

CUSTOMER REVIEWS ANALYSIS WITH DEEP NEURAL NETWORKS FOR  
E-COMMERCE RECOMMENDER SYSTEMS

by

Babak Maleki Shoja

July, 2019

Director of Thesis: Dr. Nasseh Tabrizi

Major Department: Computer Science

The first part of this thesis systematically reviews the trend of researches conducted from 2011 to 2018 in terms of challenges and problems regarding developing a recommendation system, areas of application, proposed methodologies, evaluations criteria used to assess the performance and limitations and drawbacks that require investigation and improvements. The study provides an overview for those who are interested in this field to understand the current and the future research opportunities.

The second part of this thesis proposes a new methodology to consider customer reviews in recommender systems. An essential prerequisite of an effective recommender system is providing helpful information regarding users and items to generate high-quality recommendations. Customer reviews are a rich source of information that can offer insights into the recommender systems. However, dealing with the customer feedback in text format, as unstructured data, is challenging.

Our research includes extraction of the features from customer reviews and use them for similarity evaluation of the users to generate the recommendations. To do so, we have developed a glossary of features for each product category using Latent Dirichlet Allocation. We then employed a deep neural network to extract deep features from the users-attributes matrix to deal with sparsity, ambiguity, and redundancy. Furthermore, we then applied matrix factorization as the collaborative

filtering method to provide recommendations. The experimental results using Amazon dataset demonstrate that our methodology improves the performance of the recommender system by incorporating information from reviews when compared to the baselines.



CUSTOMER REVIEWS ANALYSIS WITH DEEP NEURAL NETWORKS FOR  
E-COMMERCE RECOMMENDER SYSTEMS

A Thesis

Presented to The Faculty of the Department of Computer Science  
East Carolina University

In Partial Fulfillment of the Requirements for the Degree  
Master of Science in Computer Science

by

Babak Maleki Shoja

July, 2019

Copyright Babak Maleki Shoja, 2019

CUSTOMER REVIEWS ANALYSIS WITH DEEP NEURAL NETWORKS FOR  
E-COMMERCE RECOMMENDER SYSTEMS

by

Babak Maleki Shoja

APPROVED BY:

DIRECTOR OF THESIS:

---

Dr. Nasseh Tabrizi

COMMITTEE MEMBER:

---

Dr. Venkat Naidu Gudivada

COMMITTEE MEMBER:

---

Dr. Haiyong Liu

CHAIR OF THE DEPARTMENT

OF COMPUTER SCIENCE:

---

Venkat Gudivada, PhD

DEAN OF THE

GRADUATE SCHOOL:

---

Paul J. Gemperline, PhD

## Table of Contents

LIST OF TABLES . . . . .	vi
LIST OF FIGURES . . . . .	vii
1 INTRODUCTION . . . . .	1
1.1 Research Contribution . . . . .	5
1.2 Thesis structure . . . . .	5
2 SYSTEMATIC LITERATURE SURVEY . . . . .	6
2.1 Introduction . . . . .	6
2.2 Publication Trend . . . . .	7
2.3 Systematic Review . . . . .	8
2.4 Tags: Opportunities and Challenges . . . . .	9
Social Tagging . . . . .	11
Challenges of Tags . . . . .	12
2.5 Methodologies . . . . .	13
2.5.1 Collaborative Filtering . . . . .	14
2.5.2 Content-Based Filtering . . . . .	15
2.5.3 Hybrid and Other Techniques . . . . .	15
2.5.4 AI for Tag-Aware Recommender Systems . . . . .	17
2.6 Applications . . . . .	17

2.6.1	Music . . . . .	17
2.6.2	Movies . . . . .	18
2.6.3	E-learning . . . . .	18
2.6.4	Documents . . . . .	19
2.6.5	Social Medias . . . . .	19
2.6.6	Food . . . . .	19
2.6.7	Tourism . . . . .	20
2.6.8	Distribution of the Applications . . . . .	20
2.7	Evaluation Criteria . . . . .	21
2.8	Future Research Directions . . . . .	23
2.9	Chapter Conclusions . . . . .	25
3	RELATED LITERATURE ON CUSTOMER REVIEWS . . . . .	27
4	METHODOLOGY . . . . .	30
4.1	Latent Dirichlet Allocation . . . . .	31
4.2	Deep Neural Networks . . . . .	33
4.3	Collaborative Filtering . . . . .	36
5	EXPERIMENTAL RESULTS . . . . .	38
5.0.1	Dataset . . . . .	38
5.0.2	Baseline Methods . . . . .	42
5.0.3	Evaluation . . . . .	43
6	FUTURE WORK AND CONCLUSION . . . . .	49
	BIBLIOGRAPHY . . . . .	51



## LIST OF TABLES

2.1	Summarized information from reviewed papers . . . . .	10
5.1	General statistics of the Amazon Review dataset . . . . .	39
5.2	MSE of the proposed method vs baselines and the percentage of the improvement . . . . .	44

## LIST OF FIGURES

2.1	Number of papers published from 2011 to 2018 . . . . .	7
2.2	Number of studies dealing with a specific problem in recommender system development . . . . .	14
2.3	Areas of application for recommender systems . . . . .	21
2.4	Evaluation Criteria . . . . .	24
4.1	General framework of the proposed approach of using customer reviews in recommendation systems . . . . .	31
4.2	Demonstration of a simple autoencoder . . . . .	35
5.1	Sparsity of the Amazon Reviews Dataset . . . . .	40
5.2	Analysis of MSE based on the number of hidden layers . . . . .	41
5.3	Analysis of MSE based on the number of neurons in the first hidden layers . . . . .	42
5.4	Analysis of MSE based on the number of neurons in the second hidden layers . . . . .	43
5.5	Comparing the MSE from the proposed method and the baselines . . . . .	45
5.6	Proposed model improvement compared to MF . . . . .	46
5.7	Proposed model improvement compared to HFT . . . . .	47
5.8	Proposed model improvement compared to RMR . . . . .	48

## Chapter 1

### Introduction

We are facing with many decisions to make during our daily life such as what movie to watch, what article or news to read, which clothes to buy, which restaurant or bar to visit, and many other instances [11]. The exponential growth of data and information on the Internet confronts us with information overload. This growth of available data results in a tremendous amount of information that makes it hard for people to make choices between an enormous number of movies, books, web pages, and other products, and poses a challenge to user's ability to efficiently access required data [12, 79]. Evaluating even a small portion of such data seems to be impractical, increasing the need for automatic recommender systems with the capability of suggesting relevant items as well as new items to the customers and clients [87, 28]. Besides, personalization and customization of recommendations for users and providing suggestions in the ever-increasing information is a crucial and challenging problem for online service providers such as e-learning.

Recommender systems are a branch of information filtering systems that tries to predict users' preferences for an item and provide personalized suggestions based on this analysis for a particular user. In other words, recommender systems help users to find products or services they need based on analysis of user preferences using client profiles and their similarities or finding products or services that are similar to those

clients who have already expressed interest [3]. Recently, there is an increasing trend in employing this approach to various areas, including music, books, social tags, and wide variety of products. Several e-commerce companies such as Amazon employ recommender systems and relevant tools to enhance the recommendations to their customers with the primary purpose of increasing overall profits [28, 61, 33, 89, 34].

Generating proper recommendations to the users requires information about the users' characteristics, preferences, and needs [34]. Recommender systems mainly consider the overall rating customer gives to items where latent factor models such as Matrix Factorization (MF) are widely used to predict ratings. However, there are drawbacks for using MF models such as cold-start problem, considering only the customer overall satisfaction, and sparsity. As the literature on the MF methods show, numerous researches are devoted to tackling the weaknesses of MF methods by incorporating side information such as tags [65, 82], visual features [24], and social relations [46, 77].

Social tagging has become popular, along with e-commerce. Social tagging systems are systems that enable users to annotate tags to an item they are visiting or purchasing. This tagging process introduces a keyword describing the users' opinion on a characteristic of the item or the item as a whole and leads to a folksonomy and forms categories that facilitate developing a recommender system. These types of recommender systems are called tag-aware recommender systems and they face a similar situation as any other recommender systems discussed above. Two recommendation problems arise toward social tagging systems. First is tag recommendation for users to make tagging an item easier. Second is providing actual item recommendations for the customer using previously annotated tags [33]. There are several review papers on recommender systems published in the last few years including [45, 68, 32]. Regarding tag-aware recommender systems, the authors of [87] presented a survey

published in 2011. This thesis reviews researches in this area from 2011 to provide useful insights on the recent studies on tag-aware recommender systems and depicting the future research directions in this field of study.

Additionally, customer reviews are one of the critical resources in developing recommender systems. A written part of the review of a rating includes essential information on what the customer thinks about the product. Consequently, researchers suggest many models that exploit reviews with ratings for improving the recommendations. Some of these models are discussed in [42, 51, 72, 76]. Sentiment analysis is one of the conventional approaches toward the analysis of customer reviews. It is mainly to predict whether the attitude of a piece of text is positive or negative, supported or opposed [83]. Semantic analysis is employed to analyze customer reviews [83, 1, 70] for different objectives such as to measure e-commerce service quality [62]. Some recent studies try to use customer reviews in developing recommender systems. The approaches they utilized include semantic analysis and aspect-based latent factor models [37, 59, 10, 4].

As discussed before, in tag-aware recommender, we basically use a ternary relationship between users-tags-items. In some systems and online resources, clients annotate tags to items while they are rating them, either manually or by the help of the system which recommends tags. For the e-commerce industry, for example Amazon.com, this option is not available. As discussed above, the customer reviews are rich with information about customer opinions on the item they are talking about. The general idea in this research is that we can build a tag-aware type recommender system in which we extract product attributes from the customers' textual feedback to play the role of the tags.

Sparsity is another major problem in recommender systems, which significantly reduces the performance of recommendation systems, as is evident in the literature.

The problem of sparsity in the rating matrix is sometimes called gray sheep problem, which is peculiar to similarity-based collaborative recommendation systems. The problem arises from the fact that user-item interaction occurs for a tiny percentage of all possible interactions. This minimal interaction rate is because the user only chooses a tiny portion of all the items to interact. In other words, sparsity occurs when there are a vast number of items available, and even the most active user cannot provide feedback for even a small portion of the items [33], which in turn makes some users not similar enough to others to discover their preferences. Hence, we cannot retrieve the proper recommendation list.

As we will discuss later in Section 2, several studies try to deal with the sparsity problem for different recommender systems and contexts, however, none have used a deep neural network for handling the sparsity of the users-attributes matrix extracted from the customer reviews. Deep neural networks can extract deep features from the matrix, which not only solves the sparsity issue but also can alleviate the ambiguity and redundancy in the extracted attributes. In this thesis, we report a systematic review on tag-aware recommender systems as the basic idea of our proposed model is based on this type of recommender systems. We then perform a customer review mining and extract a set of product characteristics that users mentioned in their reviews and use the Latent Dirichlet Allocation (LDA) model along with the association rule mining method to finalize the set of characteristics. We then use the set of attributes to construct the users-attributes matrix. This matrix, however, is very sparse as each user mentions only a few attributes in the review. To deal with this problem, we use a deep neural network that plays an autoencoder role that helps to learn more abstract and latent attributes. Having users-attributes and users-items matrix, we use an MF model to predict ratings and provide recommendations.

## 1.1 Research Contribution

In this research work, we first report a systematic literature survey on the tag-aware recommender systems as the basis of the proposed approach is this type of recommender systems. In this literature survey, we systematically review the trend of researches conducted from 2011 to 2018 in terms of challenges and problems regarding developing a recommendation system, areas of application, proposed methodologies, evaluation criteria used to assess the performance and limitations and drawbacks that require investigation and improvements.

The second and primary contribution of this thesis is developing a new recommender system that extracts the product attributes from the customer reviews and deal with the sparsity, a significant problem in recommender systems, using a deep neural network that extracts deep features from the users-attributes matrix. The proposed system uses MF for recommendation generation. As the empirical experiment results show, our model outperforms baseline models for most cases.

## 1.2 Thesis structure

The structure of this thesis is as follows. Chapter 2 presents the systematic literature survey on tag-aware recommender systems. Chapter 3 provides the related work on the use of customer reviews in the recommendation systems. Chapter 4 describes the proposed methodology in detail. Chapter 5 provides the performance analysis of our model using the most extensive public dataset for product reviews, Amazon Reviews dataset. Last but not least, we conclude the current research and discuss future research directions in Chapter 6.

## Chapter 2

### Systematic Literature Survey

#### 2.1 Introduction

Social tagging has become popular, along with e-commerce. Social tagging systems are systems that enable users to annotate tags to an item they are visiting or purchasing. This tagging process is a keyword describing the user's opinion on a characteristic of the item or the item as a whole and leads to a folksonomy and forms categories that facilitate developing a recommender system. These types of recommender systems are called tag-aware recommender systems. These systems are facing a similar situation as any other recommender systems discussed above. Two recommendation problems arise toward social tagging systems. First is tag recommendation for users to make tagging an item easier. Second is providing actual item recommendations for the customer using previously annotated tags [33].

Zhang et al. [87] presented a survey on tag-aware recommender systems, which is published in 2011. This thesis reviews the researches in this area from 2011 to provide useful insights into the recent studies on tag-aware recommender systems and depicting the future research directions in this field of study. Therefore, in this thesis, we systematically retrieved and reviewed researches from 2011 to 2018 that studied tag-aware recommender systems.



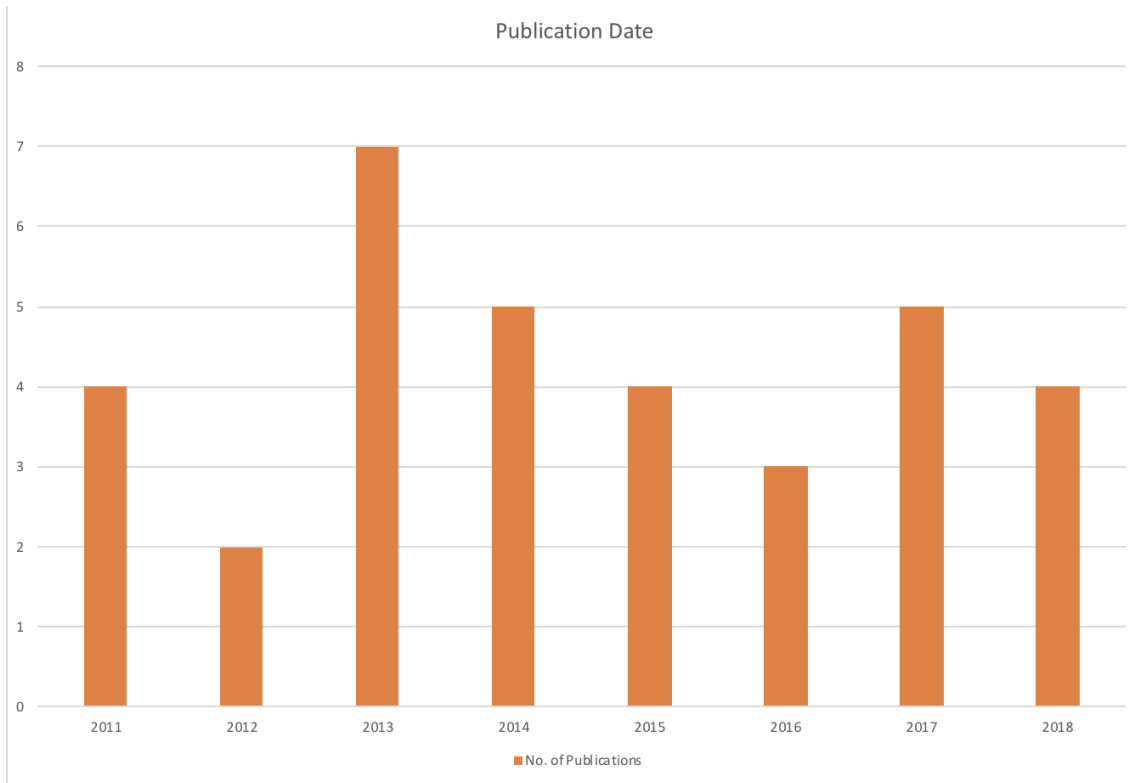


Figure 2.1: Number of papers published from 2011 to 2018

## 2.2 Publication Trend

The challenges, drawbacks, and new opportunities that have arisen due to the availability of more data and information have called for studies on developing and implementing tag-aware recommender systems in recent years. Figure 2.1 demonstrates the number of publications that use tags to develop their recommender systems or deal with tag recommendation published between 2011 and 2018. It can be seen that researchers are investigating this problem on a relatively constant level in terms of the number of the publication. We can conclude that tag-aware recommender systems are being investigated and as we see throughout the review, there are challenges and drawbacks to current literature, and we expect to see this relatively constant trend stays on for the new years (if it does not start to grow).

## 2.3 Systematic Review

As mentioned above, there is a need for a survey on tag-aware recommender systems to review the studies conducted from 2011 so far. The goals of this survey are:

- Identifying problems and challenges regarding the tag-aware recommender systems;
- Identifying algorithms and methodologies employed to tackle these problems and challenges;
- Identifying Areas of application for tag-aware recommender systems
- Identifying evaluation criteria used to evaluate developed recommender systems.

This survey tries to find the answer to the following questions.

- What are the challenges that are mostly being addressed, and which of them has acquired less attention by researchers?
- What areas of application are used more to develop recommender systems and which ones are promising for further studies?
- What are the main criteria for evaluating recommender systems and if there is a need for new criteria?
- What are the future research directions for tag-aware recommender systems?

Through the next sections of this systematic review, we try to answer these questions to reach the goals mentioned earlier.

As the protocol for this survey, first, we searched for publications via scientific search engines and gathered publications as much as possible. The search query used in search engines is as follows.

((`"recommender" AND "system"`) OR (`"recommendation" AND "system"`)) AND `"tag"`)

This search query is used for searching in several databases including ACM Digital Library, Applied Science and Technology Full Text, Computing (ProQuest), Gartner, IEEE Xplore, Lynda.com, ProQuest Science, and SpringerLink.

As the second step of the protocol, we applied exclusion criteria to analyze retrieved publications to eliminate irrelevant publications according to [58].

As the third and last step of the question, all remaining publications (after applying exclusion criteria) are studied in depth and the information and data are extracted to answer the main questions of this systematic review. We constructed a detailed table of the information extracted from each selected study. A summarized form of this table is shown in Table 2.1. In the following sections, the results and conclusions are discussed in detail.

## **2.4 Tags: Opportunities and Challenges**

Information about the content and creation of a resource can be conveyed using tags [53], which we can consider as keywords assigned to photos, movies, music tracks, and other products [2]. The purpose of tagging for users is future information retrieval and sharing [49]. Golder et al. [21] identified the characteristics of a resource and what it is about by tags. These tags are often used to represent user's preferences regarding an item that tags belong to; therefore, a user profile can be developed [75] and a similar profile can be constructed. Typically, there are two types of tags: 1) standardized meta-data which provides standard information, and 2) free-text or user-defined tags written by users which are extremely useful for a recommender system [6, 38]. User-defined tags not only provide ratings on items but also demonstrate user preferences [89]. These free tags are used in various areas of application such as music recommender systems [28], tour recommendation [69], and e-commerce [61].

Table 2.1: Summarized information from reviewed papers

Author	Year	Application	Methodology	Problem Dealt	Using	Limitations for Future Research	Evaluation Criteria
Baralis et al.	2018	Images	Generalized association rules	Sparsity	User Generated tags	1) Using tags from social networks and online communities 2) integration of incremental rule mining approach to cope with evolution	(MRR), (S@k), Precision
Cai et al.	2018	Tourism	Semantic Trajectory pattern mining	Considering spatial semantics or temporal information	Semantic trajectory pattern mining	1) incorporate more semantic datasets 2) Comparison with real world agency recommendations 3) using dislikes too 4) incorporating photo images.	Similarity between user query and recommended itinerary, computation time
Sahu	2018	e-commerce	Cross-Domain Recommender System (transferring a knowledge from other domains)	Data Sparsity - Cold Start	User Generated tags	Correlation among domains using semantic analysis and integrating it with this model	MAE
Zhang et al.	2018	Images	FOLK POPULARITY RANK	-	User Generated tags	-	Image Popularity growth
Ghabayen et al.	2017	Movies	Semantic similarity between users (neighborhood)	Sparsity, Cold Start	User Generated tags	1) Investigating user generated data type 2) larger data sets	F1-Measure
Mahboob et al.	2017	Web and movies	Heat diffusion algorithm	-	User Generated tags	Other meta-heuristics, context-based methods, and using tag similarity	Precision and Recall
Massimo et al.	2017	Food	Matrix Factorization rating prediction model	sparsity	User Generated tags	1) Larger datasets 2) temporal evaluation	MAE and MRR
Sun	2017	Tourism	Latent Dirichlet Allocation	Recommending Area by Area instead of Landmark by Landmark	User Generated tags	Biased Conclusion	-
Zhang et al.	2017	Images	Frequency-Weights from regression and Folk Popularity Rank	-	User Generated tags	-	Image Popularity growth
Bang et al.	2016	Movies	collective matrix factorization	Sparsity	User Generated tags	Word-embedding recommender systems	RMSE and MAE
Ignatov et al.	2016	Music (Radio Station)	combines a collaborative user-based approach with personalized information from tags of listened tracks	cold start, boosting of rankings, preference dynamics, and absence of explicit feedback	Tags of listened tracks	1) using non-linear schema for scoring functions 2) using reinforcement learning in addition to provided solution	F1-Measure
Zuo et al.	2016	Music and Book	Deep neural network	Data Sparsity - Cold Start - Redundancy	User Generated tags	1) Other neural networks 2) Dealing with other information in recommender systems 3) using on other application areas such as movies, mobile, books	Precision and recall and rank score
Font et al.	2015	music (sound sharing)	tag-tag similarity metrics	-	User Generated tags	analysis of tags at semantic level	Computation time and Precision
Ge et al.	2015	Food	Matrix Factorization	-	User Generated tags and rating	1) Considering Nutritional Factors for health-aware recommendation 2) Incorporating negative tags 3) Recommendation for Group of users	RMSE and MAE
Movahedian and Khayyambashi	2015	web	Frequent tag pattern and External knowledge-base YAGO2	Noisy and Unreliable data	User Generated tags	1) Multi-domain scenario 2) Considering Social influence between users	RMSE and MAE and Precision and Ranking Accuracy
Trattner et al.	2015	Web and movies	FOLK POPULARITY RANK	-	User Generated tags	1) Involving content-based recommenders 2) Matrix factorization	Recall, Precision F1-Measure MRR, MAP
Gueye et al.	2014	web movies music documents	parameter free algorithm using relevance measures	-	User Generated tags	-	F1-Measure
Hong et al.	2014	Images	Clustering and visual similarity	Noise ambiguity	User Generated tags	-	percentage of the sum of the relatedness between user and image out of the overall number of tagged images by user
Horsburgh et al.	2014	music	Hybrid (Pseudo-Tag & limited by dynamic weighting)	Tag sparsity - Cold Start	Pseudo Tag	-	Precision and Recall
Huang et al.	2014	web	Collaborative and content-based filtering	cold start, sparsity, ambiguity	User Generated tags	1) Considering other parameters including the number of similar neighbors, the weight of recency, the number of recommendations, and the length of base time 2) Applying to other application areas 3) Using larger dataset 4) using rich tag information	F1-Measure

## Social Tagging

Social tagging is the ability of the user to add tags to items. These social tags are used in recommender systems to improve the recommendations to users, which sometimes called tagommenders. The combination of social tagging and recommender systems results in a system that generates automated recommendation while retaining the flexibility of tagging information [63]. Social Annotation Services (SAS), as one of the most successful web2, are significantly developed recently in which users organize, share and retrieve online resources such as resources in Delicious, articles in CiteU-Like and images and pictures in Flickr. These user-item online interactions resulted in a large amount of annotation data generated by users, and this diffusion of collaborative tagging has attracted investigators' attention in developing and proposing tag recommendation systems [3, 8, 18, 78, 47].

Tag recommendation is an important part of collaborative tagging systems and a part of tag-aware recommender systems as a whole. Researches in this area try to effectively retrieve the most relevant tags to suggest beforehand [22, 31, 22]. The number of recommendations can be fixed or dynamically change and be optimized. While tags bring many advantages as a flexible and efficient information management approach, tagging is a cumbersome task because they have to be annotated manually by the user. Tag recommendation systems are the solution to facilitate this process by suggesting tags to users [43]. Tag recommender systems should improve the annotation quality as well as decreasing the cost of the annotation process for users. Also, they result in the consolidation of the vocabulary of collaborative tagging systems [15].

In the recent decade, the growth of using mobile devices is exploded, and the use of mobile for social networking and multimedia (e.g., Flickr, YouTube) have

become increasingly popular. The tag-based recommender system is a solution for tag-based multimedia retrieval. In this regard, some investigators have focused on using recommender systems for tag-based multimedia retrieval on mobile devices [27].

### **Challenges of Tags**

Tags do not always benefit recommender systems [6]. Tags can be problematic when they are uninformative and more importantly when there are a few numbers of tags available (cold-start) [41]. It is also possible for a tag to form an uncontrolled vocabulary either by expressing the same meaning with different words (car and automobile) which results in redundancy, or polysemy which results in ambiguity (Amazon means both the e-commerce website and the river in Africa) [87].

Sparsity is another major problem in the tag-based recommender system, which significantly reduces the performance of recommendation systems, as is evident in the literature. The problem of sparsity in the rating matrix is sometimes called gray sheep problem, which is peculiar to similarity-based collaborative recommendation systems. The problem arises from the fact that user-item interaction occurs for a tiny percentage of all possible interactions. This low interaction is because the user only chooses a small portion of all the items to interact. In other words, sparsity occurs when there are a considerable number of items available, and even the most active user cannot even rate a portion of the items [33]. These few interactions make some users not similar enough to others to discover their preferences. Hence, proper recommendations cannot be obtained.

Moreover, the user provides a small number of tags when annotating an item. The user-item matrix is the fundamental of similarity analysis, and this sparsity leads to inaccuracy and ineffectiveness of the recommender systems. In this regard, many investigators have focused on dealing with this problem to provide a solution that

mitigates the effect of sparsity [86, 19].

As the next challenge, for a given item, recommender systems try to extract tags that represent the item properly ignoring how favorite the item is. However, after the extraction of the tags, some tags will not be selected while they are semantically related to the corresponding content description [13]. We call this semantic tag annotation problem. Dealing with this problem enhances the performance of the recommender systems. Also, in recommender systems, the evolution of taste and repertoire is a challenge to deal with since in almost every area of application, the taste of the user for purchasing or preferring items changes and evolves as time passes. This preference change usually is captured via individual-based approaches where tags for newly selected items should have more weights in the analysis [30, 88, 54].

A technical issue regarding recommender systems, especially those that deal with social tagging data, is that they are facing a large amount of computation which leads to the scalability problem. Consequently, providing an efficient and faster methodology that can compute a more significant amount of data results in a more efficient and effective recommender system.

Figure 2.2 demonstrates the number of papers that deal with a specific challenge and problem that are reviewed in this survey. It shows that the majority of the studies try to tackle cold-start and sparsity problems and propose a recommender system with better performance. While these problems still can be studied for better recommender systems, other problems require a more thorough investigation.

## 2.5 Methodologies

In this section, we go over the approaches and methodologies that are used in the literature of tag-aware recommender systems.

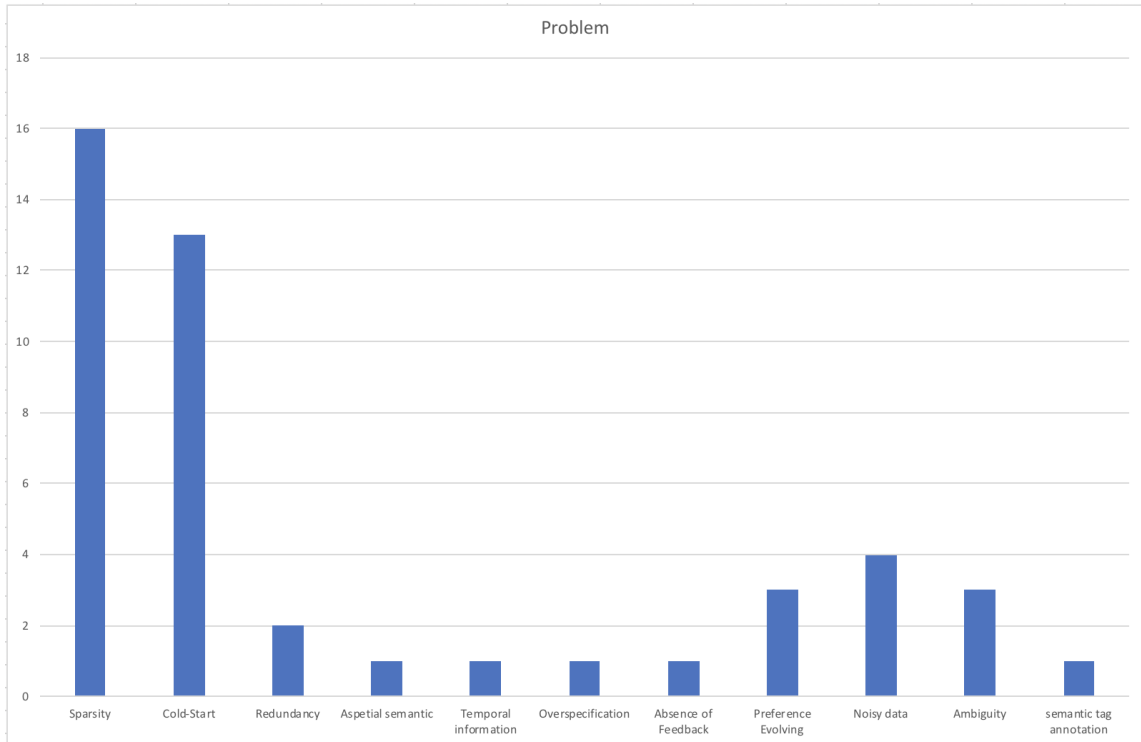


Figure 2.2: Number of studies dealing with a specific problem in recommender system development

### 2.5.1 Collaborative Filtering

Collaborative filtering (CF) is a popular and widely used method in recommender systems [25]. The core assumption in CF methods is that the users share similar interests in the future to the interest expressed in the past [20]. CF can be classified into memory-based methods and model-based methods [61]. Memory-based methods are then categorized as user-based and item-based methods. In user-based CF, the user’s rating on target is predicted based on the ratings of similar users while in item-based CF the item’s rating is based on the ratings of similar items provided by the user [66].

In this category, collaborative tagging systems are powerful tools in systematically capture user-generated tags where users create items and annotate tags to them to be



shared with other users. Delicious and MovieLens are instances in this regard. This way of sharing results in an unstructured knowledge schema known as folksonomy [67, 29, 56]. However, collaborative filtering has its limitations. One of the primary limitations of collaborative filtering is its need for a dense rating matrix which is unrealistic in practice because in most cases, users rate only a few items and a dense rating matrix cannot be obtained. This results in a sparse matrix that has a significant negative effect on CF methods. Also, it suffers from the cold-start problem as well [89].

### **2.5.2 Content-Based Filtering**

The next category is content-based recommendation systems where information regarding an item that is stored in tags is utilized, and the system tries to find an item that has similarity to the item that the user has already liked. This is performed by measuring the similarity between items consumed by the user and other available items. This approach is not always useful. For example, in the case of music recommendation, this leads to a predictable recommendation, which is not desirable [74].

### **2.5.3 Hybrid and Other Techniques**

Various techniques are suggested to deal with tags limitations. In tag-aware recommender systems, we often deal with redundancy in tags. Clustering based methods are used in the literature to deal with tag redundancy. These methods aggregate redundant tags into the same clusters to deal with redundancy. Besides, ambiguity will be alleviated when an ambiguous word is in a cluster [64]. Using tags may lead to predictable recommendations. However, sometimes it is vital for a recommender system to provide a list including unexpected items such as music and video recommenders.

The fusion-based recommender system is another approach in recommendation systems that are used in discovering serendipitous items for users which are unexpected and valuable items. These items can excite users for the first time although he/she would not be able to discover them on his/her own or he/she was not interested in [57]. This approach usually takes the recommendation of both individual-based and collaborative filtering-based into account and provide such results.

Standard social annotation systems can employ different approaches to handle recommendations. These approaches include recency, authority, linkage, popularity, and vector space model. Vector model space is one of the popular approaches which is derived from information retrieval theory. In this schema, we model each user as a vector representing a set of tags which constructs user profile. Similarly, each resource is modeled as a vector. Using similarity calculation techniques such as the Jaccard similarity coefficient and Cosine similarity, a match between a user and a resource can be obtained [78].

Some researches employed nearest neighbor approaches as a strategy for recommendations. For instance, Yuan et al. [81] suggested  $k$  nearest neighbors to find the  $k$  most similar neighbors for target user or item. Based on these nearest neighbors, a prediction will be recommended, which is obtained by combining the preferences of these neighbors. There are different nearest neighbor methods such as  $k_tNN$  and  $Rk'_uNN$  that can be employed as well. Geo-tagged items are another issue for tag-aware recommender systems. One of the methods used in dealing with geo-tagged photos is trajectory pattern mining. Cai et al. [9] used this framework to construct people trajectories from geo-tagged photos, generate semantic trajectories associated with contextual environment semantics, and extract previous users's semantic trajectory patterns. Then, they used an online itinerary recommendation where it verifies the user's query, searches for related candidate itineraries and sorts and displays them.

#### **2.5.4 AI for Tag-Aware Recommender Systems**

Some recent researches employed neural networks and deep learning to deal with problems in tag-aware recommender systems and to tackle the limitations of conventional methods. Deep neural networks show excellent performance in deep features extraction from raw data. They are invariant to various local changes in input data. Having invariant extracted features potentially makes deep neural networks a greater predictive tool [5]. In tag-aware recommender systems, users' tags may change; however, their preferences are almost invariant. Therefore, abstract and invariant features extracted from the user's tag can be effectively utilized to predict future user's preferences [89]. Convolutional neural networks are developed mainly for speech recognition and large scale image classifications. However, they can be useful for developing recommender systems. As an example, this approach is suited for predicting latent factors for music recommendation [74]. This method is used because it allows intermediate features sharing and their hierarchical structure makes it possible to operate on multiple time scales. Researches that incorporate deep convolutional neural networks for recommender systems show that their proposed approach significantly outperforms conventional methods.

### **2.6 Applications**

#### **2.6.1 Music**

Music recommendation is one of the widely investigated areas in the recommender systems field. Online services provide not only a vast number of tracks for listeners, but also thousands of radio stations to choose from on a single web site. One of the essential sources of information to build a music recommender system is social tagging, which is the basis for recommender algorithms that works based on tag similarity. As

mentioned before, music recommender systems use music genres and periods tags, mood-based and context-oriented tagging, and band or singer name [30]. This area of application is still complicated because of the sheer variety of styles and genres as well as social and geographical factors, influencing users' preferences. Having users that prefer particular songs instead of albums makes the problem more sophisticated [74]. This shows the vital importance of the recommender systems to be efficient and accurate enough to satisfy music listeners.

### **2.6.2 Movies**

Movies and TV-series providers take advantage of recommender systems to suggest movies or TV-series to their users. As an example, Kim et al. [33] conducted a series of experiments using the data provided by MovieLens in which a total of 7601 movies, 4009 users and 16,529 distinct tags are available to develop the recommender system.

### **2.6.3 E-learning**

Some studies focus on the application of tag-based recommender systems for E-learning. E-learning environment is drastically growing in terms of available information and sophistication [48]. This emphasizes the importance of personalization in this area, which demands recommender systems to provide suggestions for users [36]. Recommender systems assist learners in finding relevant e-learning materials at the right time with the right content based on the user's available information and knowledge regarding activities [73]. The authors of [36] considered collaborative tagging systems to enhance the efficiency and effectiveness of traditional recommendation methods in an e-learning environment.

#### **2.6.4 Documents**

Further application of recommender systems is the document/article recommendation. To illustrate this, we use MedWorm as an example. MedWorm is a medical RSS feed provider that works as a search engine on the data collected from RSS feeds [52]. Xu et al. [78] used the article repository in the MedWorm system and extracted four types of data, including user, resource, tags, and quads to construct the dataset for social annotation to evaluate their proposed recommender system. The purpose of the work is to recommend articles to users based on their preferences using social annotated tags.

#### **2.6.5 Social Medias**

In almost all online social media sharing, such as Flickr and Instagram, users can annotate tags to the photos they upload, and these tags play an essential role in the popularity of the social contents [85]. However, a significant portion of users does not know how to tag their photos, as well as they, do not have enough time to annotate proper tags because it is a time-consuming process. Tag recommender systems help users to tag their photos and increase the chance of enhancing the popularity of their content [84].

#### **2.6.6 Food**

Online food suppliers use recommender systems to suggest food to their users using content-based [16] and collaborative filtering [50] methods. This domain is challenging for recommendation due to variety and complexity. First, there are a huge amount of different food items e.g., almost 1000 different vegetables. Second, food items are in a combination of dishes, and rarely a user purchase only a food item. Considering

the variety of food items, the combinatorial problem is exponentially large. More importantly, the users' preferences regarding food items depend on many factors that need to be taken into account [16]. These issues cannot be addressed by traditional recommendation systems that only offer generic advises increasing a need for efficient recommender systems to deal with them. This area is recently emerging and requires more research and investigation [17].

### **2.6.7 Tourism**

One of the areas that has not taken enough attention by the researchers is itinerary recommender systems that which tries to help in travel planning [80]. They provide suggestions regarding favorite places to visit, travel route, and stay times. In this area of application, using geotagged photos, Cai et al. [9] developed an itinerary recommender system. These geotagged photos reveal the movement and trails of their photo-takers. Considering the fact that geotagged photos are inherently spatiotemporal, sequential, and implicitly containing aspatial semantics, they proposed an itinerary recommender system with semantic trajectory pattern mining from geotagged photos. The output of the system is customized and targeted semantic-level itineraries under the user's constraints. Sun et al. [69] proposed a framework that utilizes photos shared by the user in online social networks to recommend top- $k$  tours to the user. The recommendation of tours is area by area instead of landmark by landmark.

### **2.6.8 Distribution of the Applications**

Figure 2.3 illustrates the distribution of the studies focusing on each area of application. As shown in this figure, music, movies, and web recommendation are the hottest areas for implementing recommender systems. This can show that music, movie, and

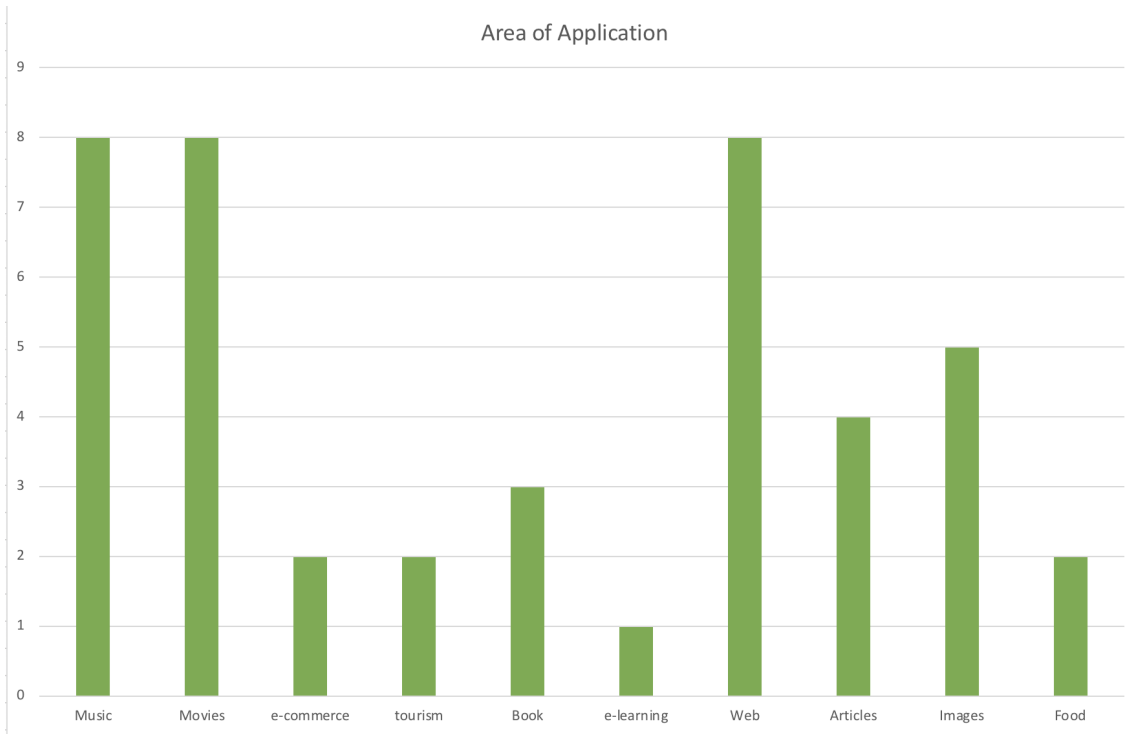


Figure 2.3: Areas of application for recommender systems

web resource service providers rely more on recommender systems. It also can show the potential for more research in those areas that have attracted less attention. For example, e-learning is getting more and more appealing as a source for self-education.

## 2.7 Evaluation Criteria

In terms of evaluation, there are several metrics available. These metrics include and not limited to, precision, recall, F-measure, and rankscore. Precision is the proportion of correct recommendations for a user over the whole set of recommended items. The number of correct item recommendations divided by the number of test items is called recall. F-measure is obtained using a harmonic mean between recall and precision, and finally, rankscore demonstrates the quality of ranking compared to the ideal item list [86]. Any other metric can be defined and used to assess the performance of the

recommender system, which will be introduced according to the nature of the problem and the proposed model. We describe the criteria used in the surveyed articles as follows.

- Root Mean Squared Error (RMSE) which can be formulated by Equation (2.1) [2].

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (t_i - p_i)^2} \quad (2.1)$$

where  $t_i$  is the test rating value and  $p_i$  is the predicted rating value.

- Mean Absolute Error (MSE) with Equation (2.2) [2].

$$MSE = \frac{1}{n} \sum_{i=1}^n |t_i - p_i| \quad (2.2)$$

- Mean Reciprocal Rank (MRR) demonstrates the capability of the system to return relevant tags at the top of the ranking (or the quality of top recommended tags) which is calculated by Equation (2.3) [3].

$$MRR = \max_{q \in Q} \frac{1}{c_q} \quad (2.3)$$

where  $c_q$  indicates the rank achieved by relevant tag  $q$ .

- Success at Rank  $k$  (S@ $k$ ) is the probability of finding relevant tag,  $q \in Q$ , in a set of top- $k$  recommended tags,  $C_k$  which is presented in Equation (2.4) [3].

$$S@k = \begin{cases} 1 & \text{if } Q \cap C_k \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \quad (2.4)$$

- Precision at Rank  $k$  (P@ $k$ ) demonstrates the percentage of relevant tags among all retrieved ones which is presented in Equation (2.5) [3].

$$P@k = \frac{|Q \cap C_k|}{|Q|} \quad (2.5)$$

- F-measure which utilize precision ( $p$ ) and recall ( $r$ ) is a comprehensive evaluation which can be calculated by Equation (2.6) [13].

$$p = \frac{T_R \cap T_U}{T_U}, r = \frac{T_R \cap T_U}{T_R}, F = \frac{2pr}{p+r} \quad (2.6)$$



where  $T_R$  is set of recommended tags and  $T_U$  is the set of user annotated tag.

- Ranking accuracy of user  $u$  at top- $k$  ranking,  $RK(u)@k$ , is a metric that is used to demonstrate if a tag with better rank is actually more relevant and it is calculated by Equation (2.7) [34].

$$RK(u)@k = \sum_{i \in T_{est}(u) \cap T_{op-k}(u)} \frac{1}{rank(i)} \quad (2.7)$$

where  $rank(i)$  denotes the rank of item  $i$  in top- $k$  list.

- Rank Score (RS) for user  $u$  and item  $i$  in a recommendation list with the length of  $L$  for the number of uncollected items by user  $u$  is defined by Equation (2.8).

$$RS_{u,i} = \frac{pos_i(L)}{B_u} \quad (2.8)$$

where  $pos_i(L)$  is the position of item  $i$  in the recommendation list. Obviously, lower values of RS are desirable.

Furthermore, computation time and cost for a system to return recommendations is an important evaluation criterion especially where the problem face a real time application or there is a large amount of data for computation [15].

Figure 2.4 shows the distribution of evaluation criteria used in the reviewed papers. The top 4 criteria are Precision, F1-score, MAE, and Recall. However, the majority of the papers use a combination of criteria to enhance their performance evaluation.

## 2.8 Future Research Directions

There are several research directions for future researches. One of the most critical challenges is dealing with computation time and cost. In each area of application, there is a massive amount of data that needs to be considered to develop recommender systems. User preference evolution makes this problem dynamic and more challenging, and there will be a need for faster and more efficient recommender systems. In the literature, studies mostly concentrate on proposing solutions for issues

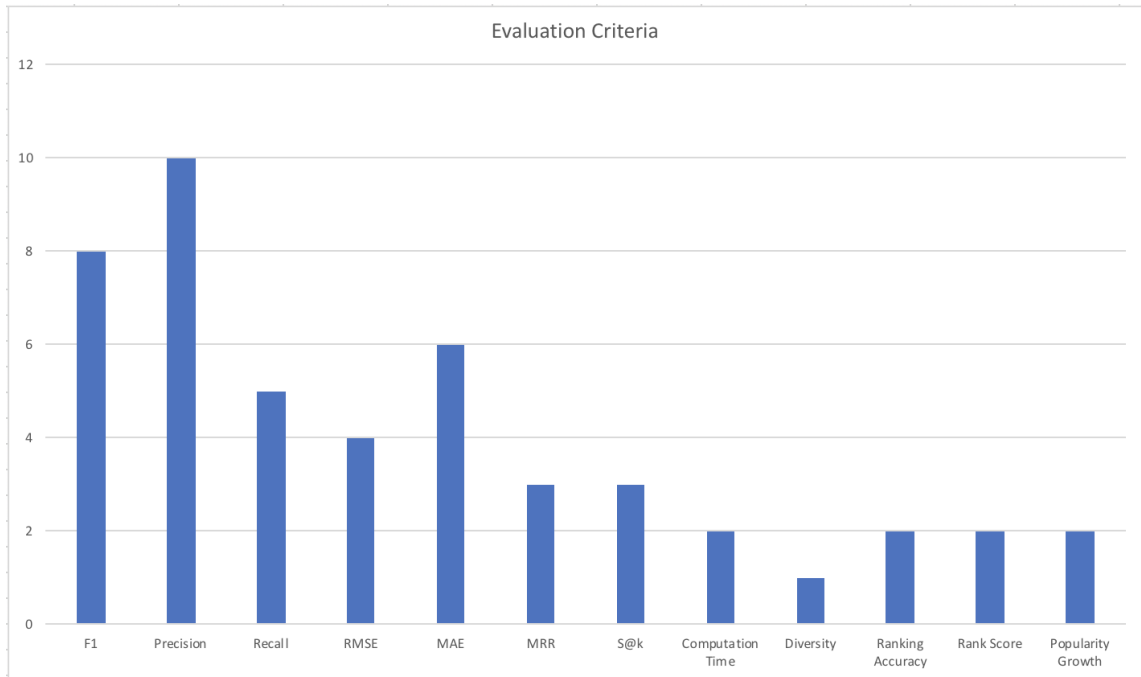


Figure 2.4: Evaluation Criteria

such as cold start and sparsity, and they do not focus on the efficiency of the models. Proposing and establishing recommender systems that can deal with a huge amount of data that updates constantly is a research area that is rarely investigated [47].

Another issue is that the proposed recommender systems are usually developed for a specific area of application. For almost all of the studies reviewed, there is a need to use the proposed system in other similar areas. An exciting future research direction can be providing a framework that covers different areas of applications.

Recently some studies are incorporating machine learning and neural networks to train and develop models for recommender systems. Different machine learning and deep learning models can be developed for recommendation systems and compare them to conventional methodologies to evaluate their performance.

Also, scoring functions are usually considered to be linear that may be unrealistic for some cases. Non-linear scoring functions can be used in the process of develop-

ing and implementing recommender systems, and the results can be compared and evaluated for the performance enhancement.

In addition, the evolution of the users' preferences is a critical factor in developing recommender systems. There are a few studies that focus on the evolution of the users' preferences, especially in tag recommendation systems.

Among the papers reviewed in this thesis, almost no paper investigated the consequences of recommender systems in terms of economics and econometrics. Economic models should be developed to study the recommender systems and evaluate the performance of these systems in terms of economic criteria. This can be a promising area of study in the recommendation systems field.

## **2.9 Chapter Conclusions**

Due to the recent growth of available online data on customers and clients as well as social tagging becoming popular, tag-aware recommender systems are attracting more attention increasingly. However, the latest review on the tag-aware recommender system is published in 2011, and there was a need to review the work that has been done since 2011. Consequently, we systematically defined goals and questions for reviewing the researches on tag-aware recommender systems, we followed defined protocol to retrieve and refine articles, and reviewed and analyzed the information extracted from collected papers. We first introduced challenges and problems that motivate researchers to develop new recommender systems. Also, we presented the methodologies and models they employ to tackle these questions. The application fields that tag-aware recommender systems have been developed for are covered. Last but not least, we discussed the limitations in the field of tag-aware recommender systems and depicted future research directions and opportunities that require researches to focus

on.

## Chapter 3

### Related Literature on Customer Reviews

Dealing with text as unstructured data is challenging. Natural Language Processing (NLP) is a branch of computer science and artificial intelligence (AI) concerned with processing and analyzing natural language data. Deep learning for NLP is one of the approaches that is improving the capability of the computer to understand human language [14]. There are a few studies that try to incorporate customer written reviews in generating recommendations.

Some researches on the integration of customer reviews in recommendation systems are under the category of aspect-based or aspect-aware recommender systems. As an aspect-based recommender system, Qiu et al. [59] proposed a model called aspect-based latent factor model which integrates ratings and review texts via latent factor model. The purpose of this research is predicting ratings by using aspects of information from users' and items' information. They constructed user-review and used it directly in the proposed model to provide rating predictions and recommendation lists. Besides, their model accomplishes a cross-domain task by transferring word embedding.

In another aspect-aware recommender system paper, the authors of [10] proposed another aspect-aware MF model that effectively combines reviews and ratings for rating predictions. It learns the latent topics from reviews and ratings without having

the constraint of a one-to-one mapping between latent factors and latent topics. Also, the model estimates aspect ratings and assign weights to the aspects. They performed experimental results on many real-world datasets and showed the performance of their models in accurately predicting the ratings.

Some aspect-based recommender systems utilize semantic analysis on reviews. For example, Bauman et al. [4] proposed a sentiment utility logistic model that uses sentiment analysis of user reviews where it predicts the sentiment that the user has about the item and then identifies the most valuable aspects of the user's possible experience with that item. For example, the system suggests a user going to a specific restaurant (as the primary recommendation), and also it recommends an aspect of that restaurant like the time to go to a restaurant (breakfast, lunch, or dinner) as a valuable aspect to the user (the secondary recommendation). The experimental results demonstrated the better experience of those users who followed the recommendations.

In the context of analyzing reviews, Susan et al. [71] analyzed customer reviews to find out what makes a review helpful to other customers. They analyzed 1,587 reviews from Amazon.com and indicated that extremity, depth of review, product type affect the perceived helpfulness of the review. While this research does not incorporate the reviews in making recommendations, it provides information that is potentially useful in developing recommender systems.

The authors of [44] proposed a new recommender system that integrates opinion mining and recommendations. They proposed a new feature and opinion extraction method based on the characteristics of online reviews which can address the problem of data sparseness. They used the part-of-speech tagging approach based on association rule mining for each review. They performed their empirical study on online restaurant customer reviews written in Chinese and illustrated the performance of

the proposed methods.

Ling et al. [42] considered using the review text using topic modeling techniques and align the topic with rating dimensions to enhance the prediction accuracy. They proposed a unified model combining content-based and collaborative filtering, which can deal with the cold-start problem. They applied the proposed framework to 27 classes of real-case datasets and showed the significant improvement of the recommendations comparing to the baseline methods.

The authors of [51] tried to incorporate the implicit tastes of each user in order to predict ratings as the text review justifies a user’s rating. They used latent review topics extracted from topic models as highly interpretable textual labels for latent rating dimensions. Also, they accurately predicted product ratings using the information extracted from the reviews, which can improve the recommendations for those that have too few ratings. Moreover, their discovered topics are useful in facilitating tasks such as automated genre discovery. In a similar study, Tan et al. [72] exploit textual review information along with ratings to model user preferences and item attributes in a shared topic space. They used an MF model for generating recommendations and used 26 real-case datasets to evaluate the performance of their model.

As presented above, none of the abovementioned studies used a deep neural network autoencoder to deal with the sparsity in the user-attributes matrix extracted from the reviews. To the best of the authors’ knowledge, this is the first study that extracts deep features from extracted latent topics from the textual user reviews to develop a recommender system. In the next section, we present the proposed approach.

## Chapter 4

### Methodology

In this chapter, we provide the proposed methodology for incorporating customer written reviews in developing recommender systems. Figure 4.1 depicts the general framework for transforming customer written reviews into a dense users-attributes matrix and predicting ratings using this matrix and users-items matrix. As described before, the idea of how to use customer written reviews is investigating what attributes of the product category are mentioned in the customer's review. In doing so, we need to match the review with a set of predefined product attributes. As Fig 4.1 demonstrates, we use Latent Dirichlet Allocation (LDA) to analyze the reviews on a product category and retrieve a dictionary of attributes. Afterward, we can construct the users-attributes matrix, which indicates what attributes the user has pointed out in his or her reviews in a binary format. The major challenge with this matrix is a well-known problem called sparsity. Besides, there are other problems, including ambiguity and redundancy, regarding the extracted attributes in the matrix. To deal with this problem, we propose a deep neural network approach to transform this sparse matrix into a dense matrix presenting a set of deep features extracted from the users-attributes matrix and construct the users-deep features matrix. We use this matrix and users-items matrix to predict ratings and generate recommendations via Matrix Factorization (MF) as a powerful and efficient collaborative filtering method.



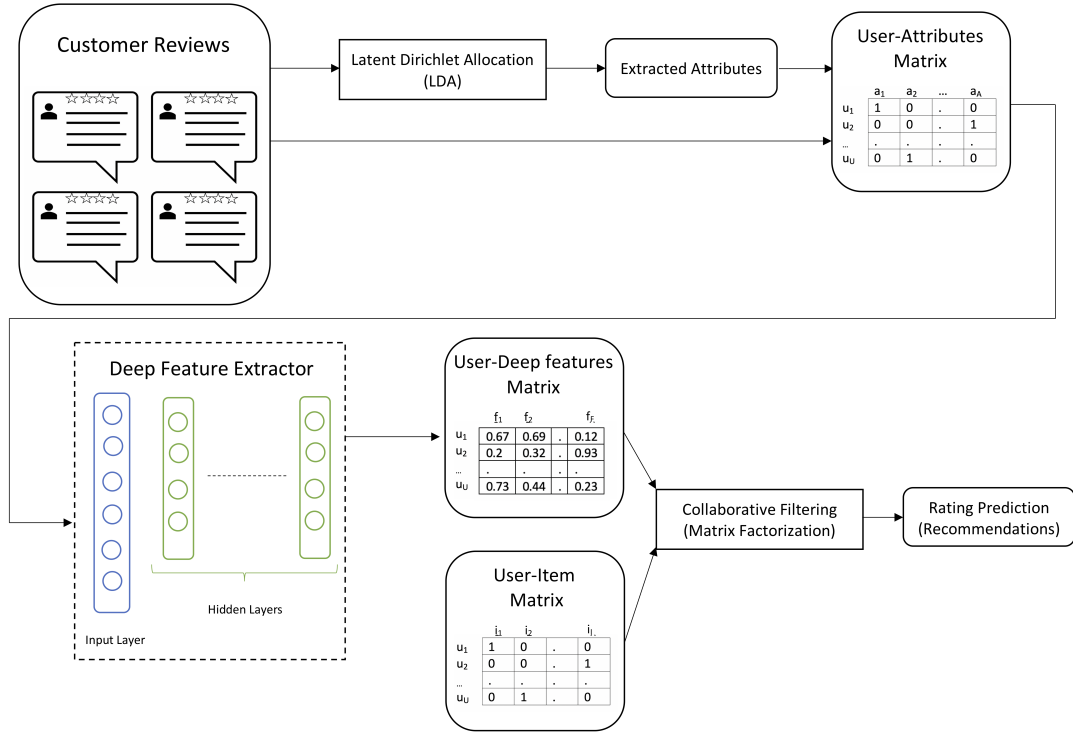


Figure 4.1: General framework of the proposed approach of using customer reviews in recommendation systems

In the following sections, we present and describe DLA, deep neural network model, and the MF method used in this research.

#### 4.1 Latent Dirichlet Allocation

The basic latent factor model predicts ratings for a user and item using user and item biases,  $K$ -dimensional user and item factors including the item's properties and the user's preferences minimizing the Mean Squared Error (MSE). There are a variety of methods for optimizing MSE for this problem, such as alternating least-squares and gradient-based methods [35]. While latent factor models try to uncover hidden

dimensions in review ratings, LDA aims to uncover hidden dimensions in the written part of the review. Introduced by Blei et al. [7]. LDA is a generative statistical model for topic modeling in the natural language processing (NLP) context. Topic modeling is the task of describing a collection of documents by identifying a set of topics. In LDA, we model each item of the collection as a finite mixture over an underlying set of topics as a three-level hierarchical Bayesian model. We also model each topic as an infinite mixture over an underlying set of topic probabilities which provides an explicit representation of a document [37]. In order to describe LDA, a set of documents  $d \in D$  and LDA associate each document with a  $K$ -dimensional stochastic vector as a topic distribution  $\theta_d$ . This association encodes the fraction of words in a document that discusses the topic  $k$  with the probability of  $\theta_{d,k}$ . LDA associate a word distribution,  $\phi_k$  to each topic to encode the probability of a word used for that topic. LDA assumes a Dirichlet distribution for the topic ( $\theta_d$ ). As a result of applying LDA, we have word distribution of each topic and topic distribution for each document. Having the word distribution and topic assignment of the words, we can calculate the likelihood of a corpus  $T$  as

$$p(T|\theta, \phi, z) = \prod_{d \in T} \prod_{j=1}^{N_d} \theta_{z_{d,j}, w_{d,j}} \quad (4.1)$$

where  $z$  is topic assignments updated via sampling. This likelihood is a product of the probability of the topic being the document and the word being the topic [51].

LDA results in a vast number of words from the reviews. Inspired by [51], we filter the extracted words using frequent itemsets using association rules to prune the set of words LDA provides. Association rule mining uses two metrics, including support and confidence where support is a measure that shows if the itemset appears in the dataset frequently, and confidence shows how often a rule can be found.

## 4.2 Deep Neural Networks

Sparsity is a significant problem in the recommender systems, which significantly reduces the performance of the rating prediction. The problem of sparsity is sometimes called gray sheep problem, which is peculiar to similarity-based collaborative recommendation systems. The problem arises from the fact that users-attributes interaction will occur for a tiny percentage of all possible interactions because the user only mentions a tiny portion of all the attributes in the written review [33] that makes some users not similar enough to others to discover their preferences. Hence, the system cannot retrieve proper recommendations. In this regard, many investigators have focused on dealing with this problem to provide a solution that mitigates the effect of sparsity [86, 19]. Here, we propose a deep neural network approach to deal with the sparsity in the users-attributes matrix and transform it into a dense matrix. Here, we describe the details of the proposed deep neural networks to process the attributes extracted using LDA.

The reason for using sparse coding is to learn more interpretable features for machine learning applications [40] and it helps at representing the input matrix as a weighted linear combination of a small number of basis vectors. The resulted matrix is capable of capturing high-level patterns that exist in the input layer. For instance, Le et al. [39] developed a sparse autoencoder as the result of combined sparse coding with the autoencoder. They implemented their idea by penalizing the deviation between the expected hidden representation and present average activation. In more relevant research, Zuo et al. [89] developed an autoencoder using deep neural networks for tag-aware recommender systems. Through experimental results, they demonstrated the usefulness of the sparse autoencoders for the recommendation algorithms.

Inspired by [89], an autoencoder constitutes an input layer, a hidden layer, and

an output layer. We can divide the autoencoder itself into an encoder and decoder. The encoder is the input layer and output layer, while the decoder is the hidden layer and the output layer. Figure 4.2 illustrates the purpose of the autoencoder, which is reconstructing the input data in the output layer with the same dimensionality. In other words, this follows an unsupervised learning framework.

Letting  $x_1, x_2, \dots, x_m$  be an unlabeled dataset, we can obtain the nonlinear representation of the input data using activation function [5]. Using a sigmoid activation for an unlabeled dataset  $x_i$ , the representation is

$$h(x_i; W, b) = \sigma(Wx_i + b) \quad (4.2)$$

where  $W$  denotes weight matrix,  $\sigma$  is the sigmoid activation function, and  $b$  is the bias term. This representation is also called the hyperbolic tangent function. On the other side, the decoder reconstructs the input into the output layer by minimizing the error between the input and the output layers. The minimizing term is defined in Equation (4.3).

$$\min \sum_{i=1}^m \|\sigma(W^T h(x_i; W, b) + c) - x_i\|^2 \quad (4.3)$$

where  $m$  denotes the number of examples. Since the minimization is a convex function, we can obtain the optimal value. We can calculate the sparsity penalty term by Kullback-Leibler divergence between a preferred activation ratio in the hidden layer and the desired hidden representations [26] using Equation (4.4).

$$P = \sum_{j=1}^n D_{KL}(\rho || \hat{\rho}_j) \quad (4.4)$$

in which  $\rho$  is a reset average activation that is set to be close to zero in practice, and  $n$  is the number of hidden units. Also,  $D_{KL}(\rho || \hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1-\rho}{1-\hat{\rho}_j}$ .

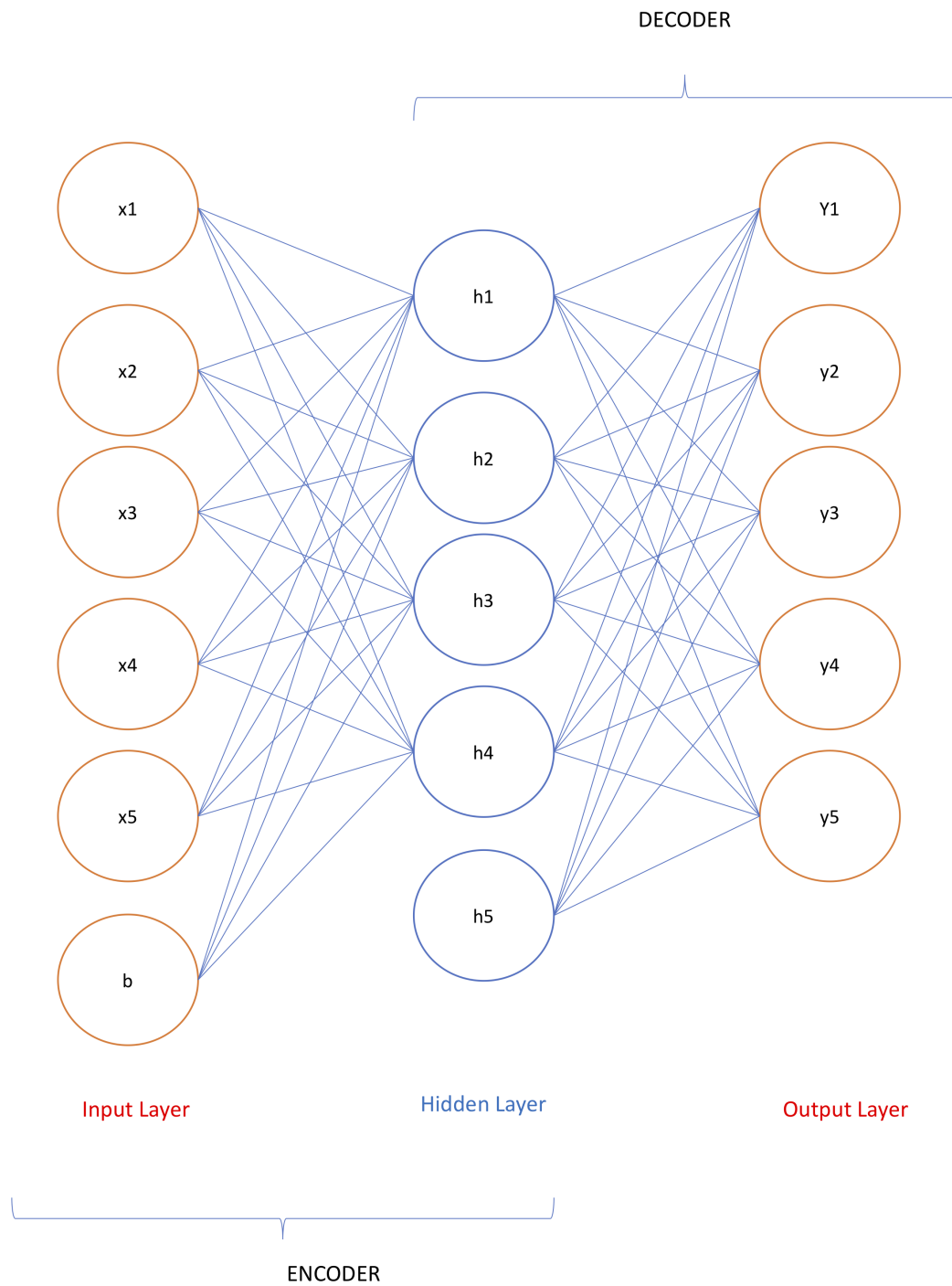


Figure 4.2: Demonstration of a simple autoencoder

Combining the objective function and penalty term introduced above, we obtain the final objective function for the autoencoder using Equation (4.5).

$$\min \sum_{i=1}^m \|\sigma(W^T h(x_i; W, b) + c) - x_i\|^2 + \beta P \quad (4.5)$$

where  $\beta$  is a hyperparameter to change the weight of the penalty term.

In order to transform this architecture into an autoencoder using a *deep* neural network, we need to use more layers as hidden layers where the output of each layer is the input for the next layer. In other words, the procedure and calculations explained above will be followed for more than one time to the result of each implementation. The input data will train the first hidden layer, and the output layer of the first hidden layer will serve as the input of the second hidden layer. We iterate these steps based on the number of hidden layers considered for the autoencoder. We use this deep neural network, which serves as the sparse autoencoder, to extract deep features from the set of retrieved attributes of a product category.

Another advantage of using this approach is that the number of features in the final output layer can be less than the number of attributes from the users-attributes matrix. Having a lower dimensionality can speed-up the learning process when the predictive model is dealing with a large dataset. Besides, the model can potentially reduce the deficiencies caused by ambiguity and redundancy in the set of attributes. These characteristics of the deep neural network significantly enhance the quality of the recommendation list for the users.

### 4.3 Collaborative Filtering

The last step in generating recommendations is using the MF model to predict ratings. First, we update the user profile based on the users-deep features matrix obtained from

the deep neural network. Conventional collaborative filtering models only use user-item or user-attributes matrix to generate a recommendation list. In order to use both users-items and users-deep features information, we employ the approach developed by Ricci et al. [60]. In this method, we should find the target user neighborhood  $N_u$  based on the similarity between the target user and other users using Equation (4.6).

$$sim_{u,v} = \frac{\hat{X}_u \cdot \hat{X}_v}{\|\hat{X}_u\| \|\hat{X}_v\|} \quad (4.6)$$

Having the similarity matrix between users, we can predict the rating of the target user using a weighted average of ratings from the neighbor users using Equation (4.7).

$$S_{u,i} = \sum_{v \in N_u} (\pi_{UI} Y)_{v,i} \quad (4.7)$$

The final and easy step is to sort the predicted ratings for items and generate the list of the recommendations according to the size of the list,  $n$ . Please note that using this approach, we are exploiting the ternary relation between users-attributes-items [89].

## Chapter 5

### Experimental Results

#### 5.0.1 Dataset

In this research, we use the Amazon Review dataset [23]. This dataset contains 142.8 million reviews on the Amazon products between May 1996 and July 2014 along with users profile and item metadata. The dataset includes the ID of the reviewer, the ID of the product (ASIN), name of the reviewer, helpfulness rating of the review, text of the review, rating of the product, a summary of the review, and time of the review. It also has the name of the product, price in US dollars, related products, sales rank information, brand name, and the list of the categories of the product. Table 5.1 presents the statistics of the Amazon Reviews dataset separated by each product category [42]. Also, Figure 5.1 demonstrates the sparsity of the reviews in the dataset where the percentage for a product category indicates the percentage of the users with no more than three ratings [72]. This sparsity can reduce the performance of recommender systems drastically. On average, there is roughly an average of 120 words in each review.

As the dataset preprocessing, there are many users without having any written review. We removed these observations from the dataset. For processing the data and implementing the proposed methodology, we used Python 3. As the cleaning up step, we removed punctuations and stop-words using NLTK stopwords. Words



Table 5.1: General statistics of the Amazon Review dataset

Dataset	Users(x10 <sup>3</sup> )	Items(x10 <sup>3</sup> )	Reviews(x10 <sup>3</sup> )	Words(x10 <sup>3</sup> )	Words per Review	Reviews per Item
Arts	24	4	28	2,007	71.73	6.64
Jewelry	41	19	59	3,101	52.9	3.12
Industrial Scientific	30	23	137	6,920	50.5	6.06
Watches	62	10	68	5,437	79.53	6.62
Cell Phones and Accessories	68	7	79	7,568	95.88	10.61
Musical Instruments	67	14	85	7,442	87.14	6.02
Software	68	11	95	11,013	115.82	8.46
Gourmet Foods	113	23	155	10,543	68.18	6.59
Office Products	110	14	138	11,206	81.16	9.71
Automotive	133	48	189	13,250	70.21	3.97
Patio	167	20	206	17,291	83.83	10.56
Pet Supplies	160	18	217	18,684	86.03	12.39
Beauty	168	29	252	17,890	70.97	8.69
Shoes	74	48	390	23,604	60.54	8.05
Kindle Store	116	4	161	21,533	133.92	36.78
Clothing and Accessories	129	66	582	34,267	58.89	8.77
Health	312	40	429	33,277	77.61	10.84
Toys and Games	291	54	436	35,034	80.35	8.13
Tools and Home Improvement	284	51	409	34,591	84.47	8.03
Sports and Outdoors	329	68	511	38,899	76.12	7.48
Video Games	229	21	464	55,532	119.77	22.05
Home and Kitchen	645	79	992	81,923	82.6	12.55
Amazon Instant Video	313	22	718	88,958	123.96	32.32
Electronics	811	82	1,242	124,065	99.91	15.13
Music	1,135	557	6,396	774,791	121.13	11.49
Movies and TV	1,224	213	7,850	997,262	127.04	36.88
Books	2,589	929	12,886	1,613,604	125.22	13.87
<b>All Categories</b>	<b>6,644</b>	<b>2,441</b>	<b>34,687</b>	<b>4,053,796</b>	<b>116.87</b>	<b>14.21</b>

that have appeared in the review corpus of a product category only once are most likely irrelevant; thus, we eliminated these words as well. Using the rest of tokens, we construct our preliminary dictionary of attributes. Note that we create a separate dictionary for each product category.

For the training part, we selected 80% of the dataset for training, 10% for validation, and 10% for testing, randomly. Furthermore, we selected 25 topics and 40 words for each topic when we applied LDA to each category review corpus. Then, we used the association rule mining technique to extract frequent itemsets from unique words obtained after the LDA step. Finally, we matched the reviews of each user with the set of extracted words and constructed the users-attributes matrix. For the rest of the parameters required to apply the deep neural network feature extractor and matrix factorization method, the hyperparameters are as follows.

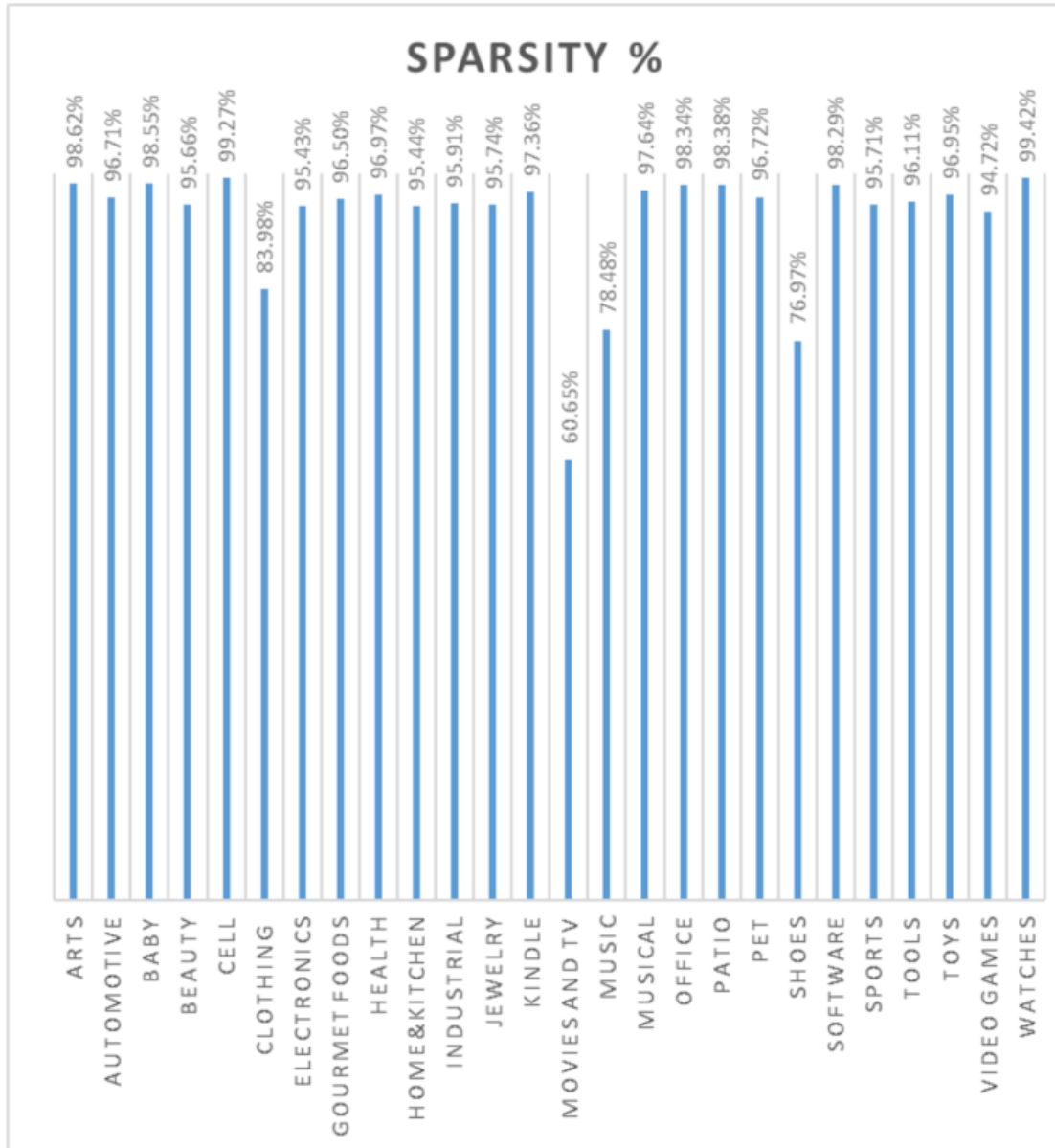


Figure 5.1: Sparsity of the Amazon Reviews Dataset

- The number of hidden layers is 2/3.
- The number of neurons in the first layer is 1000.
- The number of neurons in the second and third layer is 800.
- Average activation is 0.2.

In order to obtain these values, we changed one hyperparameter in a reasonable range

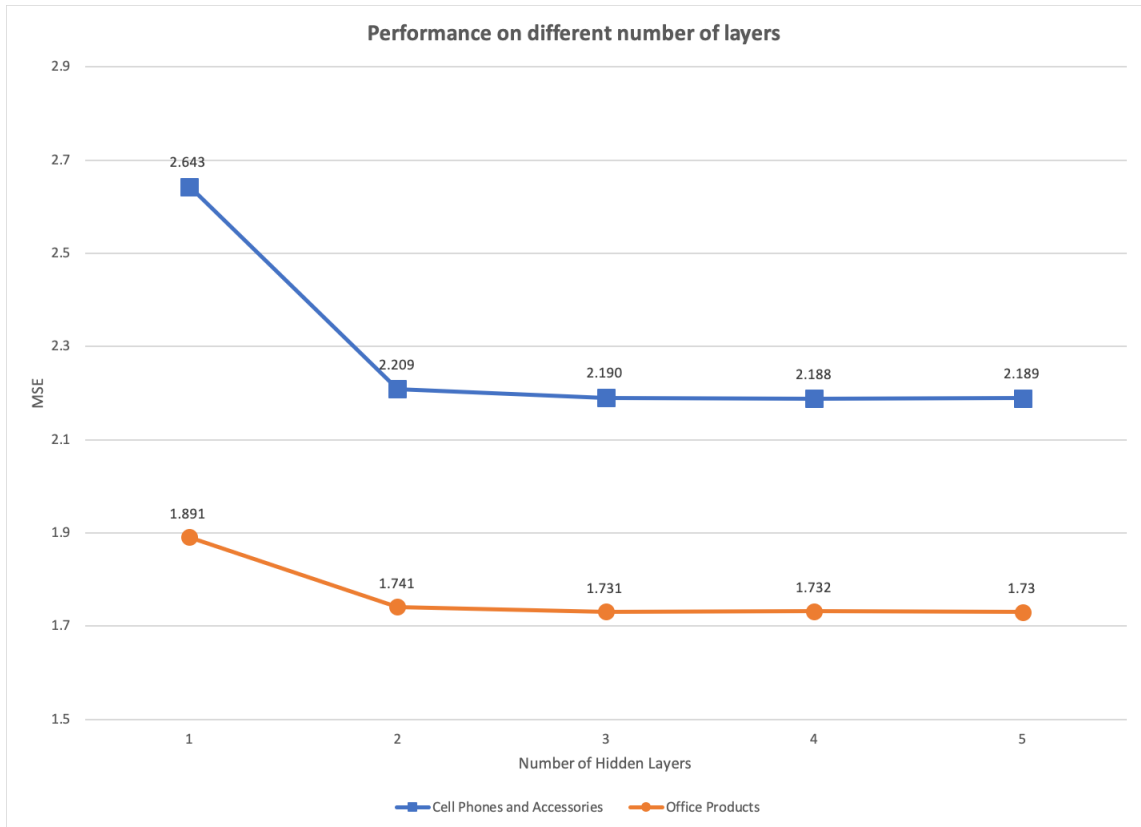


Figure 5.2: Analysis of MSE based on the number of hidden layers

to find a value that provides the best performance while we fix other hyperparameters only on one of the product categories. For the number of hidden layers, both two and three hidden layers show high performance. During the performance evaluation, we performed both on a product category to find the best results. Figs. 5.2, 5.3, and 5.4 demonstrate the MSE on two product datasets used to tune the hyperparameters. As you can see, MSE is not improving after using three hidden layers. MSE stops improving significantly at 1000 and 800 neurons in the first and second hidden layers, respectively.

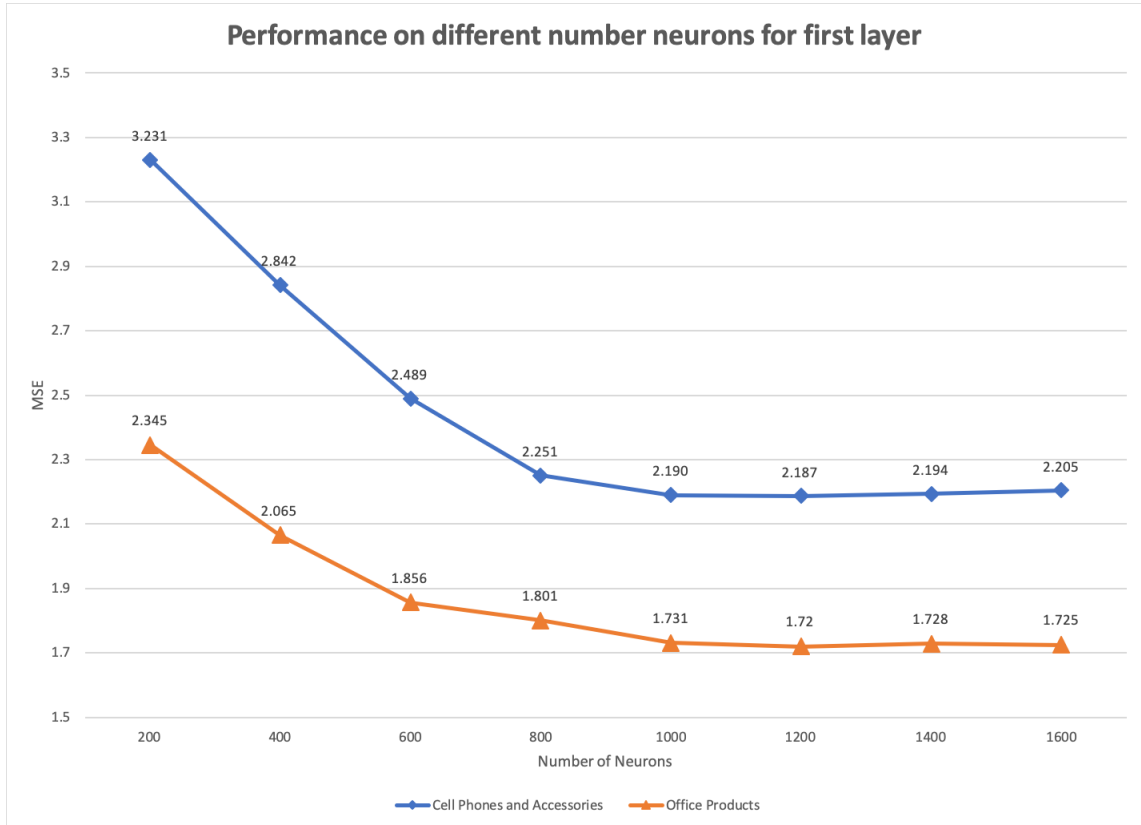


Figure 5.3: Analysis of MSE based on the number of neurons in the first hidden layers

### 5.0.2 Baseline Methods

We compare the performance of our model with three other state-of-the-art models, including MF, the Hidden Topics and Factors (HTF), and the Ratings Meet Review (RMR). The following is the explanation of these models.

- **MF** is the standard and widely used matrix factorization model. We consider the model proposed and described in [55]. This model uses the ratings of the user in generating recommendations, and the written part of the customer’s feedback is not incorporated.
- **HTF** is a model proposed by [51] that incorporates the review text with the stochastic topic distribution modeling which can be applied either on users or items. It also employs matrix factorization to deal with the ratings.
- **RMR** is a hybrid model constituted of content-based filtering and collaborative filtering suggested by [42]. This model tries to exploit the information from

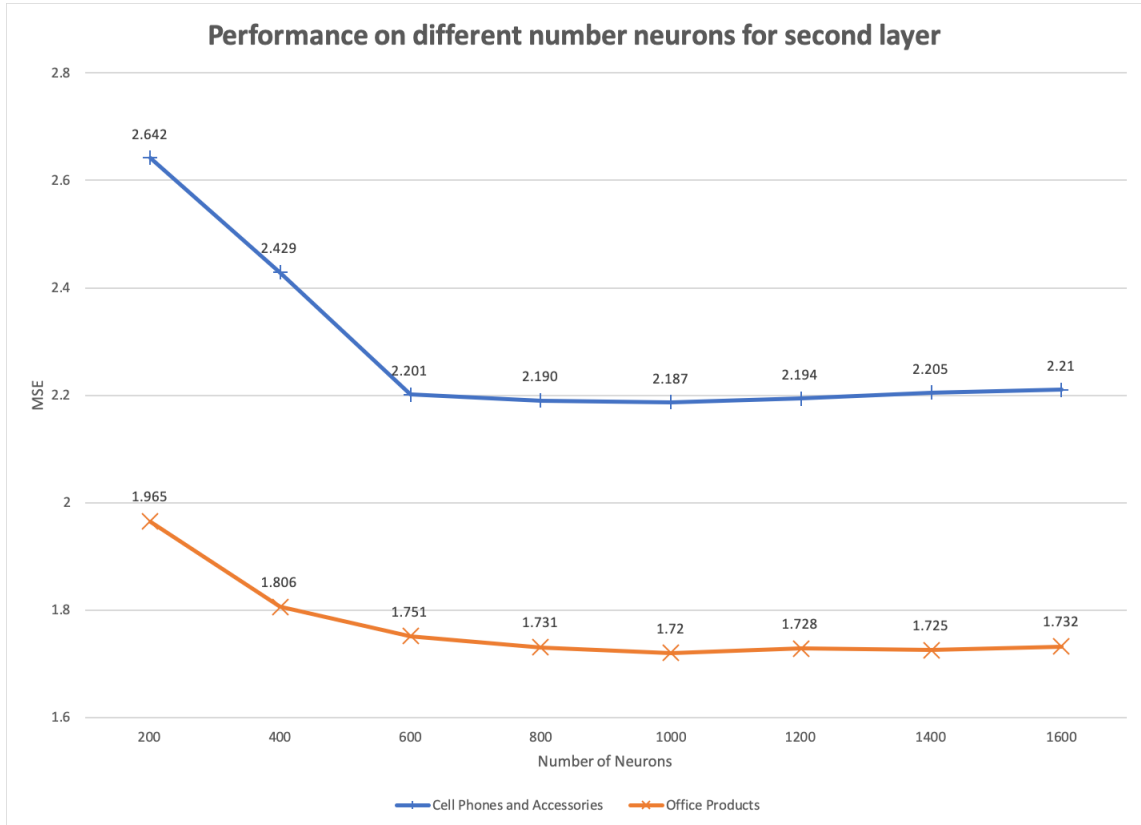


Figure 5.4: Analysis of MSE based on the number of neurons in the second hidden layers

reviews and improve the recommendation list accuracy across various classes of datasets. They tried to address the cold-start problem with collapsed Gibbs sampler for learning the model parameters.

For each product category, we report the performance of each model against our model. We consider Mean Squared Error (MSE) for evaluation of these models against the proposed approach.

### 5.0.3 Evaluation

We applied the proposed deep feature extractor method to all the product categories datasets and obtained the best MSE for our model and compared these results with our baselines. Figure 5.5 and Table 5.2 demonstrate the results.

Table 5.2: MSE of the proposed method vs baselines and the percentage of the improvement

Dataset	MF	HFT	RMR	Proposed Method	Improvement vs		
					MF	HFT	RMR
Amazon Instant Video	1.437	1.367	1.377	<b>1.359</b>	-5.43%	-0.59%	-1.31%
Arts	1.672	1.497	1.478	<b>1.457</b>	-12.89%	-2.70%	-1.45%
Automotive	1.677	1.539	1.51	<b>1.491</b>	-11.08%	-3.11%	-1.25%
Beauty	1.506	1.465	1.441	<b>1.406</b>	-6.66%	-4.05%	-2.45%
Books	<b>1.214</b>	1.245	1.22	1.221	0.58%	-1.92%	0.09%
Cell Phones and Accessories	2.337	2.242	2.192	<b>2.190</b>	-6.28%	-2.31%	-0.08%
Clothing and Accessories	0.5	0.456	0.443	<b>0.421</b>	-15.76%	-7.63%	-4.92%
Electronics	1.935	1.829	1.829	<b>1.806</b>	-6.68%	-1.28%	-1.28%
Gourmet Foods	1.622	<b>1.564</b>	1.572	1.578	-2.72%	0.89%	0.38%
Health	1.722	1.645	<b>1.619</b>	1.620	-5.92%	-1.51%	0.07%
Home and Kitchen	1.735	1.638	1.608	<b>1.593</b>	-8.19%	-2.75%	-0.93%
Industrial Scientific	0.568	0.466	0.469	<b>0.453</b>	-20.19%	-2.72%	-3.34%
Jewelry	1.364	1.284	1.267	<b>1.211</b>	-11.24%	-5.71%	-4.45%
Kindle Store	1.66	1.544	1.519	<b>1.497</b>	-9.83%	-3.05%	-1.46%
Movies and TV	<b>1.226</b>	<b>1.226</b>	1.227	1.249	1.87%	1.87%	1.79%
Music	<b>1.063</b>	1.087	1.066	1.064	0.09%	-2.12%	-0.19%
Musical Instruments	1.613	1.502	1.481	<b>1.475</b>	-8.57%	-1.81%	-0.42%
Office Products	1.921	1.776	1.745	<b>1.731</b>	-9.91%	-2.56%	-0.82%
Patio	1.878	1.805	1.776	<b>1.681</b>	-10.50%	-6.89%	-5.36%
Pet Supplies	1.807	1.69	1.669	<b>1.593</b>	-11.84%	-5.74%	-4.55%
Shoes	0.412	0.354	0.358	<b>0.339</b>	-17.71%	-4.22%	-5.29%
Software	2.516	2.326	2.28	<b>2.279</b>	-9.42%	-2.02%	-0.04%
Sports and Outdoors	1.326	1.245	1.236	<b>1.190</b>	-10.28%	-4.44%	-3.75%
Tools and Home	1.707	1.617	1.598	<b>1.519</b>	-10.99%	-6.04%	-4.92%
Toys and Games	1.574	1.477	1.479	<b>1.459</b>	-7.32%	-1.23%	-1.37%
Video Games	1.717	1.635	1.617	<b>1.581</b>	-7.90%	-3.28%	-2.20%
Watches	1.642	1.595	1.565	<b>1.470</b>	-10.48%	-7.84%	-6.08%
<b>Average</b>					<b>-8.71%</b>	<b>-3.14%</b>	<b>-2.06%</b>

As you can see in these figures, the proposed method performs better for most of the product categories. Comparing to the MF model, our method is capable of predicting ratings with an average of 8.71% improvement, in some cases up to 20.19%. In three cases, MF shows better performance, including Books, Movies and TV, and Music. For Books and Music categories, the model is off only by less than 1 percent, which implies that the performance of the model is close to the best performance for these two categories. A similar situation is happening between the HFT model and the proposed approach. Our deep neural network model beats the HFT model predictions for most of the cases. On average, our model improves the predictions by 3.14%. For the only two cases that our model is not performing better, the

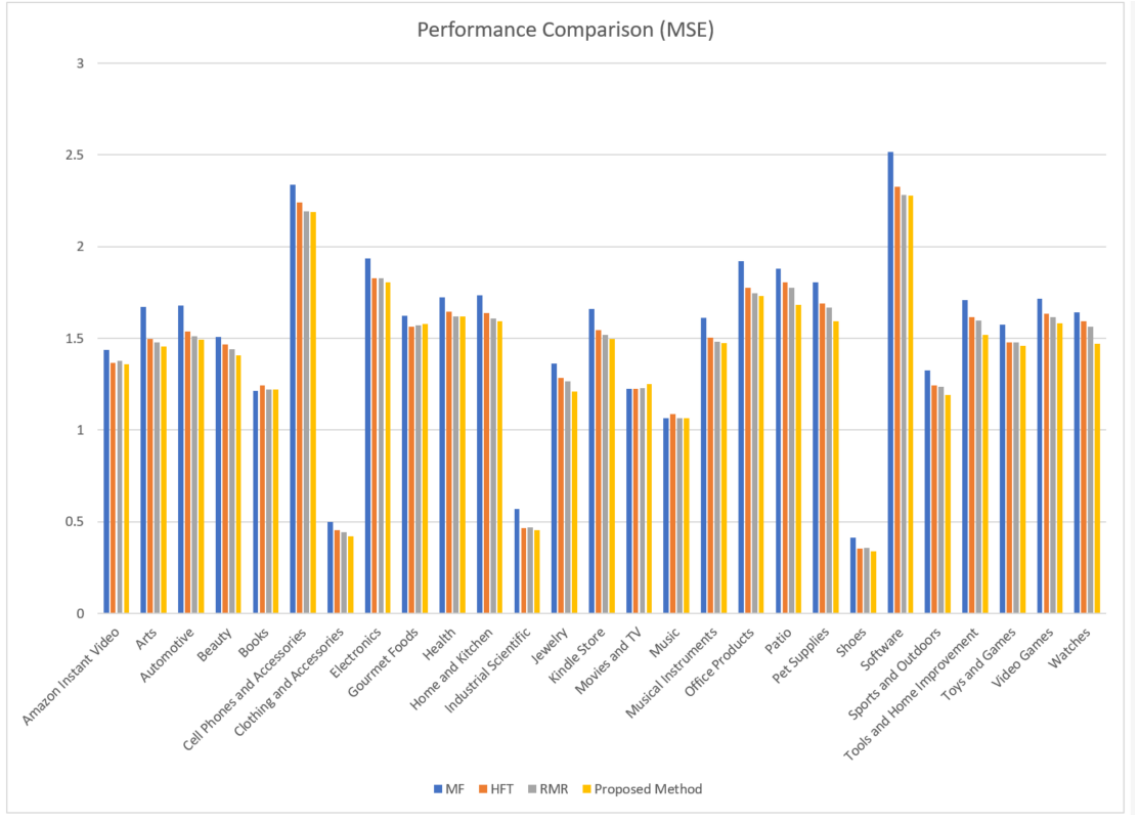


Figure 5.5: Comparing the MSE from the proposed method and the baselines

performance is close enough. In the worst case, which is the Movies and TV product category, MF and HFT models performed only 1.87% better than our model. Finally, our model outperforms the RMR method by 2.06% on average. Figs 5.6, 5.7, and 5.8 illustrate the improvements made by our model compared to MF, HFT, and RMR baseline models, respectively.

We investigated the product categories that our model is less accurate comparing the baselines. We suggest that the reason for this minor inaccuracy is the existence of more non-technical terms than technical attributes in the users-attributes matrix. For example, in the category of Industrial Scientific, we have a significant improvement between the MF model and the proposed method equal to 20.19%. Moreover, the performance improvement is higher for other categories that customers talk more

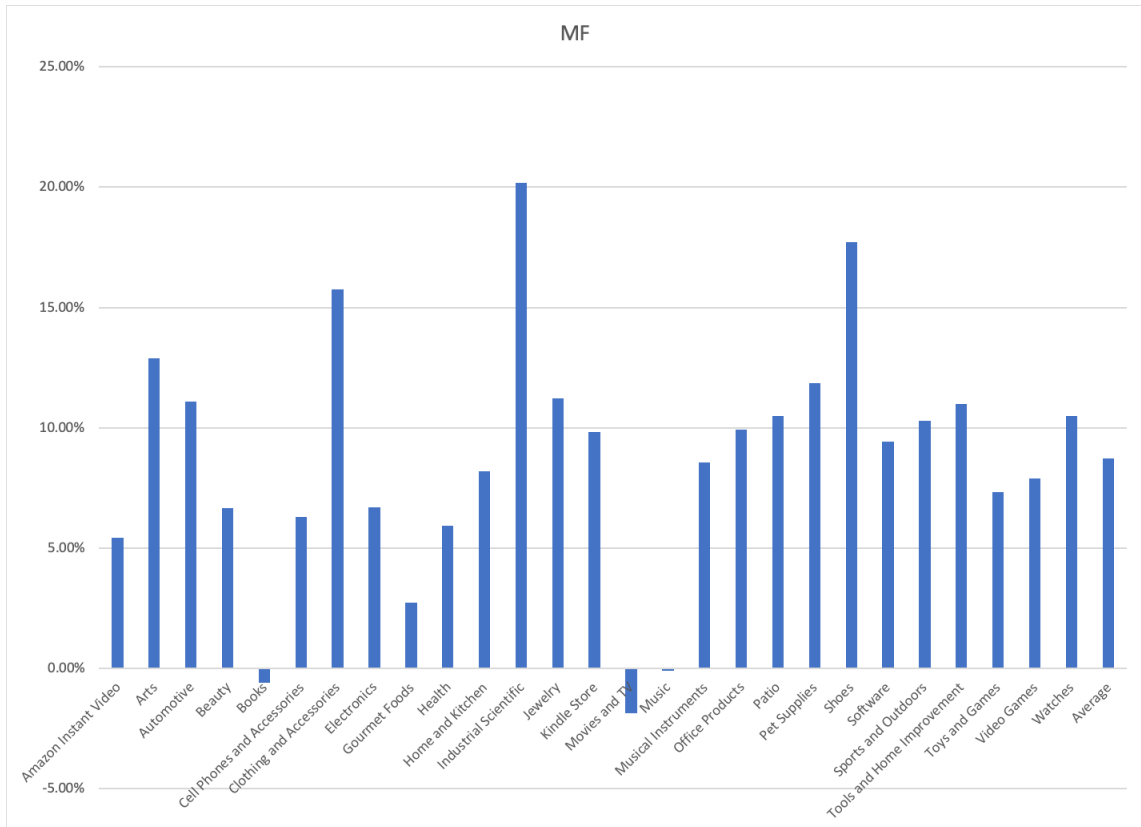


Figure 5.6: Proposed model improvement compared to MF

about product attributes such as Tools and Clothing. The superiority of the proposed model is the fact that the deep neural feature extractor retrieves the deep features and models the extracted words in a way that makes the users-attributes more informative, hence, extracting non-trivial relation between users based on the reviews they write. Our model can benefit e-commerce businesses through increasing revenue and customer satisfaction as recommendation plays a crucial role in real systems.



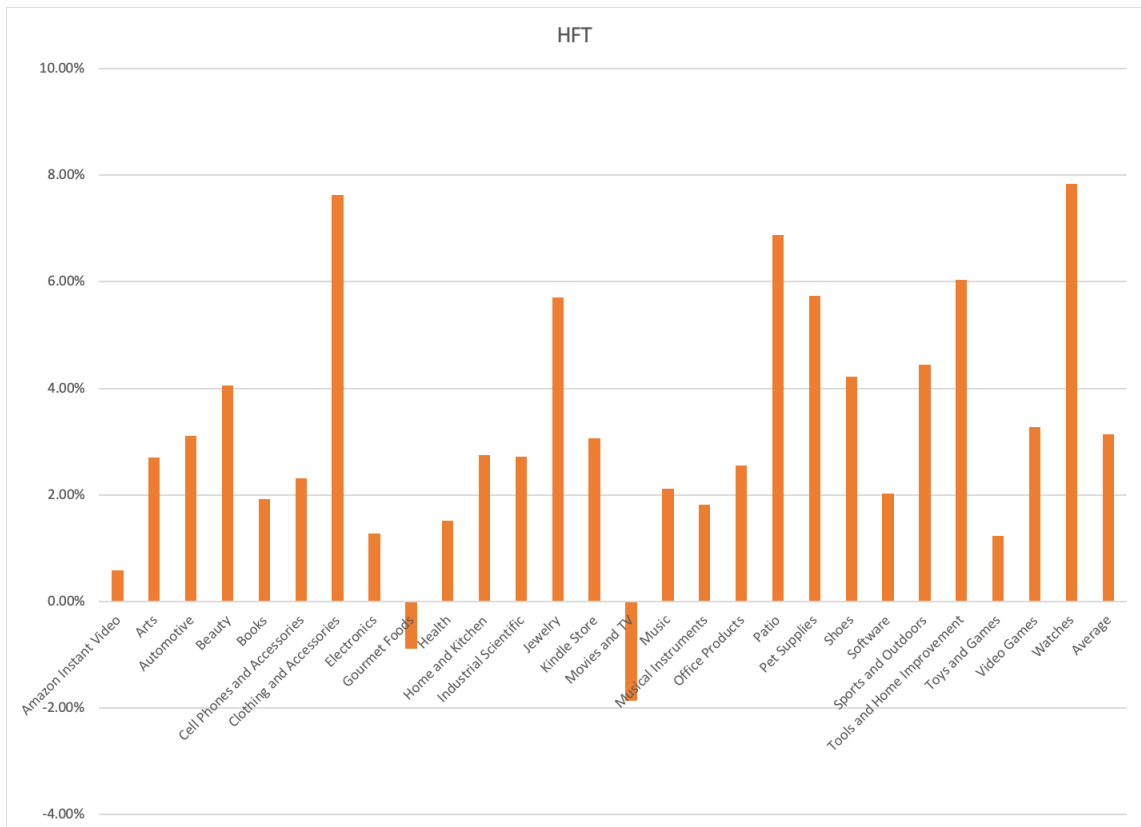


Figure 5.7: Proposed model improvement compared to HFT

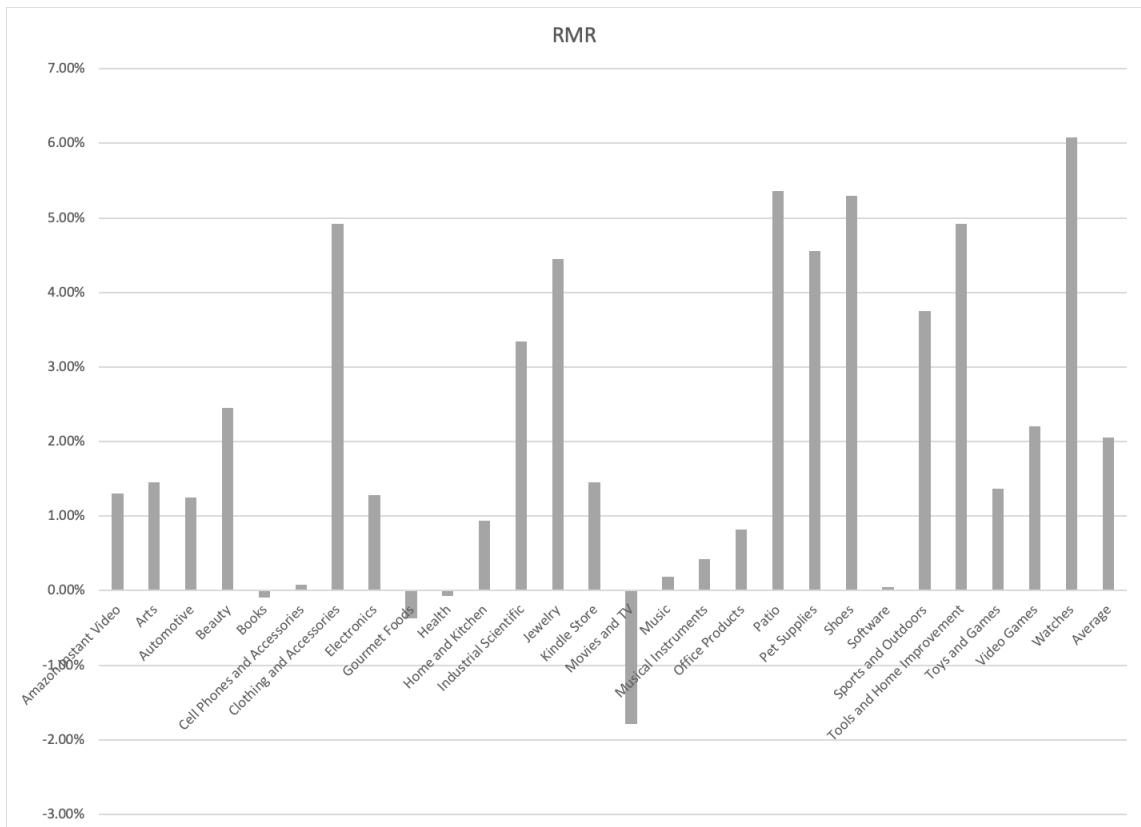


Figure 5.8: Proposed model improvement compared to RMR

## **Chapter 6**

### **Future Work and Conclusion**

Due to the recent growth of available online data on customers and clients as well as social tagging becoming popular, tag-aware recommender systems are attracting more attention increasingly. However, the latest review on the tag-aware recommender system is published in 2011, and there was a need to review the work that has been done since 2011. Consequently, we systematically defined goals and questions for reviewing the researches conducted regarding tag-aware recommender systems, we followed defined protocol to retrieve and refine articles, and reviewed and analyzed the information extracted from extracted papers. We first introduced challenges and problems that motivate researchers to develop new recommender systems. Also, we presented the methodologies and models they employ to tackle these questions. The application fields that tag-aware recommender systems have been developed for are covered. Last but not least, we discussed the limitations in the field of tag-aware recommender systems and depicted future research directions and opportunities that require researches to focus on.

In addition, we proposed a deep neural network approach to incorporate customer reviews in developing recommender systems. In our proposed model, we use Latent Dirichlet Allocation to extract attributes related to each product category. Then, we used association rule mining to use frequent terms in the dataset. Having the

set of extracted attributes, we constructed a users-attributes matrix. This matrix suffers from the sparsity problem. To deal with this challenge, we proposed a deep neural network that transforms the sparse users-attributes matrix into a dense users-deep features matrix, as an unsupervised learning tool. Finally, we used matrix factorization to predict ratings. We evaluated the performance of our model using the Amazon Review dataset, which is the largest dataset for customer reviews categorized for each product category. We also compared the MSE of our model with three baseline models from the literature, including MF, HFT, and RMR models. Our model outperforms these state-of-the-art models for most datasets.

For the future research directions, we are going to apply a deep neural network as the predicting model instead of the deep neural autoencoder and the matrix factorization method to improve the predictive power of our approach. Besides, we will investigate the application of other natural language processing tools for the construction of users-attributes matrix and compare their performance with current research.

## BIBLIOGRAPHY

- [1] BAGHERI, A., SARAEE, M., AND DE JONG, F. Care more about customers: Unsupervised domain-independent aspect detection for sentiment analysis of customer reviews. *Knowledge-Based Systems* 52 (2013), 201–213.
- [2] BANG, H., AND LEE, J.-H. Collective matrix factorization using tag embedding for effective recommender system. In *Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems, 2016 Joint 8th International Conference on* (2016), IEEE, pp. 846–850.
- [3] BARALIS, E., CAGLIERO, L., CERQUITELLI, T., CHIUSANO, S., GARZA, P., ANTONELLI, D., BRUNO, G., AND MAHOTO, N. A. Personalized tag recommendation based on generalized rules.
- [4] BAUMAN, K., LIU, B., AND TUZHILIN, A. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2017), ACM, pp. 717–725.
- [5] BENGIO, Y., COURVILLE, A., AND VINCENT, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2013), 1798–1828.
- [6] BISCHOFF, K., FIRAN, C. S., NEJDL, W., AND PAIU, R. Can all tags be used for search? In *Proceedings of the 17th ACM conference on Information and knowledge management* (2008), ACM, pp. 193–202.
- [7] BLEI, D. M., NG, A. Y., AND JORDAN, M. I. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.
- [8] CAGLIERO, L., FIORI, A., AND GRIMAUDO, L. Personalized tag recommendation based on generalized rules. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 1 (2013), 12.
- [9] CAI, G., LEE, K., AND LEE, I. Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. *Expert Systems with Applications* 94 (2018), 32–40.

- [10] CHENG, Z., DING, Y., ZHU, L., AND KANKANHALLI, M. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web* (2018), International World Wide Web Conferences Steering Committee, pp. 639–648.
- [11] CHRISTAKOPOULOU, K. Towards recommendation systems with real-world constraints.
- [12] DARABI, M., AND TABRIZI, N. An ontology-based framework to model user interests. In *Computational Science and Computational Intelligence (CSCI), 2016 International Conference on* (2016), IEEE, pp. 398–403.
- [13] FAN, M., ZHOU, Q., AND ZHENG, T. F. Content-based semantic tag ranking for recommendation. In *Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01* (2012), IEEE Computer Society, pp. 292–296.
- [14] FATHI, E., AND SHOJA, B. M. Deep neural networks for natural language processing. *Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications 38* (2018), 229.
- [15] FONT, F., SERRÀ, J., AND SERRA, X. Analysis of the impact of a tag recommendation system in a real-world folksonomy. *ACM Transactions on Intelligent Systems and Technology (TIST)* 7, 1 (2015), 6.
- [16] FREYNE, J., AND BERKOVSKY, S. Intelligent food planning: personalized recipe recommendation. In *Proceedings of the 15th international conference on Intelligent user interfaces* (2010), ACM, pp. 321–324.
- [17] GE, M., ELAHI, M., FERNAÁNDEZ-TOBÍAS, I., RICCI, F., AND MASSIMO, D. Using tags and latent factors in a food recommender system. In *Proceedings of the 5th International Conference on Digital Health 2015* (2015), ACM, pp. 105–112.
- [18] GHABAYEN, A. S., AND NOAH, S. A. M. Using tags for measuring the semantic similarity of users to enhance collaborative filtering recommender systems. *International Journal on Advanced Science, Engineering and Information Technology* 7, 6 (2017), 2063–2070.
- [19] GHAZANFAR, M. A., AND PRÜGEL-BENNETT, A. Leveraging clustering approaches to solve the gray-sheep users problem in recommender systems. *Expert Systems with Applications* 41, 7 (2014), 3261–3275.
- [20] GOLDBERG, D., NICHOLS, D., OKI, B. M., AND TERRY, D. Using collaborative filtering to weave an information tapestry. *Communications of the ACM* 35, 12 (1992), 61–70.

- [21] GOLDER, S. A., AND HUBERMAN, B. A. The structure of collaborative tagging systems.
- [22] GUEYE, M., ABDESSALEM, T., AND NAACKE, H. A parameter-free algorithm for an optimized tag recommendation list size. In *Proceedings of the 8th ACM Conference on Recommender Systems* (2014), ACM, pp. 233–240.
- [23] HE, R., AND MCAULEY, J. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web* (2016), International World Wide Web Conferences Steering Committee, pp. 507–517.
- [24] HE, R., AND MCAULEY, J. Vbpr: visual bayesian personalized ranking from implicit feedback. In *Thirtieth AAAI Conference on Artificial Intelligence* (2016).
- [25] HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G., AND RIEDL, J. T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)* 22, 1 (2004), 5–53.
- [26] HINTON, G. E. A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*. Springer, 2012, pp. 599–619.
- [27] HONG, H.-K., PARK, K.-W., AND LEE, D.-H. Tag recommendation system for multimedia retrieval in mobile environment. In *Consumer Electronics (ISCE 2014), The 18th IEEE International Symposium on* (2014), IEEE, pp. 1–2.
- [28] HORSBURGH, B., CRAW, S., AND MASSIE, S. Learning pseudo-tags to augment sparse tagging in hybrid music recommender systems. *Artificial Intelligence* 219 (2015), 25–39.
- [29] HUANG, C.-L., YEH, P.-H., LIN, C.-W., AND WU, D.-C. Utilizing user tag-based interests in recommender systems for social resource sharing websites. *Knowledge-Based Systems* 56 (2014), 86–96.
- [30] IGNATOV, D. I., NIKOLENKO, S. I., ABAEV, T., AND POELMANS, J. Online recommender system for radio station hosting based on information fusion and adaptive tag-aware profiling. *Expert Systems with Applications* 55 (2016), 546–558.
- [31] JÄSCHKE, R., MARINHO, L., HOTHO, A., SCHMIDT-THIEME, L., AND STUMME, G. Tag recommendations in social bookmarking systems. *Ai Communications* 21, 4 (2008), 231–247.
- [32] KHAN, M. M., IBRAHIM, R., AND GHANI, I. Cross domain recommender systems: a systematic literature review. *ACM Computing Surveys (CSUR)* 50, 3 (2017), 36.

- [33] KIM, H., AND KIM, H.-J. A framework for tag-aware recommender systems. *Expert Systems with Applications* 41, 8 (2014), 4000–4009.
- [34] KIM, H.-N., ALKHALDI, A., EL SADDIK, A., AND JO, G.-S. Collaborative user modeling with user-generated tags for social recommender systems. *Expert Systems with Applications* 38, 7 (2011), 8488–8496.
- [35] KIM, S.-M., AND HOVY, E. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics* (2004), Association for Computational Linguistics, p. 1367.
- [36] KLANJA-MILICEVIC, A., VESIN, B., IVANOVIC, M., AND BUDIMAC, Z. Personalisation of programming tutoring system using tag-based recommender systems. In *Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on* (2012), IEEE, pp. 666–667.
- [37] ŁADYŻYŃSKI, P. P., AND GRZEGORZEWSKI, P. A recommender system based on customer reviews mining. In *International Conference on Artificial Intelligence and Soft Computing* (2014), Springer, pp. 512–523.
- [38] LAMERE, P. Social tagging and music information retrieval. *Journal of new music research* 37, 2 (2008), 101–114.
- [39] LE, Q. V., NGIAM, J., COATES, A., LAHIRI, A., PROCHNOW, B., AND NG, A. Y. On optimization methods for deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning* (2011), Omnipress, pp. 265–272.
- [40] LEE, H., BATTLE, A., RAINA, R., AND NG, A. Y. Efficient sparse coding algorithms. In *Advances in neural information processing systems* (2007), pp. 801–808.
- [41] LIKA, B., KOLOMVATSOS, K., AND HADJIEFTHYMIADES, S. Facing the cold start problem in recommender systems. *Expert Systems with Applications* 41, 4 (2014), 2065–2073.
- [42] LING, G., LYU, M. R., AND KING, I. Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM Conference on Recommender systems* (2014), ACM, pp. 105–112.
- [43] LIPCZAK, M., AND MILIOS, E. Efficient tag recommendation for real-life data. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 1 (2011), 2.
- [44] LIU, H., HE, J., WANG, T., SONG, W., AND DU, X. Combining user preferences and user opinions for accurate recommendation. *Electronic Commerce Research and Applications* 12, 1 (2013), 14–23.



- [45] LU, J., WU, D., MAO, M., WANG, W., AND ZHANG, G. Recommender system application developments: a survey. *Decision Support Systems* 74 (2015), 12–32.
- [46] MA, H., ZHOU, D., LIU, C., LYU, M. R., AND KING, I. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining* (2011), ACM, pp. 287–296.
- [47] MAHBOOB, V. A., JALALI, M., JAHAN, M. V., AND BAREKATI, P. Swallow: resource and tag recommender system based on heat diffusion algorithm in social annotation systems. *Computational Intelligence* 33, 1 (2017), 99–118.
- [48] MALLINSON, B., AND SEWRY, D. Elearning at rhodes university-a case study. In *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on* (2004), IEEE, pp. 708–710.
- [49] MARLOW, C., NAAMAN, M., BOYD, D., AND DAVIS, M. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *Proceedings of the seventeenth conference on Hypertext and hypermedia* (2006), ACM, pp. 31–40.
- [50] MASSIMO, D., ELAHI, M., GE, M., AND RICCI, F. Item contents good, user tags better: Empirical evaluation of a food recommender system. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (2017), ACM, pp. 373–374.
- [51] MCAULEY, J., AND LESKOVEC, J. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems* (2013), ACM, pp. 165–172.
- [52] MEDWORM. <https://medworm.com/>, 2017.
- [53] MEMMEL, M., KOCKLER, M., AND SCHIRRU, R. Providing multi source tag recommendations in a social resource sharing platform. *J. UCS* 15, 3 (2009), 678–691.
- [54] MISAGHIAN, N., JALALI, M., AND MOATTAR, M. H. Resource recommender system based on tag and time for social tagging system. In *Computer and Knowledge Engineering (ICCKE), 2013 3th International eConference on* (2013), IEEE, pp. 97–101.
- [55] MNIH, A., AND SALAKHUTDINOV, R. R. Probabilistic matrix factorization. In *Advances in neural information processing systems* (2008), pp. 1257–1264.
- [56] MOVAHEDIAN, H., AND KHAYYAMBASHI, M. R. A semantic recommender system based on frequent tag pattern. *Intelligent Data Analysis* 19, 1 (2015), 109–126.

- [57] OKU, K., AND HATTORI, F. User evaluation of fusion-based approach for serendipity-oriented recommender system. In *Proceedings of the Workshop on Recommendation Utility Evaluation: Beyond RMSE (RUE 2012)* (2012), pp. 39–44.
- [58] PORTUGAL, I., ALENCAR, P., AND COWAN, D. The use of machine learning algorithms in recommender systems: a systematic review. *Expert Systems with Applications* (2017).
- [59] QIU, L., GAO, S., CHENG, W., AND GUO, J. Aspect-based latent factor model by integrating ratings and reviews for recommender system. *Knowledge-Based Systems 110* (2016), 233–243.
- [60] RICCI, F., ROKACH, L., AND SHAPIRA, B. Introduction to recommender systems handbook. In *Recommender systems handbook*. Springer, 2011, pp. 1–35.
- [61] SAHU, A. K., DWIVEDI, P., AND KANT, V. Tags and item features as a bridge for cross-domain recommender systems. *Procedia Computer Science 125* (2018), 624 – 631. The 6th International Conference on Smart Computing and Communications.
- [62] SARI, P. K., ALAMSYAH, A., AND WIBOWO, S. Measuring e-commerce service quality from online customer review using sentiment analysis. In *Journal of Physics: Conference Series* (2018), vol. 971, IOP Publishing, p. 012053.
- [63] SEN, S., VIG, J., AND RIEDL, J. Tagommenders: connecting users to items through tags. In *Proceedings of the 18th international conference on World wide web* (2009), ACM, pp. 671–680.
- [64] SHEPITSEN, A., GEMMELL, J., MOBASHER, B., AND BURKE, R. Personalized recommendation in social tagging systems using hierarchical clustering. In *Proceedings of the 2008 ACM conference on Recommender systems* (2008), ACM, pp. 259–266.
- [65] SHI, Y., LARSON, M., AND HANJALIC, A. Mining contextual movie similarity with matrix factorization for context-aware recommendation. *ACM Transactions on Intelligent Systems and Technology (TIST) 4*, 1 (2013), 16.
- [66] SHI, Y., LARSON, M., AND HANJALIC, A. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR) 47*, 1 (2014), 3.
- [67] SMITH, G. Folksonomy: social classification. *Blog article, August* (2004).

- [68] SRIFI, M., HAMMOU, B. A., MOULINE, S., AND LAHCEN, A. A. Collaborative recommender systems based on user-generated reviews: A concise survey. In *2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)* (2018), IEEE, pp. 1–6.
- [69] SUN, C.-Y., AND LEE, A. J. Tour recommendations by mining photo sharing social media. *Decision Support Systems* 101 (2017), 28–39.
- [70] SUN, Q., NIU, J., YAO, Z., AND YAN, H. Exploring ewom in online customer reviews: Sentiment analysis at a fine-grained level. *Engineering Applications of Artificial Intelligence* 81 (2019), 68–78.
- [71] SUSAN, M. M., AND DAVID, S. What makes a helpful online review? a study of customer reviews on amazon. com. *MIS Quarterly* 34, 1 (2010), 185–200.
- [72] TAN, Y., ZHANG, M., LIU, Y., AND MA, S. Rating-boosted latent topics: Understanding users and items with ratings and reviews. In *IJCAI* (2016), vol. 16, pp. 2640–2646.
- [73] TANG, T. Y., AND MCCALLA, G. Smart recommendation for an evolving e-learning system: Architecture and experiment. *International Journal on elearning* 4, 1 (2005), 105.
- [74] VAN DEN OORD, A., DIELEMAN, S., AND SCHRAUWEN, B. Deep content-based music recommendation. In *Advances in neural information processing systems* (2013), pp. 2643–2651.
- [75] VAN SETTEN, M., BRUSSEE, R., VAN VLIET, H., GAZENDAM, L., VAN HOUTEN, Y., AND VEENSTRA, M. On the importance of âwho tagged whatâ. In *Proceedings of the Workshop on the Social Navigation and Community based Adaptation Technologies at AH* (2006), vol. 6.
- [76] WANG, C., AND BLEI, D. M. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (2011), ACM, pp. 448–456.
- [77] WANG, X., HE, X., NIE, L., AND CHUA, T.-S. Item silk road: Recommending items from information domains to social users. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval* (2017), ACM, pp. 185–194.
- [78] XU, G., GU, Y., ZHANG, Y., YANG, Z., AND KITSUREGAWA, M. Toast: a topic-oriented tag-based recommender system. In *International Conference on Web Information Systems Engineering* (2011), Springer, pp. 158–171.

- [79] YAMABA, H., TANOUÉ, M., TAKATSUKA, K., OKAZAKI, N., AND TOMITA, S. On a serendipity-oriented recommender system based on folksonomy. *Artificial Life and Robotics* 18, 1-2 (2013), 89–94.
- [80] YOON, H., ZHENG, Y., XIE, X., AND WOO, W. Social itinerary recommendation from user-generated digital trails. *Personal and Ubiquitous Computing* 16, 5 (2012), 469–484.
- [81] YUAN, X., HUANG, J.-J., AND ZHONG, N. Preference structure and similarity measure in tag-based recommender systems. In *International Conference on Active Media Technology* (2013), Springer, pp. 193–202.
- [82] ZHANG, H., ZHA, Z.-J., YANG, Y., YAN, S., GAO, Y., AND CHUA, T.-S. Attribute-augmented semantic hierarchy: Towards a unified framework for content-based image retrieval. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 11, 1s (2014), 21.
- [83] ZHANG, L., ZHOU, Y., DUAN, X., AND CHEN, R. A hierarchical multi-input and output bi-gru model for sentiment analysis on customer reviews. In *IOP Conference Series: Materials Science and Engineering* (2018), vol. 322, IOP Publishing, p. 062007.
- [84] ZHANG, Y., HU, J., SANO, S., YAMASAKI, T., AND AIZAWA, K. A tag recommendation system for popularity boosting. In *Proceedings of the 2017 ACM on Multimedia Conference* (2017), ACM, pp. 1227–1228.
- [85] ZHANG, Y., HU, J., SANO, S., YAMASAKI, T., AND AIZAWA, K. Computer vision based and fprank based tag recommendation for social popularity enhancement. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion* (2018), ACM, p. 3.
- [86] ZHANG, Z., ZENG, D. D., ABBASI, A., PENG, J., AND ZHENG, X. A random walk model for item recommendation in social tagging systems. *ACM Transactions on Management Information Systems (TMIS)* 4, 2 (2013), 8.
- [87] ZHANG, Z.-K., ZHOU, T., AND ZHANG, Y.-C. Tag-aware recommender systems: a state-of-the-art survey. *Journal of computer science and technology* 26, 5 (2011), 767.
- [88] ZHENG, N., AND LI, Q. A recommender system based on tag and time information for social tagging systems. *Expert Systems with Applications* 38, 4 (2011), 4575–4587.
- [89] ZUO, Y., ZENG, J., GONG, M., AND JIAO, L. Tag-aware recommender systems based on deep neural networks. *Neurocomputing* 204 (2016), 51–60.

