Chapter 1

Introduction

Modern applications on the web are continually dealing with a large amount of data. It is a challenging task to filter this large volumes of data. In an era, where the number of choices is overwhelming on the web, it is crucial to filter, prioritize and deliver relevant information to a user. In this scenario, information filtering systems are pivotal in making content/item search more accessible and efficient to provide personalised recommendations. Recommender systems are information filtering systems that provide items/contents/services from a plethora of choices available on the web by mining users' past preferences and feedback. Recommendation helps to overcome the information overload problem and as there are many options (for a specific requirement) and more will be added at regular intervals.

"A recommender system is a popular application that facilitate in matching users to their unanticipated contents [101]."

Recommender systems collect users' past preferences and generate an appropriate recommendation of items to satisfy the users. Nowadays, the prevalence of providing personalized content to users has increased profoundly.

"A recommender system is an extensive class of a web application that makes it possible to predict user choices from the available options [72]."

Recommender systems have become an essential part of social network sites like Facebook, Twitter, LinkedIn, etc. e-commerce sites like Amazon, Walmart, Alibaba, etc., music and video streaming platforms like Spotify, Pandora, Netflix, etc. and many other popular services on the web. Recommender systems aim to satisfy the users of these platforms by helping them to discover content that matches their interests. The user

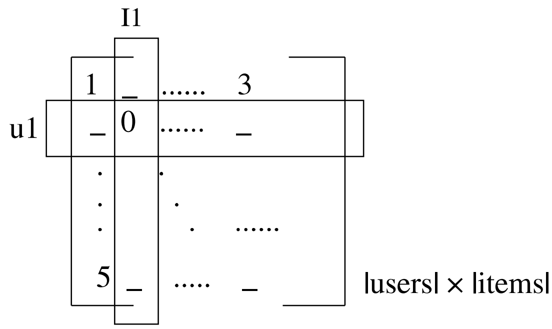

Figure 1.1: Utility Matrix

preference information can be acquired either by explicit or implicit ratings.

Recommendation model can be classified into two categories, viz., memory-based collaborative filtering and model-based collaborative filtering. Memory-based collaborative filtering considers a utility matrix, and similar users or items are determined using this utility matrix. The technique takes a utility matrix (ratings given by the users for items), and by using this interaction matrix, similar users or items are determined. Figure 1.1 depicts a utility matrix. In the given sample of a utility matrix, rows represent the numbers of users and columns signify the number of items. In the row, U1 is a user, and his corresponding preferences are given. In the column, item I1 indicates ratings provided by all the users._ represents missing ratings in the utility matrix.

This model calculates the similarity between users or items, finds the nearest neighbors, and predicts the missing rating in the utility matrix. This method ranks all the candidate items of a user based on the ratings given by similar users to that item. A model has representation learning of users-items interaction in order to recommend items in the model-based collaborative filtering technique.

A person is not an isolated entity; often, he performs activities in a group like dining in a restaurant with colleagues, watching a movie with her spouse, planning a tour itinerary with friends, etc. This kind of social activity, with the scope for group decision making, led to the development of group recommender systems.

"In group recommendations, the involvement of more than one user is required in the recommendation process [56]."

The goal of a traditional recommender system is to suggest the most relevant items to a user. A classical recommender system is mainly concerned about the satisfaction of a user. In contrast, the group recommendation aims to generate a recommendation vector for a group of users with diversified interests [1, 2, 21, 56, 117]. Mobile devices and social networking increased the importance of group recommendations in various domains. Group recommender systems are useful in many areas, viz., movies, tourism, music, TV programs, web browsing, etc. [32,34,35,74,78,79,111]. Incorporating social and behavioural aspects in the group recommendations is an active field of research [95]. The factors influencing the group recommender systems are social background, social relations, trust, and interest similarities. These factors can be directly or indirectly extracted from the social network structure [65].

Group recommendation can be classified into two categories based on aggregation:

Recommendation Aggregation [86]: The recommendation aggregation develops item recommendations for each group member individually. A preference aggregation strategy (i.e., least misery [1, 77] and aggregated voting [1, 77]) is employed to produce a single recommendation vector that is suitable for the whole group.

Profile Aggregation [61, 74, 130]: The profile aggregation approach integrates each member's observed preferences into a group profile (sometimes known as virtual user). The traditional recommendation techniques recommend items to the virtual user which in turn become the results for the corresponding group. These two approaches are memory-based.

Group Formation: Groups can be mainly of three types based on the interactions among the members of the group [14].

1. **Established group:** A group of people who explicitly choose to be a part of the group based on the shared common interests [63]. Persistent/Established groups refer to relatively static groups with stable members and sufficient group-item interaction records, such as interest-oriented groups in Meetup[1], communities having similar behavior [86], etc.

2. **Occasional group:** This class contains a group of people performing activities occasionally. Users might be together for the first time. All the group members will have a common aim at a particular moment. Example: Recommending songs in the fitness center [79], recreation at swimming pool, planning for a vacation [40],etc.

3. **Random group:** A class of users' group where people come together in an environ-

[1] https://www.meetup.com/

ment by some chance. A group of people who share an environment at a particular time without any link between them. Example: passengers travelling in public transport [32], people shopping in a mall, recommending news items publicly, i.e., GAIN (Group Adapted Interaction for News) [92], people travelling together [3], visiting a POI, etc..

4. **Automatically identified group**: A group that is automatically detected for the available resources by considering user preferences, viz., users give similar ratings to the same items, come under the same cluster [16].

In [69], the authors employ the Louvain algorithm, particle swarm optimization based K-means and Gaussian Mixture Model (GMM) techniques to auto-detect a group. The proposed model uses memory-based approaches during recommendation.

In this thesis, we proposed efficient models to auto-detect the groups. Our primary concern is retrieving automatically identified groups [16, 17, 125, 126] and generating a group recommendation vector. We consider both order preference and flexible-size preference models. We identify groups automatically by using probabilistic hashing techniques, viz., MinHash, One permutation hashing . In [70], we introduce a model to identify a potential user group while dealing with the curse of dimensionality using locality-sensitive hashing (a nearest-neighbor search strategy). In the case of MinHash, the obtained automatically identified groups are promising after amplifying locality-sensitive hashing. This model applies nearest-neighbor search techniques on the generated signature matrix to auto-detect the groups. This method is promising and results are also encouraging. This model mitigates the data sparsity problem to a great extent.

In [68], we blend one permutation hashing technique and an autoencoder model to auto-detect the groups. We also compare the proposed technique with other established clustering approaches to probe the effectiveness of the proposed model. The extensive experiments on the MovieLens and Amazon-music (Digital Music) datasets show the efficacy of the proposed models to a great extent.

Auto-detecting groups from available information, i.e., datasets and metadata, is a challenging task. The objective is to form a group where all the members share common characteristics. In many cases, the group members who share similar characteristics are unfamiliar with each other. This key aspect inspires us to explore a group before generating the recommendation as the group composition is vital in an automatically identified group [16, 17, 126] in group recommendations. Existing models identify and detect the group automatically by using classical clustering approaches like K-means clustering [16, 17, 126], Agglomerative clustering [20] and Louvain clustering [12]. In this group, generating recommendations deals with the similar items and increases the overall group recommendation score.

When we build group recommender systems, a common list of interested items is taken as positive feedback, and plays a major role in the recommendation process. Since users do not consider some uninterested items during the recommendation process. These uninterested items may also play an important role in influencing the user's positive feedback behaviors, which can provide us a better interpretation of users' preferences [39]. This type of recommendation aggregates users' preferences and infers the group's decision as a whole.

1.1 Research Gap and Research Goal

There are many effective models in the literature for auto-detecting the groups in group recommendations. These models use traditional clustering approaches to auto-detect groups. However, some issues and research gaps are still challenging and need to be addressed. These issues and research gaps form the motivation of this thesis. The problems identified in research gaps are as follows:

- Scalability and data sparsity problem not addressed effectively in group recommendation.
- Curse of dimensionality is not addressed effectively for group recommendation.
- Meta-data information is not exploited effectively in forming groups in group recommendations.

To bridge these gaps, we introduce multiple efficient models to identify groups in group recommendations. The following are the research goals of our research:

1. Design and develop scalable models that can work effectively on both sparse and voluminous data for group formation.

2. Study the role of uninterested items in group recommendations.

3. Our models should take advantage of meta-data in forming groups.

1.2 Motivation

Group formation (or how to place users in appropriate groups?) is the fundamental problem in group recommender systems. The majority of previous works have developed group recommender systems with prior knowledge about group members. In many real-world scenarios, this information is generally not available. Existing literature shows that group formation is based on traditional clustering approaches. The existing models automatically identify and detect the groups by computing correlations among all individual users [1] or by applying traditional clustering approaches over a utility matrix. The former approach has disadvantages due to time complexity. In contrast, the time complexity is less in the later approach, and this approach is promising. Exploring a particular group from the available information is a challenging task. In many cases, the group members who share similar characteristics need to be more familiar with each other. This motivated us to explore a group before producing the recommendation, as a group composition is essential in an automatically identified group [16, 17, 126] in group recommendations. None of the systems deals with the curse of dimensionality to form groups.

The motivation of this thesis is to suitably employ the probabilistic hashing techniques to auto-detect the groups in group recommendations. It is necessary to explore user communities that share similar characteristics before utilizing group recommender systems. MinHash is a locality-sensitive hashing [4] technique which provides a hash-based solution in a large space. MinHash is a probabilistic data structure used in the nearest neighbor search problem efficiently [48]. However, there are some issues with the Minhash. It requires more storage, and computational costs are also expensive during similarity searches. Also, it uses many different hash functions for a reasonable estimation. So, these challenges motivated us to explore an alternative solution to reduce this computation. One permutation hashing is a promising method to address this shortcoming. We need to compute only one hash function in this technique. This approach is computationally very efficient and also useful in dimensionality reduction. Of late, autoencoder [57, 60, 73, 87, 121] gained their importance in recommender systems. An autoencoder is an unsupervised deep learning model which became popular in the last decade [73, 88, 121, 132]. An autoencoder reconstructs the input values at the output layer. This model is also helpful in feature extraction and dimensionality reduction [93]. These commendable properties motivated us to use them (the proposed model is named as OPHAencoder) for users' clustering. The challenge is to find the optimal group of users with similar preferences to maximize users' satisfaction.

1.3 Contributions

The contributions made in this thesis are as follows:

- To the best of our knowledge, this is the first attempt to introduce a locality-sensitive hashing technique to detect and compose groups automatically in group recommender systems.
- This thesis studies the usefulness of classical clustering and community detection-based approaches to compare the effectiveness of automatically detected groups. The proposed model studies it by considering the order and flexible preferences in group recommendations.
- To the best of our knowledge, this is the first attempt to incorporate the *one permutation hashing* method and the *autoencoder* model to auto-detect groups in a group recommender system. This thesis studies the proposed model by considering both order and flexibility in user preferences for group recommendations.
- This thesis studies the rating prediction task of the automatically detected group based on the neural collaborative filtering [49] model.
- This thesis introduces models by incorporating the importance of uninterested items captured using user preferences.
- This thesis studies the effectiveness of proposed models based on a randomly formed group (random group) and an "automatically identified group" [15, 16].
- We propose model that adopts particle swarm optimization [134], Louvain algorithm [13] and Gaussian mixture model [106] techniques to detect and compose "automatically identified groups".

- To the best of our knowledge, this is the first attempt to reuse pre-trained language models to auto-detect the groups in group recommendations.
- This thesis analysis the automatically identified groups using a cluster validation approach. It studies the effectiveness of the proposed models based on the order in user preferences. In this thesis, we conduct a comparative study with the baseline and the state-of-the-art model.

1.4 Thesis outline

Throughout this research work, various research issues regarding auto-detecting groups using traditional clustering approaches are identified. Several efficient models are proposed to address the problems, and research papers are written. This thesis is arranged into seven different chapters. The core work of this thesis (derived from our own research articles) is presented in Chapter 3, Chapter 4, Chapter 5, and Chapter 6 for gaining better insights. Finally Chapter 7 concludes the thesis. Figure 1.2 depicts the structure of this thesis. A brief description of the chapters is as follows:

Chapter 1: This chapter provides the motivations behind the thesis by explaining some research issues. This chapter briefly describes the information overload problem, recommender systems, the need for group recommendations and the importance of auto-detecting groups. Chapter 1 covers the introduction of our thesis. It discusses the motivation for the thesis, research gap, research goal and summarizes our contributions.

Chapter 2: In this chapter, we discuss preliminaries that are important to understand the concepts associated with group recommender systems. We also discuss related works on group recommendations and identify several research gaps in the existing models for auto-detecting groups.

Chapter 3: This chapter describes auto-detecting groups using locality-sensitive hashing technique in group recommendations.

Chapter 4: This chapter introduces a model to auto-detect groups in group recommendations by considering one permutation hashing method and the autoencoder model.

Chapter 5: This chapter presents a model to auto-detect groups by considering metadata information.

Chapter 6: This chapter presents a study to determine the role of uninterested items in

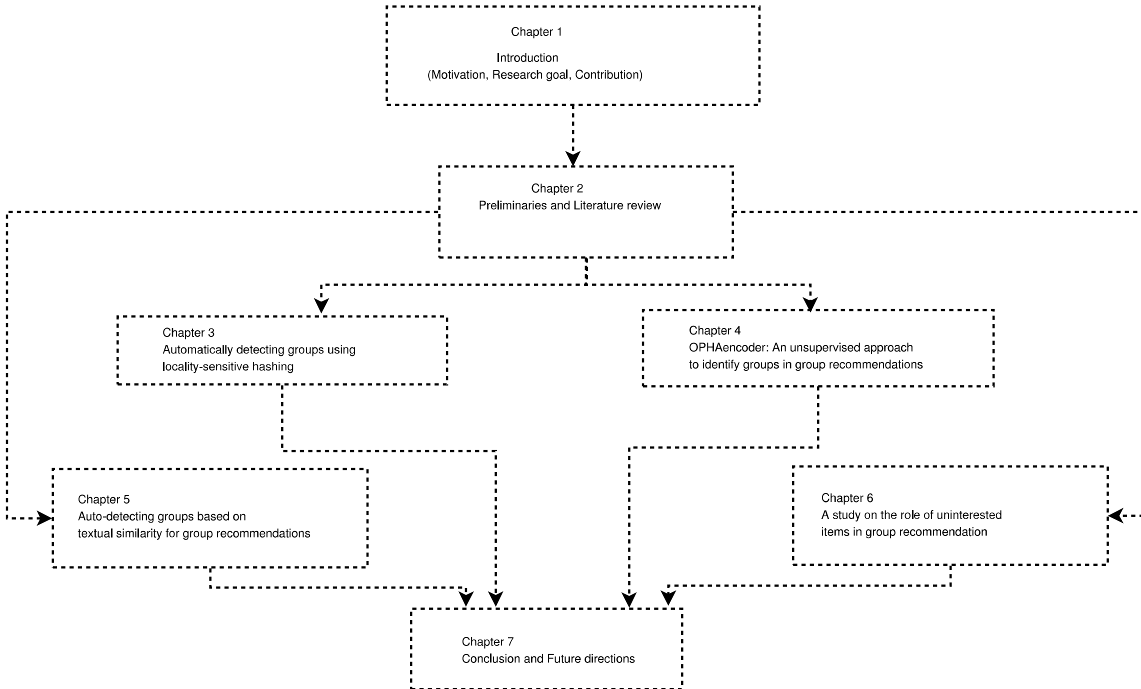

Figure 1.2: Thesis structure

group recommendations when considering a random group and an automatically identified group.

Chapter 7: This chapter concludes and discusses future work.