most settings (Yle Compass with 500 samples, Doom Eternal full data set and 500-samples, Lounge reviews full) CFTM clearly and statistically significantly outperforms all other methods. In two settings results were closer: for Lounge reviews with 500 samples, CFTM with $L = 5$ is statistically significantly better than the closest competitor SCHOLAR for 5 and 10 topics and not significantly different for 15 and 20; for the Yle Compass full data set the difference to the closest competitor LDA is not statistically significant. Overall, CFTM has consistently good performance.

3. www.airlinequality.com, we use the collection https://github.com/quankiquanki/skytrax-reviews-dataset
4. Note that the 50%-50% division is chosen according the practice used in STM(Roberts et al. 2016).
5. source code and data sets used in this work can be found in supplementary material

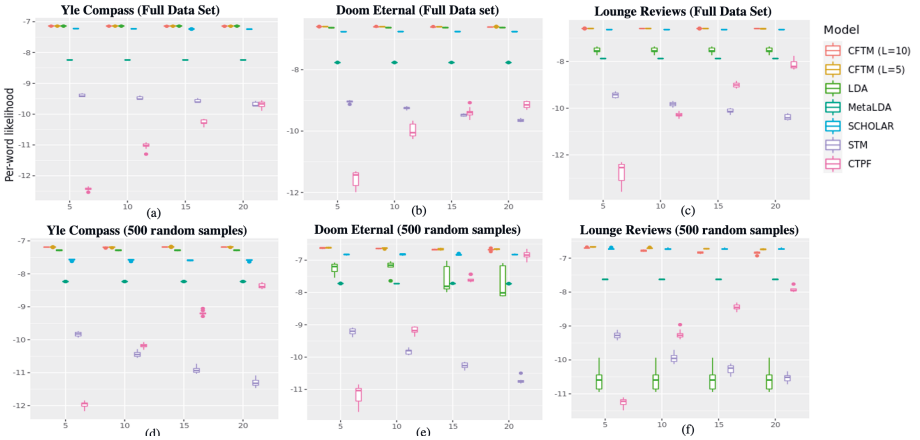

Figure 2: Quantitative evaluation, performance comparison with held-out likelihood (per word), higher is better. CFTM is compared to LDALDA, MetaLDA, SCHOLAR, STM, MetaLDA, and CTPF. (a)-(c): Comparison on the full dataset. (d)-(f): Comparison on 500 random samples. The box plots show variation of performance over random simulations or random divisions of data.

5.3. Case Studies

We conduct empirical analyses using CFTM on *Yle Election Compass 2019* and *Doom Eternal Game Reviews* datasets. The hyper-parameter setting is the same as above but the model is trained with the full datasets. Among multiple choices of the number of topics and factors, we use semantic coherence (Mimno et al. 2011) as the model selection criterion to choose the best CFTM model for inspection.

Spectrum of political positions. When fitting the *Yle Election Compass 2019* dataset, the CFTM model with 9 topics and 5 factors was selected. Figure 3(a) shows the top words for the 9 extracted topics. Topic names are assigned by authors by analysis of the topic words. CFTM has found clear topical content appropriate in the domain: each topic uncovers different aspects of political interests ranging from local politics (*Local Politics of Pirkanmaa*) to climate issues (*Climate Change and Costs*).

Figure 3 (b) displays the factor structures of three factors: *Eurosceptic*, *Green*, and *Pro-global* [6] and Figure 3 (c) presents their influences on topic content. Similarly to the topics, factor labels can be assigned by analyzing their feature weights (posterior mean of ϕ). For example, the factor *Green* supports environmental protection, having high agreement with statements such as "Climate is worth the cost", "Discourage eating meat", and "Reduce tree cutting". The factor *Eurosceptic* agrees with statements "Leave eurozone" and "Immigrants

6. The feature weights of all the 5 factors are provided in supplementary material

Topic	Top 10 Words
Bans and Actions	ban, have to, want, Swedish, choose, action, such, hundred, accept, positive
Climate Change and Costs	climate, get, suffice, Euro, price, company, discrimination, climate change, event, reality
Local Politics of Pirkanmaa	Pirkanmaa, Finland, such, must, nearby year, alcohol, God, habit, rich, decision maker
Support the Youth	end stage, young, use, business activity, rule-out, livelihood, external, support, society, both
Tourism Business	company, problem, tourism, money, strawberry, working life, solution, Russia, correct, Swedish
Young Immigrants	young, prerequisite, energy drink, decision, arrangement, crime, secure, immigration, job, can
Offer Support and Alternatives	increase, offer, support, act, alternative, reject, quite, sport, guide, grounds
Life Attitudes	working life, level, speak, decide, God, mind, suffice, act, possible, wrong
Problem Resolution	manner, part, situation, develop, strive, try, act, secure, exploit, problem

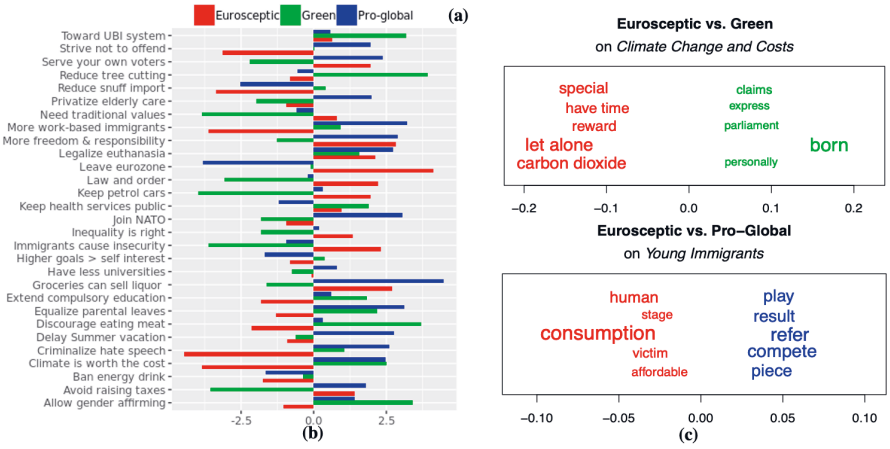

Figure 3: CFTM results for Yle Election Compass. (a) Extracted topics. Top words are shown in order of frequency in the topic. (b) Feature weights of factors *Eurosceptic*, *Green* and *Pro-global*. (c) Wording difference of *Eurosceptic* vs. *Green* on the topic *Climate Change and Costs*, and wording difference of *Eurosceptic* vs. *Pro-global* on topics *Young Immigrants*. Horizontal position of a term v shows the difference $\kappa^{(i)}_{v,l,k} - \kappa^{(i)}_{v,l',k}$ in topic k of factors l and l'.

cause insecurities", and disagrees with "Join NATO"; and the factor *Pro-global* holds an opposite position on the above statements and supports "More work-based Immigrants".

The impact of factors on wordings of a topic can be explored with the posterior of $\kappa^{(i)}$. Figure 3 (c) examines factor influence on wordings, showing the comparison of *Eurosceptic* vs. *Green* on the topic *Climate Change and Costs* and the comparison of *Eurosceptic* vs. *Pro-global* on the topic *Young Immigrants*. The horizontal axis reveals the difference of influence between two factors on prominence of words. Candidates with high loading along the *Eurosceptic* factor use more words 'let alone' and 'make time' when discussing the topic

Climate Change and Costs whereas candidates aligned along the factor *Green* emphasize 'claims', and 'personally'. On the other hand, when it comes to the topic emphasize *Young Immigrants*, candidates aligned along the factor *Eurosceptic* use more words such as 'victim', and 'consumption', whereas candidates aligned along the factor *Pro-global* use more words such as 'compete' and 'result'. The differing wording preferences among the factors corresponding to competing political orientations shows how the same issues (topics) are approached from very different perspectives by candidates aligned along those factors.

Exploring player experiences. The CFTM model of 6 topics and 7 factors was selected when fitting the *Doom Eternal Game Reviews* dataset. Figure 4 (a) displays the topic words of the extracted topics. The topics cover game mechanics (e.g. *Fighting*, *Damage and Survival*) and more general views on the game (*Feelings and Experiences*) and issues external to the play experience (*Support and Services*). Figure 4 (b) [7] and (c) further present the feature weights of factors *Doom-focused Player*, *Game Collector*, and their influences on topics *Support and Services* and *Feelings and Experiences*. Players with high loading of the factor *Doom-focused Player* are more likely to use words like 'doom' and 'account' in the topic *Support and Services* and 'feel', 'weapon' in topic *Feelings and Experiences*, whereas players with a high loading of the factor *Game Collector* prefer words 'rip', 'tear' in both topics *Support and Services* and *Feelings and Experiences*. The topics, factors and interactions are well-suited for the domain.

6. Conclusions

We presented the Cross-structural Factor-Topic Model (CFTM), a novel generative probabilistic model for text documents occurring with sophisticated covariates. It represents latent topical structure in text, factor structure in covariates, and influence of the factors on both topic prevalence and content. The model is flexible, allowing both discrete covariates with a mixture structure and continuous covariates with a factorized structure in the same model. We proposed an efficient inference scheme coupling variational inference to efficient distributed inference. In experiments the model outperformed LDA, STM, MetaLDA, and SCHOLAR; moreover, CFTM discovered meaningful topics, factors, and factor influences in case studies investigating a political survey and reviews of a computer game.

Acknowledgments

This work is supported by the Academy of Finland decisions 312395, 313748, and 327352.

References

A. Acharya, D. Teffer, J. Henderson, M. Tyler, M. Zhou, and J. Ghosh. Gamma process Poisson factorization for joint modeling of network and documents. In *Proc. ECML PKDD*, pages 283–299. Springer, 2015.

R. Agrawal, B. Trippe, J. Huggins, and T. Broderick. The kernel interaction trick: Fast bayesian discovery of pairwise interactions in high dimensions. In *Proc. ICML*, pages 141–150, 2019.

7. The feature weights of all the 7 factors are provided in supplementary material

Topic	Top 10 Words
Fighting	rip, tear, dream, frankly, potato, neon, success, kar, smoothly, hugo
Support and Services	support, response, week, account, anyway, offline, everything, paste, team, approve
Visuals and Features	doom, dream, neon, march, hdr, doot, mayhem, replayability, market, kickass
Damage and Survival	damage, thing, though, run, challenge, player, bad, lot, combat, people
Movement and Weapons	dash, weapon, contain, minute, maykr, grenade, teleport, switch, pad, ammo
Feelings and Experiences	really, game, good, recommend, software, feel, level, learn, start, whole

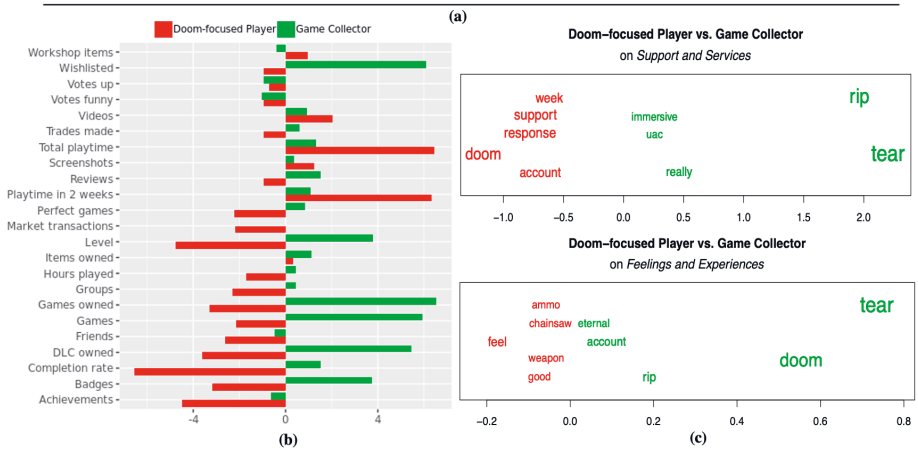

Figure 4: CFTM results for Doom Eternal. (a) Extracted topics. (b) Feature weights of factors *Doom-focused Player* and *Game Collector*. (c) Wording difference of factors *Doom-focused Player* vs. *Game Collector* on the topic *Support and Services* and *Feelings and Experiences*.

D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

M. Braun and J. McAuliffe. Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, 105(489):324–335, 2010.

D. Card, C. Tan, and N. A. Smith. Neural models for documents with metadata. In *Proc. ACL*, pages 2031–2040. ACL, 2018.

E. de Souza da Silva, H. Langseth, and H. Ramampiaro. Content-based social recommendation with Poisson matrix factorization. In *Proc. ECML PKDD*. Springer, 2017.

J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

J. Eisenstein, A. Ahmed, and E. Xing. Sparse additive generative models of text. In *Proc. ICML*, pages 1041–1048. ACM, 2011.

P. K. Gopalan, L. Charlin, and D. Blei. Content-based recommendations with Poisson factorization. In *Proc. NIPS*, pages 3176–3184, 2014.

L. Gui, J. Leng, G. Pergola, R. Xu, and Y. He. Neural topic model with reinforcement learning. In *Proc. EMNLP-IJCNLP*, pages 3469–3474. ACL, 2019.

C. Hu, P. Rai, and L. Carin. Non-negative matrix factorization for discrete data with hierarchical side-information. In *Proc. AISTATS*, pages 1124–1132. PMLR, 2016.

H. Kim, Y. Sun, J. Hockenmaier, and J. Han. Etm: entity topic models for mining documents associated with entities. In *Proc. ICDM*, pages 349–358. IEEE, 2012.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *Proc. ICLR*, 2014.

J. Mcauliffe and D. Blei. Supervised topic models. In *Proc. NIPS*, pages 121–128, 2008.

D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *Proc. UAI*, pages 411–418. AUAI Press, 2008.

D. Mimno, H. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *Proc. EMNLP*, pages 262–272. ACL, 2011.

C. Rasmussen and C. Williams. *Gaussian processes in machine learning*. MIT Press, 2006.

M. Roberts, B. Stewart, and E. Airoldi. A model of text for experimentation in the social sciences. *Journal of the American Statistical Association*, 111(515):988–1003, 2016.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proc. UAI*, pages 487–494. AUAI Press, 2004.

A. Srivastava and C. A. Sutton. Autoencoding variational inference for topic models. In *Proc. ICLR*. OpenReview.net, 2017.

M. Taddy. Distributed multinomial regression. *The Annals of Applied Statistics*, 9(3): 1394–1414, 2015.

Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101:1566–1581, 2006.

C. Wang and D. Blei. Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14(Apr):1005–1031, 2013.

X. Wang and Y. Yang. Neural topic model with attention for supervised learning. In *Proc. AISTATS*, pages 1147–1156. PMLR, PMLR, 2020.

H. Zhao, L. Du, W. Buntine, and G. Liu. MetaLDA: A topic model that efficiently incorporates meta information. In *Proc. ICDM*, pages 635–644. IEEE, 2017.

H. Zhao, P. Rai, L. Du, and W. Buntine. Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In *Proc. AISTATS*. PMLR, 2018.

PUBLICATION V

Nonparametric Exponential Family Graph Embeddings for Multiple Representation Learning

Chien Lu, Jaakko Peltonen, Timo Nummenmaa, and Jyrki Nummenmaa

In: *Uncertainty in Artificial Intelligence, 1-5 August 2022, Eindhoven, The Netherlands*. Ed. by James Cussens and Kun Zhang. PMLR, 2022, pp. 1275–1285

Nonparametric Exponential Family Graph Embeddings for Multiple Representation Learning

Chien Lu[1] **Jaakko Peltonen**[1] **Timo Nummenmaa**[1] **Jyrki Nummenmaa**[1]

[1]Tampere University, Finland

Abstract

In graph data, each node often serves multiple functionalities. However, most graph embedding models assume that each node can only possess one representation. We address this issue by proposing a nonparametric graph embedding model. The model allows each node to learn multiple representations where they are needed to represent the complexity of random walks in the graph. It extends the Exponential family graph embedding model with two nonparametric prior settings, the Dirichlet process and the uniform process. The model combines the ability of Exponential family graph embedding to take the number of occurrences of context nodes into account with nonparametric priors giving it the flexibility to learn more than one latent representation for each node. The learned embeddings outperforms other state of the art approaches in link prediction and node classification tasks.

1 INTRODUCTION

Data in the form of graphs is drastically growing across disciplines to represent complex observations and their relationships in the graph topology. One common challenge for such data is unsupervised representation learning (embedding) which discovers underlying functions or characterizations of nodes solely from the graph structure without requiring availability of node attributes. Such research has shown encouragingly that the learned latent representations can be used as features for different predictive tasks with promising performance.

Despite the success of such models, most of the proposed methods consider only the co-appearance pattern of nodes in walks across a graph. The prominence of nodes in their surroundings, for example as hubs or bridges, is an important trait of the network structure but is often ignored.

Moreover, it is a common phenomenon that each graph node can serve different functions or roles: a node can, for example, act both as a local hub for its nearby nodes and also as a crucial bridge along a path between far-off connected areas of a graph. However, most methods are unable to properly represent this: they are restricted to single representation learning where each node is only assigned one latent vector representation. A model that only supports one embedding per node tries to collapse all underlying roles of the node into one vector representation could omit necessary information: this can yield poor representations that are 'inbetween' the roles of the node and do not represent any of them well or represent only some roles while ignoring others.

In this paper we introduce a novel embedding model, which extends exponential family embedding [Rudolph et al., 2016] with nonparametric priors and allows a node to have more than one latent representation. We allocate such latent representations following two nonparametric priors, the Dirichlet process and the uniform process. While Dirichlet processes are popular in nonparametric modeling, the uniform process has been neglected in such models; our results show the uniform process is a promising prior for the proposed model. A tailored truncation-free inference algorithm is developed. Different from the traditional approaches, the algorithm introduces new latent embedding vectors over iterations which provides more efficient inference.

We evaluate the proposed model with two tasks, link prediction and node classification. Results over several datasets show the proposed multiple representation learning method improves performance compared to state of the art baselines.

The contributions of this work are:

- We introduce the notion of multiple representation to graph embeddings: each node can have more than one latent vector representation.
- We propose a graph embedding model leveraging Bayesian nonparametrics, which is unprecedented and challenging to do well. The number of latent represen-

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

tations are thus decided by the observed data.

- In addition to the Dirichlet process, we explore the uniform process, and show it is an important option for achieving best results.
- We develop an adaptive inference algorithm for efficient computation.

The paper is organized as follows. Section 2 describes background concepts. Section 3 introduces the proposed model. Section 4 develops the inference algorithm. Experiments are conducted in Section 5 and Section 6 draws the conclusions.

2 FUNDAMENTAL CONCEPTS

This section provides a brief overview of some basic concepts that are related to our approach.

2.1 EXPONENTIAL FAMILY EMBEDDING

Exponential family embedding (EFE) [Rudolph et al., 2016] is a probabilistic extension of the CBOW embedding model [Mikolov et al., 2013a,b]. Observations are made of objects v that occur at locations n surrounded by a context which is a set of other objects. In a traditional word embedding scenario an object would be a word and the context would be the surrounding words in a sentence; in the graph embedding scenario that we address, objects are instead nodes of a graph and contexts are other nodes on a random walk in the graph.

Let $x_{n,v}$ denote the observed value for object v at location n. Denote the context by a set $\mathbf{c}_n = \{v'\}$ of other objects v' and a vector $\tilde{\mathbf{x}}_{\mathbf{c}_n} = \{\tilde{x}_{n,v'}\}$ of their values in the context. In our graph embedding case, the values represent whether the object (graph node) occurs at the location and how many times the context objects (nodes) occur in the context.

In EFE, conditioning on the context set $\mathbf{c}_n$ and context values $\tilde{\mathbf{x}}_{\mathbf{c}_n}$, the observed value $x_{n,v}$ for object v is assumed to be exponential family distributed:

$$x_{n,v}|\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n} \sim \mathbf{ExpFam}\left(\eta_v\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)\right) \quad (1)$$

where $\mathbf{ExpFam}$ is an exponential family distribution, $\eta_v\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right)$ is the natural parameter, and $T\left(x_{n,v}\right)$ denotes the sufficient statistics.

In EFE, each object v is represented in two ways, with an embedding vector $\boldsymbol{\rho}_v \in \mathbb{R}^D$ and a context vector $\boldsymbol{\alpha}_v \in \mathbb{R}^D$ where D is the embedding dimensionality. The EFE captures the co-occurrence pattern by constructing the natural parameter based on interaction between the embedding vector of the center object and the context vectors of its context objects weighted by their context values. The model can be seen as a special generalized linear model since the natural parameter is modeled as a link function of an inner product, so that

$$\eta_v\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right) = g\left(\boldsymbol{\rho}_v^\top \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \boldsymbol{\alpha}_{v'}\right). \quad (2)$$

Since $\mathbf{ExpFam}$ can be any exponential distribution, CBOW can be seen as the special case of employing a Bernoulli distribution where the observed value $x_{n,v}$ can be either 1 or 0. One principal merit of the generalization to other probability distributions is the capability of capturing latent patterns by incorporating the observed values. For example, in a shopping cart scenario, quantity of an observed item is modeled by the quantities of its context items (i.e., other products in the shopping cart) which are not binary but positive integers. Similarly, in a graph embedding scenario counts of graph nodes in a context will be positive integers.

2.2 RANDOM WALK BASED NODE EMBEDDING

Let $\mathcal{G} = (\mathbb{V}, \mathbb{E})$ be a graph where $\mathbb{V}$ denotes the set of vertices, and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ denotes the edge set. A random walk $\mathbf{w} = \{w_1, \ldots, w_L\}$ of length L is a simulated sequence of nodes over the graph where each node is chosen at random from the neighbors of the previous node. Extraction of such random walks is a way to describe a graph by extracting sequence data representing graph connectivity. Such sequences can then be modeled by a generative model.

Random walk based embedding approaches [Perozzi et al., 2014, Grover and Leskovec, 2016] model co-occurrence of nodes in a set of random walks $\mathcal{W}$. The generative process models the sequence content, and thus the graph connectivity, through embeddings of nodes: the model is conditional on the nodes and generates the sequences.

Given a walk $\mathbf{w} \in \mathcal{W}$, the occurrence of node w_n at position n in the walk is conditional on the set $\mathbf{c}_n$ of its surrounding (context) nodes in the walk. The occurrence probability is modeled as depending on embedding vectors of the node and embedding vectors of the context nodes. The representation learning aims to optimize the probability of occurrence of the nodes w_n given their contexts, i.e., $\prod_{\mathbf{w} \in \mathcal{W}} \prod_n p(w_n|\mathbf{c}_n)$.

2.3 BAYESIAN NONPARAMETICS

In Bayesian nonparametric models, the number of parameters is not fixed in advance but learned during model fitting up to a potentially infinite number of parameters. The models are typically described as mixtures: each observation is modeled by a parameter drawn from a distribution G over the space of parameters (e.g. $\mathbb{R}^D$) where only a finite number of parameter values have nonzero probability, but G itself is drawn as

$$G \sim NP(G_0, \gamma) \quad (3)$$

from a stochastic process prior NP with base distribution G_0 and concentration parameter γ. The process NP yields distributions over the parameter space, with different numbers of possible values up to a potentially infinite number, but each draw from NP has a finite number. Thus fitting the model to data with the prior NP will infer how many parameters are needed to describe the data.

2.4 RELATED WORK

Among random walk based unsupervised node embeddings, Deepwalk [Perozzi et al., 2014] has been the classical method. Grover and Leskovec [2016], Ribeiro et al. [2017] simulate variant random walks emphasizing different structural features of the graph. Celikkanat and Malliaros [2020] extend the models with different likelihoods with EFE framework; in their work, the context vectors are taken to represents the vertices.

A group of models have been proposed to learn multiple representations. Among those, Sun et al. [2019] decide the number of embedding with a community detection task; Liu et al. [2019], Park et al. [2020], Chen et al. [2020] impose a fixed number of embedding vectors for all nodes with a predefined value. The most similar method to ours is Epasto and Perozzi [2019] which uses local neighborhood clustering to generate multiple representations for nodes where different nodes can have different number of embedding vectors. Those methods often depend on extra simulations of the graph data in addition to the random walks data, whereas our method only requires the generated random walks.

Besides random walk based methods, there are other proposed approches include, for example, methods based on matrix factorization [Ou et al., 2016, Wang et al., 2017, Qiu et al., 2018] and neural network based approaches [Li et al., 2018, Velickovic et al., 2019, Wu et al., 2020].

3 PROPOSED MODEL

The proposed model is a Bayesian nonparametric extension of exponential family node embedding. We next describe the two notions and how they are used to learn multiple node representations. Figure 1 shows an overall illustration. In the figure, random walks are first extracted from a graph, yielding sequences whose sliding windows each contain a center node and counts of other nodes in the context. The occurrence of the center node will be modeled based on the context, where dependency is characterized using vectorial embeddings: each node has one embedding as a context node and can have multiple embeddings as a center node. The generation of the observed sequence content can be written as a graphical plate representation where nonparametric priors are used to generate the embedding vectors of center nodes, and the center and context embedding vectors together are used to generate observed values, that is, the observed center nodes in each window of a random walk.

3.1 EXPONENTIAL FAMILY NODE EMBEDDINGS

Given a simulated random walk node sequence $\mathbf{w} = \{w_1, \ldots, w_L\}$ of length L, we slide windows of length K along it. In each window the center node w_n is surrounded by context nodes $\{w_{n-K}, \ldots, w_{n-1}, w_{n+1}, \ldots, w_{n+K}\}$. For each possible vertex v we denote $x_{n,v} = 1$ if it was the center node so that $w_n = v$, otherwise $x_{n,v} = 0$. The context is denoted by the set $\mathbf{c}_n$ of unique vertices in the context nodes and the counts $\tilde{\mathbf{x}}_{\mathbf{c}_n} = \{\tilde{x}_{n,v'}\}$ how many times each vertex $v' \in \mathbf{c}_n$ occurred in them, $\tilde{x}_{n,v'} \leq K - 1$.

We will model dependency of node occurrences along a sequence, based on distributions whose natural parameter compares observed values to their context. In more detail, the natural parameter is based on comparison of node embedding vectors that characterize what kind of surroundings each node tends to appear in. We first describe the distribution and then describe the construction of the natural parameter for different exponential families (different likelihoods).

We model the co-occurrence pattern between w_n and the context $(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n})$ with an exponential family

$$x_{n,v}|\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n} \sim \mathbf{ExpFam}\left(\eta_n\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)\right) \quad (4)$$

where $\eta_v\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right)$ is the natural parameter and $T\left(x_{n,v}\right)$ the sufficient statistics.

In this work occurrence of a node is represented as a one-hot choice vector and it is modeled as a draw from an exponential family distribution whose parameters depend on the surrounding nodes. Concretely, if the vertex appears at the location n, the positive likelihood is then defined as

$$p(x_{n,v} = 1) = f(x_{n,v} = 1|\eta_n\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)) \quad (5)$$

where f is the corresponding probability density function of the exponential family distribution. For a vertex that does not appear at location n, the likelihood of the non-appearance (also called a 'negative likelihood') is

$$p(x_{n,v} = 0) = f(x_{n,v} = 0|\eta_n\left(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}\right), T\left(x_{n,v}\right)) \ . \quad (6)$$

Since random walks only yield positive samples of vertices that occurred in the center of their windows, learning from them alone would bias the model; thus we use a popular negative sampling approach, and randomly generate several negative samples (5 in experiments) for each location n. A negative sample has the same context $(\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n})$ as the positive sample at n, but $x_{n,v}$ is instead set to 1 for a random vertex among those that did not appear in the location.In this work, we explore three different exponential family distributions: Bernoulli, Poisson, and Gaussian.

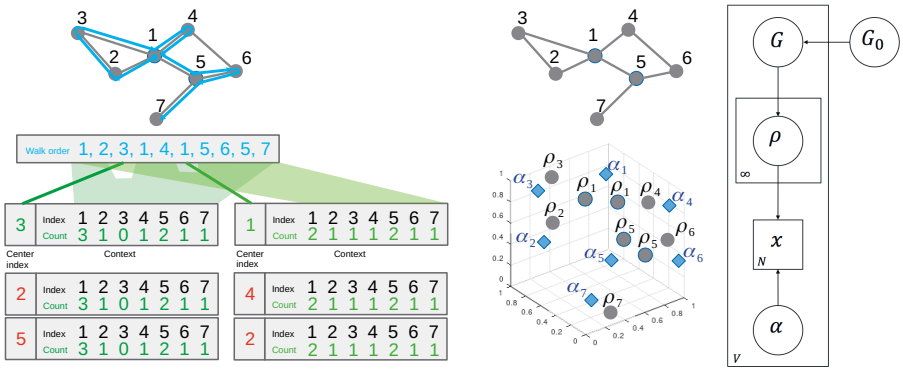

Figure 1: Illustrations of the proposed model. **Left:** random walk (light blue) along a graph from which windows are extracted as positive samples (green) of vertices that were center nodes and counts of other nodes in their context, and corresponding negative samples (red) of vertices that did not occur in the center. **Middle:** each vertex has one or more d-dimensional vector representations $\boldsymbol{\rho}$ as center nodes (circles), and one representation $\boldsymbol{\alpha}$ as a context node (diamonds). The picture shows a $d = 3$ dimensional example. **Right:** graphical plate representation of the proposed model.

Bernoulli Likelihood. We employ Bernoulli distribution to model the co-occurrence patterns of nodes. Let $\boldsymbol{\rho}_{n,v} \in \mathbb{R}^D$ denote the embedding vector of the node v at the location n, $\boldsymbol{\alpha}_v \in \mathbb{R}^D$ denote the embedding vector for the vertex v, the natural parameter is then defined as

$$p_n = \mathcal{S}\left(\boldsymbol{\rho}_{n,v}^\top \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \boldsymbol{\alpha}_{v'}\right) \tag{7}$$

where $\mathcal{S}$ denotes the sigmoid function $\mathcal{S} = \frac{1}{1+e^{-x}}$, and $|\mathbf{c}_n|$ is the number of distinct nodes in the context. The appearance of the node v at the location n, i.e. whether $x_{n,v} = 1$ or $x_{n,v} = 0$, is thus sampled from a Bernoulli distribution with parameter p_n so that

$$x_{n,v} \sim Bern(p_n) \; . \tag{8}$$

Note that we use the Bernoulli likelihood to model only the co-appearance of the nodes, which can be seen as an extension of Skip-gram based models. The number of occurrences of nodes in the context is not taken into the account. To incorporate the number of occurrences of nodes, we employ the Poisson and Gaussian distributions.

Poisson Likelihood. For a Poisson distribution, the parameter λ_n is defined as

$$\lambda_n = \exp\left(\boldsymbol{\rho}_{n,v}^\top \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \boldsymbol{\alpha}_{v'}\right) \tag{9}$$

where $|\mathbf{c}_n|$ is again the number of distinct nodes in context and $x_{n,v'}$ denotes the number of occurrences of node v' in the context. The appearance of the node v is generated as

$$x_{n,v} \sim Pois(\lambda_n) \tag{10}$$

The pivotal difference between the Bernoulli and Poisson cases is that the latter takes the number of occurrences of nodes in the context into account when constructing the natural parameter. The Gaussian case takes the same setting.

Gaussian Likelihood. Similar to the settings for Poisson Likelihood, the natural parameter here is defined as

$$\mu_n = \boldsymbol{\rho}_{n,v}^\top \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \boldsymbol{\alpha}_{v'} \tag{11}$$

without a specific link function, and the appearance of the node v at the location n is generated as

$$x_{n,v} \sim Norm(\mu_n, \sigma) \tag{12}$$

where we set σ as a fixed hyper-parameter; in the experiments we arbitrarily choose the σ from $\{1, 5, 10\}$.

When several different likelihoods are feasible, The model choice can depend on domain expertise, or cross-validation can be used as a model selection process.

3.2 NONPARAMETRIC EMBEDDING

Instead of restricting each vertex v to have a single role represented, to better capture the complexity of vertex roles in a graph as observed in random walks, we present a multiple representation learning model which enables each vertex to have multiple latent vector representations, so that the ocurrence of the the vertex at each location in a walk can arise from a different role of the vertex. To do so, we set a nonparametric prior on the embedding vectors $\boldsymbol{\rho}$. That is, we assume that at each location n, an embedding vector $\boldsymbol{\rho}_{n,v}$ is generated from a stochastic process G_v specific to

the vertex, so that

$$\boldsymbol{\rho}_{n,v} = \boldsymbol{\rho}_v^{(s)} \sim G_v(G_0, \gamma) \tag{13}$$

where G_v is a stochastic process with a base distribution G_0 and a concentration parameter γ. The base distribution G_0 has an infinite number of possible embedding vectors and G_v is a draw from it allocating nonzero probability to a finite number of possibilities $\{\boldsymbol{\rho}_v^{(1)}, \ldots, \boldsymbol{\rho}_v^{(s)}, \ldots, \boldsymbol{\rho}_v^{(S)}, \ldots, \}$ where S is the number of observed embedding vectors. We set the base distribution to be a d-dimensional Normal distribution $N(0, \sigma_0)$. In experiments we set $\sigma_0 = 5$ for Bernoulli likelihood and $\sigma_0 = 10$ for both Poisson and Gaussian likelihood. For simplicity, similar to the settings of Rudolph et al. [2017], Rudolph and Blei [2018], although we allow multiple embedding vectors $\boldsymbol{\rho}_{n,v}$ for a vertex we will use only one context vector $\boldsymbol{\alpha}_v$ per vertex; this setting can already generate good results in the experiments, and generalization to allow multiple context vectors is a future work.

In the following, let $\mathbf{n}_v = \mathbf{n}_v^+ \cup \mathbf{n}_v^-$ denote locations related to vertex v, so that $\mathbf{n}_v^+$ denotes locations where the v appears and $\mathbf{n}_v^-$ locations where v is the negative sample. Moreover, denote by $\mathbf{n}_{v,<n}$ the subset of $\mathbf{n}_v$ where the location is before n, and denote by superscript (s) those locations where the embedding vector was the s:th embedding vector of v.

Dirichlet Process. One of the most common nonparametric process priors is a Dirichlet process. The predictive probability of $\boldsymbol{\rho}_{n,v}$ is defined based on numbers of occurrences of embedding vectors of v at earlier locations $n' < n$ in positive or negative samples, so that

$$P(\boldsymbol{\rho}_{n,v} | \{\boldsymbol{\rho}_{n',v}; n' \in \mathbf{n}_{v,<n}\}) = \begin{cases} \frac{|\mathbf{n}_{v,<n}^{(s)}|}{\sum_{s'} |\mathbf{n}_{v,<n}^{(s')}| - 1 + \gamma} & \boldsymbol{\rho}_{v,n} = \boldsymbol{\rho}_v^{(s)},\ \forall \boldsymbol{\rho}_v^{(s)} \in \{\boldsymbol{\rho}_v^{(1)} \ldots \boldsymbol{\rho}_v^{(S_v)}\} \\ \frac{\gamma}{\sum_{s'} |\mathbf{n}_{v,<n}^{(s')}| - 1 + \gamma} & \boldsymbol{\rho}_{v,n} = \boldsymbol{\rho}_v^{(S_v+1)} \sim G_0 \end{cases} \tag{14}$$

where $|\mathbf{n}_{v,<n}^{(s)}|$ is the number of locations before n where $\boldsymbol{\rho}_v^{(s)}$ has been selected, and γ governs the generation of a new embedding vector.

Uniform process. An alternative to Dirichlet process is a uniform process [Wallach et al., 2010] with the predictive probability

$$P(\boldsymbol{\rho}_{n,v} | \{\boldsymbol{\rho}_{n',v}; n' \in \mathbf{n}_{v,<n}\}) = \begin{cases} \frac{1}{S_v + \gamma} & \boldsymbol{\rho}_{n,v} = \boldsymbol{\rho}_v^{(s)}, \forall \boldsymbol{\rho}_v^{(s)} \in \{\boldsymbol{\rho}_v^{(1)} \ldots \boldsymbol{\rho}_v^{(S_v)}\} \\ \frac{\gamma}{S_v + \gamma} & \boldsymbol{\rho}_{n,v} = \boldsymbol{\rho}_v^{(S_v+1)} \sim G_0 \end{cases} \tag{15}$$

where S_v denotes the number of different embedding vectors used for v before location n, and the embedding vector $\boldsymbol{\rho}_{n,v}$ is generated independently from the occurrence frequencies of previous generated values. The generation is only controlled by the concentration parameter γ.

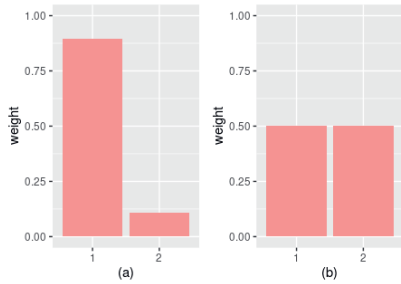

Figure 2: A comparison bewteen two nonparametric priors on the embeedings of the node [YGR078C] in Yeast dataset. (a): Weights of each embedding vector Dp-Pois model ($\gamma = 0.01$). (b): from up-Pois model ($\gamma = 0.000001$).

Despite the popularity of Dirichlet process, it suffers from the "rich get richer" issue, as it tends to repeat previous values and tends to model the first (or first few) embedding vectors as highly dominant, which can limit model flexibility. The uniform process was proposed to address this issue. Figure 2 show an example where the Dirichlet process concentrates on the first embedding vector and uniform process delivers smoother weights. The uniform process has been neglected by the research community, with most applications employing Dirichlet processes as priors.

Overall generative process. The proposed model can be summarized with the generative process shown below (corresponding plate model shown in Figure 1, Right):

1. For each vertex $v \in \mathbb{V}$:
   - $G_v \sim NP(G_0, \gamma)$
   - $\boldsymbol{\alpha}_v \sim N(0, \sigma_0^2 I)$
2. For each walk $\mathbf{w} = \{w_1, \ldots, w_L\} \in \mathcal{W}$
   - For location n:
     - $\boldsymbol{\rho}_{n,v} \sim G_v$
     - $\eta_{n,v} = g\left(\boldsymbol{\rho}_{n,v}^\top \frac{1}{|\mathbf{c}_n|} \sum_{v' \in \mathbf{c}_n} \tilde{x}_{n,v'} \boldsymbol{\alpha}_{v'}\right)$
     - $x_{n,v} \sim P(\eta_{n,v})$

4 INFERENCE

We adapt a truncation-free variational inference algorithm proposed by [Huynh et al., 2016]. Using a stick-breaking construction [Sethuraman, 1994], for vertex v we have

$$G_v = \sum_{s=1}^{\infty} \beta_v^{(s)} \delta_{\boldsymbol{\rho}_v^{(s)}}\ ,\ \boldsymbol{\rho}_v^{(s)} \sim G_0\ , \tag{16}$$

$$\beta_v^{(s)} = \zeta_v^{(s)} \prod_{i=1}^{s-1} \left(1 - \zeta_v^{(i)}\right),\ \zeta_v^{(s)} \sim Beta(1, \gamma)\ . \tag{17}$$

The posterior distribution for the stick breaking parameters $\boldsymbol{\beta}_v = (\beta_v^{(1)}, \ldots, \beta_v^{(S_v)}, \beta_v^{(S_v+1)})$ is then

$$(\beta_v^{(1)}, \ldots, \beta_v^{(S)}, \beta_v^{(S+1)}) \sim Dir(\theta_v^{(1)}, \ldots, \theta_v^{(S_v)}, \gamma) \quad (18)$$

where parameter $\boldsymbol{\theta}_v$ governs the general prevalence over all potential embedding vectors. For each location, the embedding vector $\boldsymbol{\rho}_{n,v}$ is decided by a label $z_{n,v}$ sampled from a Multinomial distribution

$$z_{n,v} \sim Multinomial(\boldsymbol{\beta}_v)\ ,\ \boldsymbol{\rho}_{n,v} = \boldsymbol{\rho}_v^{(z_{n,v})}\ . \quad (19)$$

The variational distribution $q(z_{v,n})$ is updated as

$$\exp\left(E_q\left[\ln z_{n,v}\right]\right) \propto \exp\left(E\left[\ln p(x_{n,v}|\mathbf{c}_n, \tilde{\mathbf{x}}_{\mathbf{c}_n}; \boldsymbol{\rho}_v^{(s)}, \boldsymbol{\alpha})\right] + E\left[\ln p(z_{n,v}|z_{\mathbf{n}_v \setminus n,v}; \gamma)\right]\right) \quad (20)$$

where the first term is the fitness of the selected embedding $\boldsymbol{\rho}_v^{(s)}$, and the second term is related to the prior. If the prior is a Dirichlet process, the second term in Equation (20) is

$$E\left[\ln p(z_{n,v}|z_{\mathbf{n}\setminus n,v}; \gamma)\right] = \begin{cases} \ln \frac{E[\theta^{(s)}_{\mathbf{n}_v \setminus n,v}]}{|\mathbf{n}_v|-1+\gamma} - \frac{1}{2}\frac{Var[\theta^{(s)}_{\mathbf{n}_v \setminus n,v}]}{E[\theta_v^{(-s)}]^2} & s \le S \\ \ln \frac{\gamma}{|\mathbf{n}_v|-1+\gamma} & s > S \end{cases} \quad (21)$$

where $\mathbf{n}_v$ denotes the locations of vertex v and $|\mathbf{n}_v|$ denotes its size. We then have

$$E[\theta_v^{(s)}] = \sum_{n \in \mathbf{n}_v} q(z_{n,v} = s) \quad (22)$$

$$E[\theta^{(s)}_{\mathbf{n}_v \setminus n,v}] = \sum_{n \in \mathbf{n}_v \setminus n} q(z_{n,v} = s) \quad (23)$$

$$Var[\theta_v^{(s)}] = \sum_{n \in \mathbf{n}_v} q(z_{n,v} = s)(1 - q(z_{n,v} = s)) \quad (24)$$

$$Var[\theta^{(s)}_{\mathbf{n}_v \setminus n,v}] = \sum_{n \in \mathbf{n}_v \setminus n} q(z_{n,v} = s)(1 - q(z_{n,v} = s)) \quad (25)$$

On the other hand, if the prior is a uniform process, the second term in Equation (20) has a simpler form:

$$E\left[\ln p(z_{n,v}|\gamma)\right] = \begin{cases} \ln \frac{1}{|\mathbf{n}_v|+\gamma} & s \le S \\ \ln \frac{\gamma}{|\mathbf{n}_v|+\gamma} & s > S \end{cases} \quad (26)$$

The truncation-free algorithm starts with setting $S = 1$, where $q(z_{v,n}^{(S+1)}) = 0$. When $E[\theta_v^{(S+1)}] > 1$, the algorithm sets $S = S + 1$, increasing the dimension of vector $z_{v,n}$, and sets $q(z_{v,n}^{(S+1)}) = 0$. We can then use the θ_v to calculate the expected weighting of the vector $\boldsymbol{\rho}_v^{(s)}$.

$$\hat{\beta}_v^{(s)} = E_q\left[\beta_v^{(s)}\right] = \frac{E_q\left[\theta_v^{(s)}\right]}{\sum_{s=1}^{S_v} E_q\left[\theta_v^{(s)}\right]} \quad (27)$$

Algorithm 1: Inference Algorithm

input : Random walks $\mathcal{W}$, negative samples $\tilde{\mathcal{W}}$, initial learning rate ξ, number of epochs, number of mini-batches M

output : embedding vectors $\Phi = \{\boldsymbol{\rho}, \boldsymbol{\alpha}\}$, embedding weights $\{\tilde{\boldsymbol{\beta}}\}$

```
foreach v ∈ 𝕍 do
    Set S_v = 1, initialize embedding vectors ρ_v^(1), α_v
end
foreach epoch do
    Divide input data into M random partitions.
    for m ← 1 to M do
        Use the subset W^(m) and W̃^(m)
        foreach v do
            foreach n ∈ n_v^(m) do
                update z_{n,v} with Equation (20)
            end
            updata θ_v with Equation (22) - (25)
            Calculate β̂_v with Equation (27)
            if E[θ_v^(S+1)] > 1 then
                S_v = S_v + 1
                foreach n ∈ n_v do
                    increase the dimension of z_{n,v} and
                      set z_{n,v}^(S+1) = 0
                end
            end
        end
        update embedding vectors Φ = {ρ, α}
        Φ = Φ − ξ * ∂L/∂Φ
        ξ is set with Adam[Kingma and Ba, 2015]
    end
end
```

Inference of embedding vectors.

After updating the $E_q\left[z_{n,v}\right]$, the inference is conducted by optimizing the objective function $\mathcal{L} = \mathcal{L}_{prior} + \mathcal{L}_{likelihood}$.

The term $\mathcal{L}_{prior} = \log p(\rho) + \log p(\alpha)$ is derived from the Gaussian prior $N(0, \sigma_0^2)$ for the embedding vectors:

$$\log p(\boldsymbol{\rho}_v^{(s)}) = \frac{\left\|\boldsymbol{\rho}_v^{(s)}\right\|^2}{-2\sigma_0^2}\ ,\ \ \log p(\boldsymbol{\alpha}_v) = \frac{\|\boldsymbol{\alpha}_v\|^2}{-2\sigma_0^2}\ . \quad (28)$$

Table 1: Datasets for Link Prediction

Data	$\|V\|$	$\|E\|$	Avg.deg	Density
GitHub	37700	289003	15.332	0.00041
Wikipedia	11631	180020	30.955	0.00266
Twitch	7126	35324	9.914	0.00140

Table 2: Datasets for Node Classification

Data	$\|V\|$	$\|E\|$	$\|K\|$	Avg.deg	Density
LastFM	7624	27806	18	7.294	0.00095
CiteSeer	3327	4237	6	2.845	0.00043
Yeast	2617	11855	13	9.060	0.00346

For Bernoulli likelihood we have

$$\mathcal{L}_{likelihood} = \sum_{v\in\mathbb{V}}\Big(\sum_{n\in\mathbf{n}_v^+}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right] p_n + \sum_{n\in\mathbf{n}_v^-}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right](1-p_n)\Big)\,. \quad (29)$$

For Poisson likelihood we have

$$\mathcal{L}_{likelihood} = \sum_{v\in\mathbb{V}}\Big(\sum_{n\in\mathbf{n}_v^+}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right](\log\lambda_n - \lambda_n) - \sum_{n\in\mathbf{n}_v^-}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right]\lambda_n\Big)\,. \quad (30)$$

For Gaussian likelihood, we have

$$\mathcal{L}_{likelihood} = \sum_{v\in\mathbb{V}}\Big(\sum_{n\in\mathbf{n}_v^+}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right]\left(\frac{(1-\mu_n)^2}{-2\sigma^2}\right) + \sum_{n\in\mathbf{n}_v^-}\sum_{s\in S_v} E_q\left[z_{n,v}=s\right]\left(\frac{\mu_n^2}{-2\sigma^2}\right)\Big)\,. \quad (31)$$

We then use gradient descent to update the embedding vectors over iterations.

4.1 STOCHASTIC INFERENCE

We employ stochastic inference. For each epoch, the input data is randomly partitioned into M mini-batches and only one mini-batch is used for each iteration. When mini-batch m is used, the sum over locations $\mathbf{n}_v$ can be approximated by a sum over a subsampled set $\mathbf{n}_v^{(m)}$, so the right-hand side of (22) is approximated by $\frac{|\mathbf{n}_v|}{|\mathbf{n}_v^{(m)}|}\sum_{n\in\mathbf{n}_v^{(m)}} q(z_{n,v}=s)$ and similarly in the other sums. The inference procedure is summarized in Algorithm 1. For all the experiments conducted in this work, we run two epochs with 1000 mini-batches and initial learning rate $\xi = 0.01$. For the negative samples, we generate $\tilde{\mathcal{W}}$ with 5 negative samples for each location following the procedure of Mikolov et al. [2013b], Celikkanat and Malliaros [2020].

5 EXPERIMENTS

For generality, we run experiments with two standard tasks commonly adopted in graph embedding works, link prediction and node classification, with 3 data sets [Csardi and Nepusz, 2006, Rossi and Ahmed, 2015, Rozemberczki et al., 2020] for each task (Tables 1 and 2). The data sets cover varied domains and aim to represent typical use scenarios of the proposed method. We denote our method variants by prior (dp: Dirichlet process, up: uniform process) and ExpFam distribution (Bern, Pois, Norm), e.g. 'up-emb (Bern)'. We compare to random walk based methods DeepWalk [Perozzi et al., 2014], node2Vec [Grover and Leskovec, 2016], struc2vec [Ribeiro et al., 2017], and EFGE [Celikkanat and Malliaros, 2020], and Splitter [Epasto and Perozzi, 2019]. To evaluate effect of embedding dimensionality, for each method we run three dimension settings: $D = 50, 100,$ and 150. The concentration parameter for our model is chosen from $\gamma = \{0.01, 0.05, 0.1\}$ for Dirichlet process and $\gamma = \{0.0000001, 0.0000005, 0.000001\}$ for uniform process. The input random walks are generated with the R package igraph [Csardi and Nepusz, 2006] with 80 walks per node with length $L = 10$, the random walks are also fed to EFGE. For other methods, parameters are all set to default values.

5.1 TASK: LINK PREDICTION

In link prediction, for each graph we first randomly move 50% of the edges into a held-out test set while keeping the remaining training graph connected. In both training and test sets, randomly sampled negative edges are added in equal amount to the positive edges. A classifier is trained based on the reduced training graph and the training negative edges; the classifier is used to classify the held-out test-set edges. As in the previous single-representation learning works including Deepwalk, node2vec, struc2vec, and EFGE, logistic regression is selected as the classifier. In our approach, to incorporate multiple representations when training the classifier, we employ logistic regression with sample weights, embedding $\boldsymbol{\rho}_v^{(s)}$ is weighted by $\hat{\beta_v^{(s)}}$. The Splitter used maximum dot-product similarity, we transform the similarity into a class probability using logistic regression.

Note that when logistic regression is trained with sample weighting, embeddings of all nodes in our model are separate samples weighted in the log-likelihood by their occurrence probabilities. The regression learns to classify nodes based on all their embedding vectors, and at test time, a node is classified by weighted average of class probabilities predicted for each of its embedding vectors. Thus, the

Table 3: Results for Link Prediction

	GitHub			Wikipedia			Twitch		
	D = 50	D = 100	D = 150	D = 50	D = 100	D = 150	D = 50	D = 100	D = 150
Deepwalk	0.722	0.695	0.694	0.911	0.915	0.922	0.659	0.649	0.672
node2vec	0.731	0.734	0.731	0.913	0.931	0.941	0.681	0.691	0.698
struc2vec	0.849	0.864	0.874	0.820	0.881	0.863	0.830	0.828	0.840
EFGE (Bern)	0.729	0.726	0.736	0.939	0.950	0.962	0.681	0.687	0.707
EFGE (Pois)	0.728	0.771	0.771	0.950	0.955	0.964	0.679	0.708	0.714
EFGE (Norm)	0.862	0.868	0.888	0.977	0.983	0.985	0.791	0.791	0.802
Splitter	0.898	0.600	0.900	0.876	0.880	0.884	0.836	0.823	0.823
dp-emb (Bern)	0.823	0.831	0.830	0.986	0.991	0.991	0.757	0.787	0.782
dp-emb (Pois)	0.737	0.723	0.780	0.979	0.984	0.986	0.656	0.704	0.716
dp-emb (Norm)	0.923	**0.932**	0.929	0.985	0.985	0.985	0.847	0.845	**0.871**
up-emb (Bern)	0.813	0.838	0.843	**0.989**	**0.991**	**0.992**	0.750	0.788	0.784
up-emb (Pois)	0.741	0.767	0.780	0.979	0.982	0.986	0.658	0.706	0.714
up-emb (Norm)	**0.926**	0.932	**0.931**	0.985	0.985	0.986	**0.849**	**0.846**	0.869

multiple embedding vectors are treated separately instead of being combined in a simplistic weighted average.

Three different datasets are used for the link prediction task.

GitHub: a social network where each node is a GitHub developer, links between nodes are mutual follow relations.
Wikipedia: a network of English Wikipedia pages. Edges between pages reflect their mutual links.
Twitch: a user-user interaction network between gamers. Edge between two nodes represents mutual friendship.

We evaluate the binary link classification by area under the curve (AUC). Table 3 shows our model performs well on all datasets; the model with Gaussian likelihood works best.

5.2 TASK: NODE CLASSIFICATION

In this task, each node has a class. The learned embedding vectors are used as input features to train a classifier to predict the class of each node. Again, for Deepwalk, node2vec, struc2vec and EFGE, a logistic regression classifier is used. For our model, the logistic regression with sample weights is used. For Splitter, we take the same procedure with each embedding equally weighted. Three different datasets are used for the node classification task.

LastFM Asia: a network of people living in Asia using the streaming site LastFM. Links represent followership relations. The class of each node is its location.
CiteSeer: a scientific publication network from the CiteSeer digital library. Each node belongs to 1 of 6 categories.
Yeast:a protein-protein interaction network. The "Class" attribute of each protein is based on its function (e.g. energy).

We evaluate the performance by Micro-averaged F1, reported in Table 4. Our model outperforms other methods. Rozemberczki et al. [2020] Additionaly, in general, our model took 2-4 hours to converge (depends on different tasks and settings) without GPU. The Splitter, which also learns multiple representations for each node, took 10+ hours on a GPU machine and 100+ hours without GPU. Our approach achieved better results with less resources.

6 DISCUSSIONS AND CONCLUSIONS

We proposed nonparametric exponential family graph embedding, allowing multiple node representations, drawn both with a Dirichlet process prior, and also exploring uniform processes. A tailored algorithm for efficient computation is provided. The experiments demonstrate the learned multiple representations can enhance performance in two tasks. We considered three classical exponential family distributions, Bernoulli, Poisson, and Gaussian, which yielded promising results. Our model can be adapted to other distributions such as Geometric and Chi-square with the proposed nonparametric framework. In our experiments, the hyperparameter γ of the nonparametric prior was fixed for the nodes, which already yielded promising results in the standard tasks; having differing γ values could be useful for extending the model to scenarios such as learning multiple representations for under-represented nodes, or imbalanced classification tasks.

Acknowledgements

This work is supported by the Academy of Finland, decisions 312395 and 327352.

References

Abdulkadir Celikkanat and Fragkiskos D Malliaros. Exponential family graph embeddings. In *Proceedings of the*

Table 4: Results for Node Classification

LastFM	(D = 50)				(D = 100)				(D = 150)			
	10%	30%	60%	90%	10%	30%	60%	90%	10%	30%	60%	90%
Deepwalk	0.756	0.800	0.819	0.823	0.754	0.796	0.819	0.829	0.750	0.797	0.819	0.826
node2vec	0.741	0.796	0.820	0.828	0.741	0.802	0.824	0.829	0.740	0.799	0.826	0.834
struc2vec	0.116	0.127	0.130	0.138	0.128	0.149	0.165	0.174	0.131	0.159	0.178	0.189
EFGE-Bern	0.749	0.805	0.826	0.831	0.758	0.805	0.824	0.830	0.758	0.803	0.826	0.832
EFGE-Pois	0.741	0.791	0.820	0.825	0.743	0.793	0.817	0.822	0.745	0.798	0.821	0.825
EFGE-Norm	0.758	0.807	0.826	0.832	0.752	0.804	0.824	0.830	0.755	0.808	0.827	0.833
Splitter	0.428	0.519	0.541	0.546	0.426	0.490	0.530	0.573	0.451	0.469	0.533	0.567
Dp-Bern	**0.809**	**0.833**	**0.839**	0.833	**0.810**	**0.835**	**0.846**	0.849	0.800	**0.835**	0.843	**0.850**
Dp-Pois	0.776	0.821	0.831	0.833	0.782	0.822	0.831	0.830	0.782	0.823	0.832	0.833
Dp-Norm	0.751	0.807	0.822	0.823	0.740	0.804	0.820	0.821	0.744	0.807	0.823	0.831
up-Bern	0.806	0.831	0.835	**0.841**	0.802	0.835	0.841	**0.852**	0.804	0.833	**0.844**	0.844
up-Pois	0.781	0.818	0.828	0.829	0.802	0.835	0.841	0.852	0.779	0.821	0.830	0.834
up-Norm	0.754	0.811	0.822	0.823	0.742	0.805	0.821	0.823	0.733	0.806	0.821	0.827
Citeseer	(D = 50)				(D = 100)				(D = 150)			
Deepwalk	0.432	0.479	0.487	0.519	0.453	0.497	0.520	0.530	0.459	0.504	0.525	0.532
node2vec	0.456	0.503	0.508	0.555	0.493	0.529	0.539	0.544	0.501	0.538	0.570	0.582
struc2vec	0.224	0.240	0.278	0.314	0.226	0.250	0.274	0.294	0.224	0.243	0.254	0.297
EFGE-Bern	0.468	0.502	0.508	0.518	0.477	0.503	0.516	0.556	0.478	0.520	0.532	0.580
EFGE-Pois	0.460	0.504	0.501	0.518	0.497	0.490	0.491	0.562	0.497	0.491	0.499	0.566
EFGE-Norm	0.456	0.496	0.503	0.526	0.473	0.500	0.516	0.533	0.471	0.505	0.533	0.581
Splitter	0.164	0.162	0.183	0.181	0.169	0.166	0.188	0.186	0.165	0.162	0.166	0.177
Dp-Bern	0.461	0.481	0.528	**0.589**	0.478	0.519	0.533	0.563	0.504	0.540	0.564	0.559
Dp-Pois	0.435	0.462	0.479	0.538	0.430	0.460	0.476	0.528	0.403	0.441	0.460	0.510
Dp-Norm	0.475	0.490	0.510	0.556	0.509	0.523	0.529	**0.559**	0.512	0.529	0.527	0.562
up-Bern	0.459	0.492	0.509	0.538	0.481	0.522	**0.534**	0.538	0.502	0.557	0.581	0.585
up-Pois	0.437	0.465	0.498	0.540	0.436	0.467	0.492	0.546	0.404	0.438	0.473	0.529
up-Norm	**0.518**	**0.527**	**0.532**	0.568	**0.521**	**0.531**	0.514	0.561	**0.521**	**0.559**	**0.580**	**0.616**
Yeast	(D = 50)				(D = 100)				(D = 150)			
Deepwalk	0.283	0.330	0.360	0.413	0.290	0.358	0.401	0.436	0.288	0.361	0.400	0.441
node2vec	0.280	0.320	0.351	0.388	0.293	0.338	0.371	0.410	0.297	0.354	0.401	0.437
struc2vec	0.134	0.150	0.169	0.256	0.134	0.153	0.166	0.238	0.141	0.161	0.171	0.225
EFGE-Bern	0.269	0.324	0.347	0.380	0.281	0.339	0.374	0.418	0.289	0.349	0.400	0.414
EFGE-Pois	0.271	0.320	0.365	0.373	0.281	0.331	0.372	0.399	0.286	0.339	0.374	0.409
EFGE-Norm	0.285	0.325	0.354	0.383	0.281	0.332	0.367	0.405	0.288	0.354	0.392	0.428
Splitter	0.164	0.207	0.228	0.246	0.157	0.214	0.263	0.263	0.165	0.211	0.273	0.297
Dp-Bern	0.285	**0.343**	0.373	0.401	0.296	0.377	0.402	0.442	0.296	**0.376**	0.416	0.472
Dp-Pois	0.275	0.328	0.354	0.375	0.285	0.327	0.360	0.383	0.301	0.338	0.375	0.402
Dp-Norm	0.285	0.330	0.364	0.352	0.277	0.339	0.352	0.381	0.266	0.350	0.382	0.407
up-Bern	**0.290**	0.338	**0.382**	**0.414**	**0.301**	**0.361**	**0.406**	**0.443**	**0.304**	0.367	**0.419**	**0.479**
up-Pois	0.281	0.336	0.358	0.392	0.288	0.326	0.355	0.385	0.277	0.348	0.395	0.426
up-Norm	0.282	0.340	0.372	0.393	0.289	0.345	0.391	0.381	0.288	0.320	0.364	0.382

AAAI Conference on Artificial Intelligence, volume 34, pages 3357–3364, 2020.

Yujun Chen, Juhua Pu, Xingwu Liu, and Xiangliang Zhang. Gaussian mixture embedding of multiple node roles in networks. *World Wide Web*, 23(2):927–950, 2020.

Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006. URL `https://igraph.org`.

Alessandro Epasto and Bryan Perozzi. Is a single embedding enough? learning node representations that capture multiple social contexts. In *The world wide web conference*, pages 394–404, 2019.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

Viet Huynh, Dinh Phung, and Svetha Venkatesh. Streaming variational inference for dirichlet process mixtures. In *Asian Conference on Machine Learning*, pages 237–252. PMLR, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Thirty-Second AAAI conference on artificial intelligence*, 2018.

Ninghao Liu, Qiaoyu Tan, Yuening Li, Hongxia Yang, Jingren Zhou, and Xia Hu. Is a single vector enough? exploring node polysemy for network embedding. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 932–940, 2019.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013b.

Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1105–1114, 2016.

Chanyoung Park, Carl Yang, Qi Zhu, Donghyun Kim, Hwanjo Yu, and Jiawei Han. Unsupervised differentiable multi-aspect network embedding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1435–1445, 2020.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

Jiezhong Qiu, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang. Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 459–467, 2018.

Leonardo FR Ribeiro, Pedro HP Saverese, and Daniel R Figueiredo. struc2vec: Learning node representations from structural identity. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 385–394, 2017.

Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL `https://networkrepository.com`.

Benedek Rozemberczki, Oliver Kiss, and Rik Sarkar. Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*, page 3125–3132. ACM, 2020.

Maja Rudolph and David Blei. Dynamic embeddings for language evolution. In *Proceedings of the 2018 World Wide Web Conference*, pages 1003–1011, 2018.

Maja Rudolph, Francisco Ruiz, Stephan Mandt, and David Blei. Exponential family embeddings. In *Advances in Neural Information Processing Systems*, pages 478–486, 2016.

Maja Rudolph, Francisco Ruiz, Susan Athey, and David Blei. Structured embedding models for grouped data. *Advances in neural information processing systems*, 30, 2017.

Jayaram Sethuraman. A constructive definition of dirichlet priors. *Statistica sinica*, pages 639–650, 1994.

Fan-Yun Sun, Meng Qu, Jordan Hoffmann, Chin-Wei Huang, and Jian Tang. vgraph: A generative model for joint community detection and node representation learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *ICLR (Poster)*, 2(3):4, 2019.

Hanna Wallach, Shane Jensen, Lee Dicker, and Katherine Heller. An alternative prior process for nonparametric bayesian clustering. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 892–899. JMLR Workshop and Conference Proceedings, 2010.

Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiqiang Yang. Community preserving network embedding. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

PUBLICATION VI

Gaussian Copula Embeddings

Chien Lu and Jaakko Peltonen

In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022

Gaussian Copula Embeddings

Chien Lu **Jaakko Peltonen**
Tampere University

Abstract

Learning latent vector representations via embedding models has been shown promising in machine learning. However, most of the embedding models are still limited to a single type of observed data. We propose a Gaussian copula embedding model to learn latent vectorial representations of items in a heterogeneous-data setting. The proposed model can effectively incorporate different types of observed data and, at the same time, yield robust embeddings. We demonstrate that the proposed model can effectively learn in many different scenarios, outperforming competing models in modeling quality and task performance.

1 Introduction

Representation learning is a prominent machine learning approach for working with originally non-vectorial data. Embedding models learn latent vectorial representations for data items that appear together with a context, through modeling the interactions between each center item and its context items. The approach was first introduced as a language model [14] which learns word representations through modeling the probability of the appearance of a central word given surrounding context words. The word appearance is modeled as an observation from a multinomial word distribution. Building on this notion, exponential family embeddings were proposed [19] which further generalized the original model to a class of models suitable for many observed data types, which have been shown promising in different domains. However, the ability of such models to incorporate heterogeneous data is still limited.

Data in many modern domains is heterogeneous, involving simultaneous observation of different data types such as categorical values, integers and real-valued numbers, and having varied distributions within each data type, hence it is difficult to model them together as observations in vectorial embedding; naive unified solutions that ignore the difference of the data types would not yield good models. In particular, the different data types are often distributed over varied scales and with various distributional shapes: naive normalization coupled to modeling with a single distributional assumption would not suffice to yield robust embedding models, and would leave them vulnerable to extreme values and distributional shapes not corresponding to the assumed ones. An equally pressing problem is how to flexibly model relationships (dependencies) between the several observed variables with their differing distributions: naive modeling strategies ignoring the variable dependencies would again yield poor embedding models.

We solve the mentioned challenges by introducing a novel Gaussian copula based latent representation learning model. The model learns vectorial embedding representations for items leveraging the center-context item interactions, but unlike previous embedding models the proposed model is able to learn embeddings in a setting with multivariate data having heterogeneous data types and distributions.

For computational efficiency, we introduce a set of variational auto-encoder based inference algorithms. In experiments on five different scenarios, the proposed model is shown to be effective, outperforming competitive methods in task-based evaluations and yielding insights in a social media analysis task.

Our contributions are:

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

- We introduce a general-purpose representation learning framework which incorporates multiple, heterogeneous data. Our work allows embedding models to include multiple data types and distributional assumptions in well-founded probabilistic joint modeling through the Gaussian copula; other approaches such as exponential family embeddings have generalised to different data types but still treat them individually.
- To our knowledge, ours is the first work which brings the advantages of Gaussian copula to learning representation vectors from heterogeneous data. The Gaussian copula is intuitive and has proven effective in machine learning research, thus it is an attractive solution which was neglected in representation learning. We close this gap, and the result shows it handles heterogeneous data well.
- We develop an efficient inference procedure based on semiparametric estimation and variational autoencoder. Previous works have used MCMC for inference and in many such works, lack of scalability has limited their application to larger amounts of data.
- We demonstrate the effectiveness of our model on five different scenarios, each with a real-world data set and corresponding quantitative and qualitative evaluation.

This paper is organized as follows. Section 2 introduces the necessary background notions of embedding models and copula models. Section 3 introduces the Gaussian copula model and Section 4 develops the inference algorithms. Five different scenarios of using the proposed model are provided 5, each evaluated with an experiment on a real-world data set. Section 6 draws the conclusions.

2 Background

2.1 Embedding models

Learning latent representations based on the interactions between the observed item and its contexts has been an imperative topic in machine learning. It was first introduced as a language model to model relations between words [14]. The framework has been later generalized to model other different co-appearance patterns such as in community embedding [23].

In brief, let the item i and its context j be two items (such as two words, or two communities). The probability of them appearing (D = 1) in the same context (such as in the same sliding window) can be modeled as

$$P(D = 1|i, j) = \left(\frac{1}{1 - e^{-\boldsymbol{\rho}_i{}^\top \boldsymbol{\rho}_j}}\right) \tag{1}$$

where $\boldsymbol{\rho}_i$ and $\boldsymbol{\rho}_j$ are the embedding vectors of the items. Note that instead of using the same kinds of vectors for both roles, the context items can have their own context vectors $\boldsymbol{\alpha}$, thus the above probability becomes

$$P(D = 1|i, j) = \left(\frac{1}{1 - e^{-\boldsymbol{\rho}_i{}^\top \boldsymbol{\alpha}_j}}\right) . \tag{2}$$

Exponential Family Embeddings [19] is an extension which has further generalized the model to data beyond text. Let $x_n^{(i)}$ be the value of the item i at the location n, which has its context $\mathbf{c}_n$. In exponential family embedding, the value of x_n depends on its context $\mathbf{c}_n$ and is generated from an exponential family distribution

$$x_n^{(i)}|\mathbf{c}_n \sim \mathbf{ExpFam}\left(\eta_n\left(\boldsymbol{x}_{\mathbf{c}_n}\right), t\left(x_n\right)\right) \tag{3}$$

where $\eta_n\left(\boldsymbol{x}_{\mathbf{c}_n}\right)$ is the natural parameter, and $t\left(x_{n,v}\right)$ denotes the sufficient statistics. The natural parameter is modeled as a function of an inner product of the embedding vector $\boldsymbol{\rho}$ and the context vector $\boldsymbol{\alpha}$ so that

$$\eta_n\left(\boldsymbol{x}_{\mathbf{c}_n}\right) = g\left(\boldsymbol{\rho}_i^\top \frac{1}{|\mathbf{c}_n|} \sum_{n' \in \mathbf{c}_n} x_{n'}^{(i')} \boldsymbol{\alpha}_{i'}\right) . \tag{4}$$

As the exponential family can model different observation distributions, the embedding models are no longer limited to modeling co-appearance (binary) observations. It has been applied to different domains such as grouped data [18] and graph data [1].

Negative sampling or sub-sampling is a common practice when training embedding models. The notion is to consider only a randomly generated subset of the items that do not occur at the location n. That is, if an item i does not occur at a location, the probability of that negative occurrence $P(D = 0|i, j) = \left(\frac{1}{1-e^{-\rho_i{}^\top \alpha_j}}\right)$ is integrated into the objective function. In exponential family embeddings, if the item i is generated as a negative sample, the corresponding pseudo observed value is encoded as $x_n^{(i)} = 0$.

2.2 Gaussian copula

A J-dimensional copula $\mathbb{C}$ is a probability distribution on $[0, 1]^J$ where each of its univariate marginal distributions is a uniform distribution on $[0, 1]$. That is, given a set of uniform distributed random variables $U_1, \ldots, U_J$, a copula is the joint cumulative distribution

$$\mathbb{C}(u_1, \ldots, u_J) = P(U_1 \leq u_1, \ldots, U_J \leq u_J) \ . \tag{5}$$

The key idea of copula modeling is to use the copula to model the dependencies between several variables having arbitary types and marginal distributions. Let $\mathbf{x}$ be a random vector of length J, and let $j \in 1 \ldots J$ index the elements (random variables) in $\mathbf{x}$. According to Sklars' theorem [21], the cumulative distributions (CDFs) of the variables in $\mathbf{x}$ can be modeled by a copula

$$F(x_1, \ldots x_J) = \mathbb{C}\left(F_1(x_1), \ldots, F_J(x_J)\right) \tag{6}$$

where F is the joint CDF and $F_j(x) = P(X_j \leq x)$ is the j-th marginal CDF. Since each marginal CDF value is in $[0, 1]$, the right-hand side is a copula regardless of what distributions the individual marginal CDFs have. If every F_j is continuous, then the $\mathbb{C}$ is unique. In this way, the copula encodes the structure of variable dependencies, while allowing each of the variables $x_1 \ldots x_J$ to be of a different type and to have differing kinds of marginal CDFs.

In this paper, we consider a Gaussian copula, which is one of the widely used copula models; we introduce the model to the representation learning task. A Gaussian copula is defined as

$$\mathbb{C}(u_1, \ldots u_J) = \Phi_J\left(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_J)|\mathbf{C}\right) \tag{7}$$

where Φ_J is a J-dimensional Gaussian CDF with a correlation matrix $\mathbf{C}$, and Φ^{-1} is the inverse function of the standard univariate Gaussian CDF. With the Gaussian copula, the joint CDF of observed data can be modeled as

$$F(x_1, \ldots x_J) = \mathbb{C}(F_1(x_1), \ldots F_J(x_J)) = \Phi_J\left(\Phi^{-1}(F_1(x_1)), \ldots, \Phi^{-1}(F_J(x_J))|\mathbf{C}\right) \ . \tag{8}$$

The Gaussian copula can also be expressed in terms of a latent Gaussian variable representation. First, a latent vector $\mathbf{z}$ is generated from a Gaussian distribution

$$\boldsymbol{z} \sim \boldsymbol{N}(0, \Omega) \tag{9}$$

wih covariance matrix Ω which corresponds to the correlation matrix $\mathbf{C}$ in equation (7). Then for each j, the observed data value x_j is obtained from the inverse of the univariate marginal F_j^{-1} according to the generated latent variable z_j so that

$$x_j = F_j^{-1}\left(\Phi\left(\frac{z_j}{\sqrt{\omega_{jj}}}\right)\right) \tag{10}$$

where ω_{jj} is the j-th diagonal element of Ω. In principle, equations (9) and (10) could be used to derive an equation for the likelihood of the data values x_j at each observation, in order to use such likelihoods for parameter fitting. However, instead the practical approach in Gaussian copula research is to derive an *extended likelihood*, and we will do that for the vectorial embedding task.

3 Gaussian Copula Embeddings

We now introduce Gaussian Copula Embeddings (GCE), which perform representation learning for data with heterogeneous-type observations of items in contexts. Let $\{\mathbf{x}_1 \ldots \mathbf{x}_N\}$ be the observations.

Each observation $\mathbf{x}_n$ is of a particular item, occurring in a context; moreover, each occurrence of the item is observed with multiple feature values (variables). Let $\mathbf{x}_n = \boldsymbol{x}_n^{(i)}$ denote that the item i occurs at location n and is observed carrying J variables $\boldsymbol{x}_n^{(i)} = [x_{n,1}^{(i)}, \ldots, x_{n,J}^{(i)}]^\top$, where the location n comes with a context $\boldsymbol{c}_n$ which contains a collection of context item indices.

The principle of the GCE model is that the multiple heterogeneous observations will be generated based on the relationship of the central item i to its context, characterized by several embedding vectors, and the dependencies between the observations will be characterized by a Gaussian copula. For each central item i there is an underlying embedding vector $\boldsymbol{\rho}_i \in \mathbb{R}^{K\times 1}$. In addition to occurring as a central item, each item may also occur as part of a context. The context $\boldsymbol{c}_n$ of the location n will contain several items i'. The roles of the items in the context will be characterized by context vectors: unlike a traditional embedding model that generates only one type of observation, here each context item i' has a set of variable specific context vectors $\{\boldsymbol{\alpha}_{i',j} | j \in 1, \ldots J\}$.

We develop the GCE model based on the latent representation equations (9) - (10). For each item i the underlying embedding vector $\boldsymbol{\rho}_i \in \mathbb{R}^{K\times 1}$ is generated from a multivariate normal distribution

$$\boldsymbol{\rho}_i \sim \boldsymbol{N}(0, \boldsymbol{I}) \; . \tag{11}$$

Following the latent variable representation of Gaussian copula, all J observations of an item at a location will be generated based from a latent vector. The latent variable vector $\mathbf{z}_n$ is generated as

$$\boldsymbol{z}_n^{(i)} \sim \boldsymbol{N}(0, \mathbf{I} + \boldsymbol{R}_n \boldsymbol{R}_n{}^\top) \Longleftrightarrow \boldsymbol{z}_n^{(i)} \sim \boldsymbol{N}(\boldsymbol{R}_n \boldsymbol{\rho}_i, \boldsymbol{I}) \tag{12}$$

where the matrix $\boldsymbol{R}_n \in \mathbb{R}^{K\times J}$ is constructed based on the embedding vectors of the items in the context $\boldsymbol{c}_n$ for all observation variables. For each observation variable j, the corresponding column $\boldsymbol{r}_{n,j}$ of the matrix $\boldsymbol{R}_n$ is constructed as

$$\boldsymbol{r}_{n,j} = \frac{1}{|\boldsymbol{c}_n|} \sum_{i' \in \boldsymbol{c}_n} \boldsymbol{\alpha}_{i',j} \tag{13}$$

where i' are the items in the context of the location n, $\boldsymbol{c}_n$. Here, for simplicity, we set the prior of $\boldsymbol{\alpha}$ to be a multivariate normal distribution

$$\boldsymbol{\alpha}_{i',j} \sim \mathbf{N}(0, \lambda_\alpha^{-1} \mathbf{I}) \tag{14}$$

with a diagonal covariance matrix where the constant λ_α is a precision parameter which controls the constraints on $\boldsymbol{\alpha}$. Using the exchangeability in equation (12), the above generating process can be also written as

$$\boldsymbol{z}_n^{(i)} \sim \boldsymbol{N}(\boldsymbol{\mu}_n^{(i)}, \boldsymbol{I}), \text{ where } \boldsymbol{\mu}_n^{(i)} = [\mu_{n,1}^{(i)}, \ldots, \mu_{n,J}^{(i)}] \text{ and } \mu_{n,j}^{(i)} = \boldsymbol{\rho}_i{}^\top \frac{1}{|\boldsymbol{c}_n|} \sum_{i' \in \boldsymbol{c}_n} \boldsymbol{\alpha}_{i',j} \; . \tag{15}$$

The observations are then obtained from the latent variables according to the Gaussian copula equations. The jth observed value $x_{n,j}^{(i)}$ is obtained as

$$x_{n,j}^{(i)} = F_j^{-1}\left(\Phi\left(\frac{z_{n,j}^{(i)}}{\sqrt{1 + \sum_{k=1}^{K} r_{n,j,k}^2}}\right)\right) \tag{16}$$

where $z_{n,j}^{(i)}$ is the jth element of the latent vector $\boldsymbol{z}_n^{(i)}$ and $r_{n,j,k}$ is the kth dimension of the context representation column $\boldsymbol{r}_{n,j}$, and F_j^{-1} is the inverse CDF of the marginal distribution of variable j. Inference of the embedding parameters based on the observations will be done with an extended likelihood approach introduced in the next section.

The major difference between our approach and other embedding models is that the GCE can take heterogenous, multivariate observed data into account. Another difference is that GCE further specializes the roles of embedding vectors $\boldsymbol{\rho}$ and context vectors $\boldsymbol{\alpha}$: in traditional exponential family embeddings their roles can be seen as ambiguous, having the same form and a very similar place in the generative equations. In contrast, here the roles are made distinct: the $\boldsymbol{\rho}$ are used to model a general representation of each item as a central item, and the variable-specific context vectors $\boldsymbol{\alpha}_j$ govern the role of each item as a context item for the multiple variables, telling how the different variables interact with the central item as well as controlling the dependencies between the variables.

4 Inference

The task of the inference is to fit the embedding parameters $\boldsymbol{\rho}$ and $\boldsymbol{\alpha}_j$ of all items to the observations. We will use a variational inference approach; a key aspect of it is to replace direct evaluation of the likelihood by an *extended likelihood* approach described next.

4.1 Extended Likelihood

As pointed out by [7], the naive inverse of equation (16) during the inference is only recommended when the observed data is continuous and follows an easy-to-invert parametric distribution. Moreover, directly inverting the CDF for discrete variables can lead to pushing the latent variables to extremes and affect the validity of the inference. Therefore, an extended rank likelihood has been proposed for the inference of Gaussian copula.

Consider all observations $x_{n,j}$ of variable j at all locations $n = 1, \ldots, N$, and the corresponding latent variables $z_{n,j}$ where for brevity we drop the item indices i; each observation may arise from a different item. Denote the latent variables together by a vector $\mathbf{z}_j$ and observed variables together by vector $\mathbf{x}_j$. Since each $z_{n,j}$ is related to the corresponding observed $x_{n,j}$ by two monotonic functions (a gaussian CDF and an inverse CDF), the rank order of the $z_{n,j}$ is the same as that of the $x_{n,j}$. The vector $\mathbf{z}_j$ is one of a set $\mathbf{D}(\mathbf{x}_j)$ of vectors having that rank order:

$$\mathbf{z}_j \in \mathbf{D}(\mathbf{x}_j) = \{\mathbf{z}_j \in \mathbb{R}^N : x_{n,j} < x_{n',j} \Rightarrow z_{n,j} < z_{n',j}\} \,. \tag{17}$$

The $\mathbf{D}(\mathbf{x}_j)$ is the set of possible $\mathbf{z}_j = (z_{1,j}, \ldots, z_{n,j})$ which preserve the ordering of the observed data [20]. Let $\mathbf{D} = \{\mathbf{Z} \in \mathbb{R}^{J \times N} : \mathbf{z}_j \in \mathbf{D}(\mathbf{x}_j)\ \forall 1 \leq j \leq J\}$ be the set of possible latent variable combinations, such that the rank order is satisfied for each observed variable j. Note that ties may happen in the rank orders, for example when discrete variables yield the same value at multiple observations.

In the Gaussian copula model the observed variables are directly obtained as unique transformations of the latent variables. Because the possible latent variables must satisfy the rank orders of the observations, the rank perservation can be inserted into the full likelihood which can then be factorized as

$$\begin{aligned} P(\mathbf{X}|\mathbf{C}, F_1, \ldots, F_J) &= P(\mathbf{X}, \mathbf{Z} \in \mathbf{D}|\mathbf{C}, F_1, \ldots, F_J) \\ &= P(\mathbf{Z} \in \mathbf{D}|\mathbf{C}) \times P(\mathbf{X}|\mathbf{Z} \in \mathbf{D}, \mathbf{C}, F_1, \ldots, F_J) \,. \end{aligned} \tag{18}$$

The $P(\mathbf{Z} \in \mathbf{D}|\mathbf{C})$ is then taken as the alternative likelihood. It has been proved [8] that it shares the same information bound as using estimator of the full data. Ranks also bring an advantage of robustness as they are unaffected by precise value and thus are less prone to outliers. Using an extended likelihood based on the rank of observed data is the current state of the art practice for Gaussian copula inference [15, 3].

4.2 Amortized variational autoencoder

Most of the previous works taking the extended likelihood use Gibbs sampling for inference [15, 3]. Despite some recent improvement such as [9], the sampling process inevitably must update the latent variables for every location n, which makes the computation inefficient when the volume of data grows large. To avoid this inefficiency, we develop a stochastic variational inference procedure exploiting the idea of amortized inference [4] and a variational autoencoder [11].

Since the essence of the estimator using $P(\mathbf{Z} \in \mathbf{D}|\mathbf{C})$ is to keep the ranking of $\mathbf{z}_j$ corresponding to the ranking of $\boldsymbol{x}_j$, we then employ the Plackett-Luce model with ties [24] as an alternative likelihood. Let $r(x_{n,j})$ denote the rank of $x_{n,j}$ and $\boldsymbol{r}(\boldsymbol{x}_j)$ denote the vector of rankings corresponding to $\boldsymbol{x}_j$. We then have

$$p(\boldsymbol{r}(\boldsymbol{x}_j)|\mathbf{z}_j) = \prod_q \left(\frac{e^{z_{n,j}}}{\sum_{n' \in C_q} e^{z_{n',j}}} \right)^{\frac{1}{|A_q|}} \tag{19}$$

where q in the product goes over the rank positions, $A_q = \{n : r(x_{n,j}) = q\}$ denotes the items ranked at position q (there may be more than one due to ties), and $C_q = \{n : r(x_{n,j}) \geq q\}$ are items ranked at q or higher. Note that if there are no ties, equation (19) reduces to the standard

Plackett-Luce distribution. The ties are considered in order to handle discrete data. The likelihood for all variables j is then simply $\prod_j \log p(\boldsymbol{r}(\boldsymbol{x}_j)|\boldsymbol{z}_j)$. Moreover, the likelihood can be optimized using a stochastic procedure with a random subset of each $\boldsymbol{x}_j$.

Let us consider again the J latent variables per location, denoted $\mathbf{z}_n^{(i)}$. To avoid exhaustively updating every $\mathbf{z}_n^{(i)}$, we follow the framework proposed by [11]. According to the "reparameterization trick", equations (15) can be rewritten as

$$\mathbf{z}_n^{(i)} = \boldsymbol{\mu}_n^{(i)} + \boldsymbol{\epsilon}_n^{(i)},\ \boldsymbol{\epsilon}_n^{(i)} \sim N(0, \mathbf{I})\ . \tag{20}$$

Therefore, during the stochastic inference, the latent variables $\tilde{\mathbf{z}}$ can be simulated based on the other parameters and the random noise. This will save computation and memory because the $\tilde{\mathbf{z}}$ only need to be simulated at the subset of positions samples in the ongoing mini-batch of optimization. The noise is first sampled from a standard normal distribution and a pseudo variable $\tilde{\mathbf{z}}$ is then generated as

$$\tilde{\mathbf{z}}_n^{(i)} = \hat{\boldsymbol{\mu}}_n^{(i)} + \boldsymbol{\epsilon}_n^{(i)}, \boldsymbol{\epsilon}_n^{(i)} \sim N(0, \mathbf{I}) \tag{21}$$

where $\hat{\boldsymbol{\mu}}_n^{(i)} = \hat{\boldsymbol{\rho}}_i^\top \frac{1}{|\boldsymbol{c}_n|} \sum_{i' \in \boldsymbol{c}_n} \hat{\boldsymbol{\alpha}}_{i',j}$, and $\hat{\boldsymbol{\rho}}$ and $\hat{\boldsymbol{\alpha}}$ are point estimates of the embedding vectors which we are optimizing. The log-likelihood function becomes $\tilde{\mathcal{L}}(\boldsymbol{\rho}, \boldsymbol{\alpha}; \mathbf{x}) = \sum_j \log p(r(\boldsymbol{x}_j)|\tilde{\mathbf{z}}_j)$ and the objective function is computed as

$$\mathcal{F} = \tilde{\mathcal{L}}(\boldsymbol{\rho}, \boldsymbol{\alpha}; \mathbf{x}) + \sum_i \log p(\boldsymbol{\rho_i}) + \sum_{i'} \sum_j \log p(\boldsymbol{\alpha_{i'}}, \boldsymbol{j})\ . \tag{22}$$

In each iteration, the gradients of the log-likelihood with respect to $\hat{\boldsymbol{\rho}}$ and $\hat{\boldsymbol{\alpha}}$ can be simply obtained with $\tilde{\mathbf{z}}$ by chain rule via $\nabla_{\boldsymbol{\rho}} \tilde{\mathcal{L}} = \sum_j \frac{\partial \log p(\boldsymbol{x}_j|\tilde{\boldsymbol{z}}_j)}{\partial \tilde{\boldsymbol{z}}_j} \frac{\partial \tilde{\boldsymbol{z}}_j}{\partial \boldsymbol{\rho}}$, and $\nabla_{\boldsymbol{\alpha}_j} \tilde{\mathcal{L}} = \frac{\partial \log p(\boldsymbol{x}_j|\tilde{\boldsymbol{z}}_j)}{\partial \tilde{\boldsymbol{z}}_j} \frac{\partial \tilde{\boldsymbol{z}}_j}{\partial \boldsymbol{\alpha}_j}$. The $p(\boldsymbol{\rho_i})$ and $p(\boldsymbol{\alpha_{i'}}, \boldsymbol{j})$ are set to $\mathbf{N}(0, \mathbf{I})$ and $\mathbf{N}(0, \lambda_\alpha^{-1}\mathbf{I})$ according to equations (11) and (14). The gradients of the log-priors are $\sum_i \frac{\partial \log p(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}_i}$ and $\sum_{i'} \sum_j \frac{\partial \log p(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}_{i',j}}$ respectively. The gradient to update $\hat{\boldsymbol{\rho}}$ or $\hat{\boldsymbol{\alpha}}$ with respect to $\mathcal{F}$ is then the sum of the corresponding log-likelihood and log-prior gradients.

The complete stochastic inference procedure is given in Algorithm 1. We optimize the embedding vectors iteratively over epochs. In each epoch data is partitioned randomly into mini-batches, and negative samples are generated for each batch in addition to its positive samples: a negative sample has the same location n and context as a positive sample but a randomly chosen different item i, and its observed variable values are all set to zero since the item did not occur at that location. The items for the negative samples are chosen from a distribution proportional to a power of the overall item distribution, as is done in word embedding [14]. In experiments we use $M = 1000$ mini-batches and 5 negative samples for each positive sample. Due to the stochastic partitions, for each epoch the latent variables only need to be simulated at the positions in the mini-batch. The optimization then updates the embedding vectors in each epoch by gradient steps with step sizes chosen by the Adam optimizer.

5 Empirical Case Studies

In this section we describe 5 different scenarios of using GCE to model the observed data. The precision parameter λ_α is set to 0 corresponding to a very wide prior for $\boldsymbol{\alpha}$. We have also tried another setting with a constrained prior, they yield similar results (see supplementary materials).

5.1 Product rating data

Data. The Anime rating data[1] is a set of user ratings on anime movies and series collected from myanimelist.net. It contains 17562 different anime rated by 325770 different users. Unlike typical product rating data sets, the data set provides how many episodes (integer) the user had watched when rating the anime (discrete), thus there are multivariate heterogeneous observed variables.

Modeling. We evaluate GCE by comparing with two other models on their capability to predict held-out ratings. Compared models are exponential family embeddings (Poisson distribution, p-emb) and Poisson matrix factorization model ([2], Pois-MF). The p-emb is selected as a comparison method

[1]From Kaggle, https://www.kaggle.com/datasets/CooperUnion/anime-recommendations-database

Algorithm 1: Inference Algorithm

input : Observations $\{\boldsymbol{x_1} \dots \boldsymbol{x_N}\}$, Context $\{\boldsymbol{c_1} \dots \boldsymbol{c_N}\}$, initial learning rate $\boldsymbol{\xi}$
output : Point estimates of embedding vectors $\hat{\boldsymbol{\rho}}$
and context vectors $\hat{\boldsymbol{\alpha}}$

foreach *epoch* **do**
  Divide input data into M random partitions.
  Generate negative samples.
  for $m \leftarrow 1$ **to** M **do**
    Use the m-th batch of the data
    Simulate $\tilde{\mathbf{z}}_n$ with (15) for every n in the mini-batch
    Compute gradients $\nabla_{\boldsymbol{\rho}}\mathcal{F} = \sum_j \frac{\partial \log p(\boldsymbol{x}_j|\tilde{\boldsymbol{z}}_j)}{\partial \tilde{\boldsymbol{z}}_j} \frac{\partial \tilde{\boldsymbol{z}}_j}{\partial \boldsymbol{\rho}} + \sum_i \frac{\partial \log p(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}_i}$,
    $\nabla_{\boldsymbol{\alpha}_j}\mathcal{F} = \frac{\partial \log p(\boldsymbol{x}_j|\tilde{\boldsymbol{z}}_j)}{\partial \tilde{\boldsymbol{z}}_j} \frac{\partial \tilde{\boldsymbol{z}}_j}{\partial \boldsymbol{\alpha}_j} + \sum_{i'} \sum_j \frac{\partial \log p(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}_{i',j}}$
    Update $\boldsymbol{\rho}$ and $\boldsymbol{\alpha}$ with $\boldsymbol{\rho} = \boldsymbol{\rho} - \boldsymbol{\xi} * \nabla_{\boldsymbol{\rho}}\mathcal{F}$, and $\boldsymbol{\alpha} = \boldsymbol{\alpha} - \boldsymbol{\xi} * \nabla_{\boldsymbol{\alpha}}\mathcal{F}$
    $\boldsymbol{\xi}$ is set with Adam[10]
  end
end

Table 1: Left: Held-out MAE for Anime rating. Right: Held-out MAE for Match Records.

				Kills		Deaths	
Model	K = 50	K = 100	Model	K = 50	K = 100	K = 50	K = 100
Pois-MF	7.4866	7.4872	n-emb	30.0125	31.3224	29.4128	30.6495
p-emb	3.0857	3.0691	p-emb	32.5609	32.5409	32.5620	32.5433
GCE	**1.2170**	**1.2207**	GCE	**14.0867**	**14.0766**	**13.5245**	**13.5106**

since our model can been seen as an extension of exponential family embeddings; the Pois-MF is a model for the user-rating scenario and it was also a compared method to p-emb in [19]. For all methods, when training the model, we hold out 10% of the data as the testing data set, and the trained models are used to predict the ratings in the test data. For each anime rated by a specific user, other anime rated by the same user are its context. We train GCE with two variables, anime rating and number watched episodes, whereas p-emb and Pois-MF which can only model one variable are trained to model the anime ratings.

To compute the predictive ratings we use (16) with $z_{n,j}^{(i)}$ set to the means $\mu_{n,j}^{(i)}$ computed from the optimized embedding vectors and with F_j^{-1} computed as the inverse of the empirical CDF in the training data. The held-out mean absolute error (MAE) is used as the performance metric. The results are shown in the Table 1 (Left): our model strongly outperforms p-emb and Pois-MF.

5.2 Player modeling in online games

Data. The HLTV match record data is collected from HLTV.org and records professional match histories of a multiplayer first-person shooter game Counter-Strike: Global Offensive [22]. We used a web crawler to gather the histories of 34900 matches of 4751 professional players from the website. For each match, we collect the match ID, player ID, and records of each player in the match including number of kills and deaths, again yielding multivariate heterogeneous observations.

Modeling. We train GCE to learn representation vectors for each player. The context for the player in each match is the set of other players in the same match. Observed data are the numbers of kills and deaths. We again compare our model to two exponential family embeddings (normal and Poisson) because they are the methodologically closest approaches. We train GCA incorporating the two variables at the same time whereas the exponential family embeddings train the model for each variable separately. The predictions for the variables are done as in Section 5.1. We measure MAE for both variables separately. The results are shown in Table 1 (Right): we strongly outperform the comparison methods.

Table 2: Results for Darknet Traffic Classification

Model	Precision	Recall	F1	Accuracy
DeepImage [6]	0.86	0.86	0.86	0.86
Random forest (original features)	0.8374	0.7963	0.8117	0.8917
GCE (K = 20) + Random forest	**0.8955**	0.8789	0.8846	0.9347
GCE (K = 30) + Random forest	0.8945	**0.8803**	**0.8851**	**0.9355**
GCE (K = 40) + Random forest	0.8952	0.8786	0.8844	0.9346

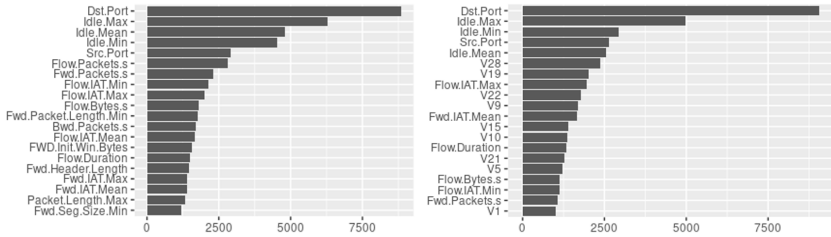

Figure 1: Importance of top-20 variables in the random forest classifiers, x-axis is the mean decrease of Gini impurity, the higher the more important the variable. Left: Variable importances in the random forest for original input variables. Right: The random forest model trained with additional 30 learned features. Learned features V28, V19, and other 7 features are in the 20 most important variables.

5.3 Internet traffic classification

Data. The CIC Dark-net traffic data set [6] contains 141532 records of darknet traffic. Each record is categorized into a traffic category (Audio-Stream, Browsing, Chat, Email, P2P, Transfer, Video-Stream, and VOIP, 8 categories in total) and contains the source IP, destination IP, and communication observations such as forward and backward bytes, flows, duration, subflow and so on.

Model. We use GCE to learn the latent representation for each source IP. In each traffic record, the destination IP is the context of the source IP. The observed data used to train the GCE model are IP co-appearance (Boolean), TCP Flag counts (SYN, RST, PSH, and ACK; integer), and three subflow related measurements (continuous value). We first train the embedding model with different vector dimensions. After training GCE, we incorporate the learned representation vector of each source IP as additional features to the original input variables and train a random forest classifier to predict the traffic category. We compare our model to DeepImage, which is a convolutional neural network based, end-to-end solution proposed in [6]. The results in Table 2 show that the learned additional features not only improve performance of the random forest classifier, but also outperform DeepImage, the state-of-the-art deep learning based classifier. Figure 1 further demonstrates that the learned features play important roles in the classification task.

5.4 Graph embedding with node meta-data

Data. Spanish Twitch gamers is a subgraph of the Twitch gamers graph data [17]; each node is a Twitch gamer; an edge denotes mutual friendship. The data has 5538 nodes and 85893 edges.

Model. We train the embedding vectors incorporating the node-level observations including number of views and life duration. Following the customary graph embedding procedure, we first generate random walks on the graph to simulate a node sequence as input data: 80 walks per node with length 10 steps. Conventionally, limited by the capability of embedding models, most graph embedding models only take appearance of nodes into account. With GCE, we incorporate not only the appearance of nodes but also the views and lifetime of the nodes into the model.

To evaluate our model, we take on link prediction, a classical task for graph embedding models. We hold out 50% of edges randomly into a test set while keeping the remaining training graph connected. In both training and test sets, randomly sampled negative edges are added in equal amount to the

Table 3: Results for Link Prediction: area under the curve (AUC) of the link classification

Model	Deepwalk	Node2vec	EFGE-bern	EFGE-pois	EFGE-norm	GCE
K = 50	0.7151	0.7143	0.5795	0.5934	0.6063	**0.7853**
K = 100	0.7063	0.6612	0.5887	0.6004	0.6291	**0.7832**

Table 4: Example GCE model output with the subreddit hyperlink network data. We show the top 4 closest subreddits in terms of the embedding vectors $\boldsymbol{\rho}$ and three different context vectors. The $\boldsymbol{\alpha}_1$ corresponds to the fraction of the characters, $\boldsymbol{\alpha}_2$ corresponds to the fraction of the digits, and $\boldsymbol{\alpha}_7$ corresponds to the semantics.

	Top 4 closest subreddits			
Embedding	r/environment			
$\boldsymbol{\rho}$	r/cornbreadliberals	r/basicincome	r/energy	r/northcarolina
$\boldsymbol{\alpha}_1$	r/climate	r/green	r/oil	r/water
$\boldsymbol{\alpha}_2$	r/conservation	r/climate	r/likeus	r/tdcs
$\boldsymbol{\alpha}_7$	r/invasivespecies	r/lockcarbon	r/metageopolitics	r/earthdisaster
	r/cryptocurrency			
$\boldsymbol{\rho}$	r/litecoin	r/noblecoin	r/siacoin	r/bitcoinserious
$\boldsymbol{\alpha}_1$	r/altcoin	r/blackcoin	r/ripple	r/karmacoin
$\boldsymbol{\alpha}_2$	r/cannabis	r/xdp	r/flappycoin	r/litecoin
$\boldsymbol{\alpha}_7$	r/dogenews	r/ethtrader	r/ethdev	r/vos

positive edges. A logistic regression classifier is trained based on the reduced training graph and the training negative edges, using Hadamard product of embeddings of the edge endpoint nodes as input features; the classifier is used to classify the held-out test-set edges. We compare our model to state-of-the-art, random walk based solutions including Deepwalk [16], Node2vec [5] and Exponential Family Graph Embeddings (EFGE; [1]). The results in the Table 3 show our model outperforms the other competitive models.

5.5 Social media community interactions

Data. The Reddit Hyperlink Network [12] is a data set of 858488 hyperlinks between 55863 subreddits. For each hyperlink, the data set records the source and destination subreddit, and the description of the hypertext including, e.g., number of words, sentiments, fractions of 5 different character types (i.e., alphabetical, digits, uppercase characters, special characters, white space) and so on.

Model. We train the model based on the pairs of source and destination subreddits. The source subreddit in each hyperlink is the context for the destination subreddit. The 5 fractions of different character types, the number of words, and the sentiment are taken as the observed variables. We train a GCE model with $K = 100$.

We demonstrate the closest subreddits based on embedding vectors, and three context vectors. Each reflects a different aspect: take the r/environment for example, when it comes to fraction of alphabetical characters, closest subreddits are related to resources such as r/climate, r/green, and r/oil, and r/water. However, when it comes to sentiment, the closest subreddits for r/environment become r/invasivespecies, r/lockcarbon, r/metageopolitics, and r/earthdisaster.

6 Discussions and Conclusions

We introduced Gaussian copula embeddings (GCE), a representation learning model that can incorporate observed data of different data types. A stochastic variational inference algorithm based on semi-parametric estimation for efficient computation is introduced. The empirical case studies demonstrate that our model is effective in many domains outperforming competitive comparison methods, and can provide analytical insights. Moreover, our model can extend the representation learning task to more complex settings and thus bring more opportunities to the research community.

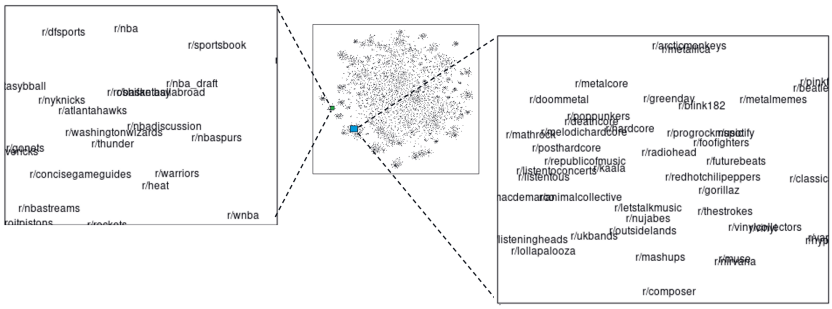

Figure 2: t-SNE visualization of embedding vectors $\boldsymbol{\rho}$. The green area contains the subreddits related to basketball and green area contains subreddits related to music. The learned representation from GCE are semantically meaningful.

In this paper we used a streightforward parametric construction of context and its combination with embedding vectors; however, the GCE framework can be flexibly adapted to other parameterizations, such as integrating it as a layer within deep learning architectures, and integrating context selection mechanisms such as [13]. Our method brings a new way of analyzing data through vectorial embedding which has the potential to bring greater understanding of several phenomena; as usual such tools must be used responsibly to avoid negative societal impact.

Acknowledgement

This work is supported by the Academy of Finland decisions 312395 and 327352.

References

[1] A. Celikkanat and F. D. Malliaros. Exponential family graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3357–3364, 2020.

[2] D. Cortes. Fast non-bayesian poisson factorization for implicit-feedback recommendations. *arXiv preprint arXiv:1811.01908*, 2018.

[3] R. Cui, P. Groot, M. Schauer, and T. Heskes. Learning the causal structure of copula models with latent variables. 2018.

[4] S. Gershman and N. Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014.

[5] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[6] A. Habibi Lashkari, G. Kaur, and A. Rahali. Didarknet: A contemporary approach to detect and characterize the darknet traffic using deep image learning. In *2020 the 10th International Conference on Communication and Network Security*, pages 1–13, 2020.

[7] P. D. Hoff. Extending the rank likelihood for semiparametric copula estimation. *The Annals of Applied Statistics*, 1(1):265–283, 2007.

[8] P. D. Hoff, X. Niu, and J. A. Wellner. Information bounds for gaussian copulas. *Bernoulli: official journal of the Bernoulli Society for Mathematical Statistics and Probability*, 20(2):604, 2014.

[9] A. Kalaitzis and R. Silva. Flexible sampling of discrete data correlations without the marginal distributions. *Advances in Neural Information Processing Systems*, 26, 2013.

[10] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

[11] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[12] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. Community interaction and conflict on the web. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 933–943. International World Wide Web Conferences Steering Committee, 2018.

[13] L. Liu, F. Ruiz, S. Athey, and D. Blei. Context selection for embedding models. *Advances in Neural Information Processing Systems*, 30, 2017.

[14] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[15] J. S. Murray, D. B. Dunson, L. Carin, and J. E. Lucas. Bayesian gaussian copula factor models for mixed data. *Journal of the American Statistical Association*, 108(502):656–665, 2013.

[16] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[17] B. Rozemberczki, R. Davies, R. Sarkar, and C. Sutton. Gemsec: Graph embedding with self clustering. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2019*, pages 65–72. ACM, 2019.

[18] M. Rudolph, F. Ruiz, S. Athey, and D. Blei. Structured embedding models for grouped data. *Advances in neural information processing systems*, 30, 2017.

[19] M. Rudolph, F. Ruiz, S. Mandt, and D. Blei. Exponential family embeddings. In *Advances in Neural Information Processing Systems*, pages 478–486, 2016.

[20] J. Segers, R. Van den Akker, and B. J. Werker. Semiparametric gaussian copula models: Geometry and efficient rank-based estimation. *The Annals of Statistics*, 42(5):1911–1940, 2014.

[21] M. Sklar. Fonctions de repartition an dimensions et leurs marges. *Publ. inst. statist. univ. Paris*, 8:229–231, 1959.

[22] H. P. E. Valve Corporation. Counter-strike: Global offensive, 2012.

[23] I. Waller and A. Anderson. Quantifying social organization and political polarization in online platforms. *Nature*, 600(7888):264–268, 2021.

[24] R. C. Weng and C.-J. Lin. A bayesian approximation method for online ranking. *Journal of Machine Learning Research*, 12(1), 2011.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] Provided in supplementary material.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] In supplementary.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In supplementary.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]