Fairness in Group Recommendations in the Health Domain

Maria Stratigi

Thesis submitted in partial fulfillment of the requirements for the

Masters' of Science degree in Computer Science and Engineering

University of Crete
School of Sciences and Engineering
Computer Science Department
Voutes University Campus, 700 13 Heraklion, Crete, Greece

Thesis Advisors: Prof. *Dimitris Plexousakis*

Assoc. Prof. *Kostas Stefanidis*

Dr. *Haridimos Kondylakis*

This work has been performed at the University of Crete, School of Sciences and Engineering, Computer Science Department.

The work has been supported by the Foundation for Research and Technology - Hellas (FORTH), Institute of Computer Science (ICS).

University of Crete
Computer Science Department

Fairness in Group Recommendations in the Health Domain

Thesis submitted by
Maria Stratigi
in partial fulfillment of the requirements for the
Masters' of Science degree in Computer Science

THESIS APPROVAL

Author: ______________________
Maria Stratigi

Committee approvals: ______________________
Dimitris Plexousakis
Professor, University of Crete, Thesis Supervisor

____
Yannis Tzitzikas
Associate Professor, University of Crete, Committee Member

____
Irini Fundulaki
Researcher, FORTH - ICS, Committee Member

____
Kostas Stefanidis
Associate Professor, University of Tampere, Committee Member

Departmental approval: ______________________
Antonios Argyros
Professor, Director of Graduate Studies

Heraklion, Octomber 2017

Fairness in Group Recommendations in the Health Domain

Abstract

During the last decade, the number of users who look for health-related information has impressively increased. On the other hand, health professionals have less and less time to recommend useful sources of such information online to their patients. To this direction, we target at streamlining the process of providing useful online information to patients by their caregivers and improving as such the opportunities that patients have to inform themselves online about diseases and possible treatments. Using our system, relevant and high quality information is delivered to patients based on their profile, as represented in their personal healthcare record data, facilitating an easy interaction by minimizing the necessary manual effort.

Specifically, in this work, we propose a model for group recommendations incorporating fairness, following the collaborative filtering approach. As in collaborative filtering, it is crucial to identify the correct set of similar users, in addition to traditional methods, we pay particular attention on how to exploit user's health information. To this direction, we define a novel similarity measure that is based on the semantic distance between users' health problems. Our special focus is on providing valuable suggestions to a caregiver who is responsible for a group of users. We interpret valuable suggestions as ones that are both highly related and fair to the users of the group. As such, we introduce in addition, a new aggregation method incorporating fairness and we compare it with current state of the art. Our experiments demonstrate the advantages of both the semantic similarity method and the fair aggregation design. To the best of our knowledge, this is the first work that introduces the concept of group recommendations incorporating fairness in the health domain.

Δικαιοσύνη στις Συστάσεις σε Ομάδες στον Τομέα της Υγείας

Περίληψη

Στην τελευταία δεκαετία ο αριθμός των χρηστών, οι οποίοι αναζητούν πληροφορίες σχετικά με την υγεία, έχει αυξηθεί δραματικά. Από την άλλη πλευρά, οι επαγγελματίες του τομέα της υγείας, έχουν όλο και λιγότερο χρόνο να συστήσουν στους ασθενείς τους χρήσιμες πηγές τέτοιων πληροφοριών στο διαδίκτυο. Προς αυτήν την κατεύθυνση, στοχεύουμε στην απλοποίηση της διαδικασίας παροχής χρήσιμων ηλεκτρονικών πληροφοριών προς τα άτομα υπεύθυνα για ασθενείς και στην βελτίωση ως εκ τούτου των ευκαιριών που έχουν οι ασθενείς να ενημερώνονται για ασθένειες και πιθανές θεραπείες. Χρησιμοποιώντας το σύστημα μας, οι σχετικές και υψηλής ποιότητας πληροφορίες παραδίδονται στους ασθενείς με βάση το προφίλ τους, όπως αυτό αναπαρίστανται στο προσωπικά τους αρχεία δεδομένων υγειονομικής περίθαλψης, διευκολύνοντας την αλληλεπίδραση, με την ελαχιστοποίηση της απαραίτητης χειρωνακτικής προσπάθειας.

Συγκεκριμένα, σε αυτήν την εργασία προτείνουμε ένα πρότυπο για συστάσεις σε μια ομάδα σύμφωνα με την προσέγγιση του συνεργατικού φλιταρίσματος που ενσωματώνει την έννοια της δικαιοσύνης. Δεδομένου ότι το συνεργατικό φιλτράρισμα είναι κρίσιμο για τον προσδιορισμό ενός σωστού συνόλου παρόμοιων χρηστών για έναν εν λόγω χρήστη, εκτός από τις παραδοσιακές αξιολογήσεις δίνουμε ιδιαίτερη προσοχή στον τρόπο αξιοποίησης των πληροφοριών που σχετίζονται με την υγεία, για τον υπολογισμό των ομοιοτήτων μεταξύ των χρηστών. Για να το καταφέρουμε αυτό, ορίζουμε ένα νέο μέτρο ομοιότητας που βασίζετε στην σημασιολογική απόσταση μεταξύ των προβλημάτων υγείας. Δίνουμε έμφαση στην παροχή πολύτιμων προτάσεων σε ένα άτομο υπεύθυνο για μια ομάδα χρηστών. Ερμηνεύουμε χρήσιμες προτάσεις, ως τις προτάσεις οι οποίες είναι τόσο σχετικές όσο και δίκαιες με τους χρήστες της ομάδας. Προς αυτήν την κατεύθυνση, εισάγουμε μια νέα μέθοδο συγκέντρωσης που ενσωματώνει την αμεροληψία και την συγκρίνουμε με σύγχρονες τεχνολογίες. Τα πειράματα μας καταδεικνύουν τα πλεονεκτήματα τόσο της μεθόδου σημασιολογικής ομοιότητας όσο και της δίκαιας μεθόδου συσσωμάτωσης. Από όσο γνωρίζουμε, αυτό είναι το πρώτο έργο που εισάγει την έννοια των συστάσεων σε ομάδα και της δικαιοσύνης στο τομέα της υγείας.

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω θερμά τους κ. Χαρίδημο Κονδυλάκη και Κώστα Στεφανίδη, για την άψογη συνεργασία, ορθή καθοδήγηση και ουσιαστική συμβολή τους στην ολοκλήρωση της παρούσας μεταπτυχιακής εργασίας. Επίσης θέλω να εκφράσω τις ευχαριστίες μου στον κ. Δημήτρη Πλεξουσάκη που ανέλαβε τα καθήκοντα του επόπτη της εργασίας μου και για την συμμετοχή του στην επιτροπή. Τέλος ευχαριστίες στους κ. Γιάννη Τζίτζικα και κ. Ειρήνη Φουντουλάκη για την προθυμία τους να συμμετέχουν στην τριμελή επιτροπή.

Ακόμα να ευχαριστήσω το Ινστιτούτο Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας για την πολύτιμη υποστήριξη σε υλικοτεχνική υποδομή και τεχνογνωσία και για την υποτροφία που μου πρόσφερε.

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τους γονείς μου για την αγάπη και στήριξη τους καθ΄ όλη την διάρκεια των σπουδών μου καθώς και στον αδελφό μου για την υπομονή που έδειξε μαζί μου και για τα γέλια που μου πρόσφερε όταν ήμουν στα κάτω μου. Ακόμα, θέλω να ευχαριστήσω τους φίλους μου για την υποστήριξη τους όλα αυτά τα χρόνια. Τέλος, η εργασία αυτή είναι αφιερωμένη στην μητέρα μου, που αποτελεί τον πιο θερμό υποστηρικτή μου, και χωρίς αυτήν δεν θα είχα καταφέρει να φτάσω σε αυτό το σημείο.

στους γονείς μου

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Medicine is undergoing a revolution that is transforming the nature of health-care from reactive to preventive. The changes are catalyzed by a new systems approach to disease which focuses on integrated diagnosis, treatment and prevention of disease in individuals. This will replace our current mode of medicine over the coming years with a personalized predictive treatment. While the goal is clear, the path is fraught with challenges. One of these challenges is the amount of information that can be found online [10] since health information is one of the most frequently searched topics on the Web.

During the last decade, the number of users who look for health and medical information has dramatically increased. Already from 2002, a percentage of 80% of all adults in the United States were estimated to have looked online for health information, whereas in 2006, the 23% of the Europeans were using the Internet to be informed about their health needs [32]. However despite the increase in those numbers and the vast amount of information currently available online, it is very hard for a patient to accurately judge the relevance of some information to his own case and to identify the quality of the provided information.

Furthermore, according to recent research [10] the optimal solution for patients is to be guided by healthcare providers to more optimal resources over the web. Delivering accurate sources to a patient, increases his/her knowledge and changes the way of thinking which is usually referred as patient empowerment. As a result, the patient's dependency for information from the doctor is reduced. Moreover, patients feel autonomous and more confident about the management of their disease [55]. To this direction, health providers have the history of their patient's and their interests in either paper or mental records, in order to make an informed decision about the information that would likely be beneficial for them. However, health providers have less and less time to devote to their patients. As such, guiding each individual patient appropriately is a really difficult task.

On the other hand, the use of group-dynamics-based principles of behavior change have been shown to be highly effective in enhancing social support through promoting group cohesion in physical activity [22], in reducing smoking relapse [13] and in promoting healthy dietary habits [33]. Especially for cancer, latest studies [9] suggest that group therapy improves the well-being of cancer patients because of enhanced discussion and social support. In small groups, therapy sessions enjoy a social component as participants can share experiences and discussion. In those therapy sessions, a caregiver guides patients to more optimal resource over the web. However, if identifying online information content for a single patient is a difficult task, identifying information for a group of participants is a really challenging one.

To this direction, in this work we focus on recommending interesting health documents, to groups of users incorporating the notion of fairness, using a collaborative filtering approach.

Our motivation for this work, is to offer a list of recommendations to a caregiver that is responsible for a group of patients. The recommended documents need to be relevant, based on the patients current profile. To exploit patient's profile we exploit their data as stored in their personal healthcare record (PHR) data. These patient do not necessarily suffer from the same health problems but a variety of them. As such, we introduce the notion of fairness in the recommendation process. Our argument is that if the group recommendation list provides a high relevant document for a patient, then that patient may be tolerant of the existence of documents that are not relevant to him/her.

More specifically the contributions of our work are the following:

- We demonstrate the first group recommendation model incorporating fairness in the health domain.
- We propose a novel semantic similarity function that takes into account the patient's medical profile showing its superiority over a traditional measure.
- We introduce a new aggregation method that encapsulates the notion of fairness, called *Fair*.
- We explore four different aggregation methods, *Minimum*, *Average*, *Borda* and *Fair* demonstrating that the Borda and Fair designs have better quality than the rest.
- We demonstrate the first synthetic dataset for benchmarking works in the area.

To the best of our knowledge, this is the first work that introduces the concept of group recommendations and fairness in the health domain. We have to note that a preliminary version of our approach was already successfully presented in ICDE 2017 [53] whereas the requirements for generating such a recommendation tool originally came from the iManageCancer EU research project[1]. The project, still

[1]http://imanagecancer.eu/

ongoing, has the objective to provide a cancer specific self-management platform focusing on empowering patients and their caregivers, and to provide the patients with personalized, context-sensitive, data driven information services in a language they understand and help them to make informed choices on treatment options in collaboration with their health carers.

1.2 Outline

The rest of this thesis is structured as follows: In Chapter 2, we present related work. In Chapter 3, we focus on single user recommendations, identifying user similarities and the single user rating model. Then, in Chapter 4, we focus on groups incorporating fairness in our recommendations. Chapter 5 demonstrates our experimental evaluation, and finally, Chapter 6, concludes this thesis and provides directions for future work.

Chapter 2

Related Work

2.1 Recommender Systems

The research literature on recommendations is extensive. Typically, recommendation approaches are distinguished between: content-based, that recommend items similar to those the user previously preferred (e.g.,[41, 34]), collaborative filtering, that recommend items that users with similar preferences like (e.g., [29, 11]) and hybrid, that combine content-based and collaborative ones (e.g., [7]).

Content-based filtering, also referred to as cognitive filtering, recommends items based on a comparison between the content of the items and a user profile [12, 4]. The content of each item is represented as a set of descriptors or terms, typically the words that occur in a document. The user profile is represented with the same terms and built up by analyzing the content of items which have been seen by the user. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past. In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. A key issue with content-based filtering is whether the system is able to learn user preferences from users' actions regarding one content source and use them across other content types. For example, recommending news articles based on browsing of news is useful, but would be much more useful when music, videos, products, discussions etc. from different services can be recommended based on news browsing [36].

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, activities or preferences and predicting what users will like based on their similarity to other users. It is based on the idea that people who agreed in their evaluation of certain items in the past are likely to agree again in the future. A person who wants to see a movie for example, might ask for recommendations from friends. The recommendations of some friends who have similar interests are trusted more than recommendations from others. This information is used in the decision on which movie to see. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. In contrast collaborative filtering algorithms suffer from three main problems: cold start, scalability, and sparsity. These systems often require a large amount of existing data on a user in order to make accurate recommendations, which results in the cold start problems [46, 17]. Furthermore in many of the environments in which these systems make recommendations, there are millions of users and products. Thus, a large amount of computation power is often necessary to calculate recommendations. Finally in most cases the number of items is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings.

In the hybrid approach, collaborative filtering and content-based filtering are combined. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model. Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem [19]. Netflix is a good example of the use of hybrid recommender systems. The website makes recommendations by comparing the watching and searching habits of similar users (i.e., collaborative filtering) as well as by offering movies that share characteristics with films that a user has rated highly (content-based filtering).

Recently, there are also approaches focusing on enhancing recommendations with further contextual information (e.g., [1, 39]). In these approaches, context is defined as a set of dimensions, or attributes, such as location, companion and time, with hierarchical structure. While a traditional recommendation system considers only two dimensions that correspond to users and items, a context-aware recommendation system considers one additional dimension for each context attribute. Helou et al. [20] for example, present the 3A recommender system that targets context-aware recommendation in personal learning environments. The authors give interesting insights into technology that can be used to extract contextualized user profiles from emerging information systems. Context is measured and represented by actors, activity spaces and assets in learning environments as well as explicit interest parameters such as tags and queries of the user. The authors propose a contextual and multi-relational ranking mechanism that adapts a version of Google's PageRank algorithm to the particular modeling framework, recommending to users not only assets (content), but also relevant activities and actors to interact with [54].

Moreover, there are some approaches which incorporate temporal information to improve recommendations effectiveness. Xiang et al. [57] present a graph-based recommendation system which incorporates temporal information to model long-term and short-term preferences simultaneously. They propose a computationally efficient online method that balances those preferences in order to improve predictions accuracy. Xiang et al. [56] consider how time can be used into matrix factorization models by examining four different time effect. Time bias, where the interests and habits of the society change with time. User bias, where users may change their rating habits with time. Item bias, where the popularity of items change over time and finally user preferences shifting with time. Ding et al. [16] use a strategy, that decreases the importance of known ratings as time distance from recommendation time increases. The proposed algorithm uses clustering to discriminate between different kinds of items. Stefanidis et al. [50] propose a framework for time-aware recommendations that models the different types of time effects, that is, the age and the temporal context of ratings. Furthermore, they consider different cases for selecting the appropriate set of users for estimating the recommendations of a user. Additionally they introduce the notion of support in recommendations to model how confident the recommendations of an item for a user is, in order to deal with the sparsity of the explicitly defined user ratings.

Agarwal et al. [3] propose a recommender system that augments users through a subspace clustering algorithm. The proposed algorithm, also addresses the challenges associated with sparse, high-dimensional, binary-valued data, that are prominent in the research paper domain. However, only binary ratings are considered, and therefore, the problem is simplified, since typically ratings lie in a value range with higher values indicating stronger preferences. Li et al. [30] also use sub-space clustering to improve the diversity of recommendations, the dimensions considered though, are not the items but rather more general information extracted upon these items. This way, neither the high dimensionality of the data nor the missing values problem is confronted.

Baltrunas et al. [8] introduce the idea of micro-profiling, which splits the user preferences into several sets of preferences, each representing the user in a particular temporal context. The predictions are computed using these micro-profiles instead of a single user model. The main focus of this work is on the identification of a meaningful partition of the user preferences using implicit feedback.

2.2 Group Recommendations

While traditional research on recommender systems has almost exclusively focused on providing recommendations to single users, there exist many cases, where the system needs to suggest items to groups of users [6, 14]. As an example consider a group of friends deciding to dine at a restaurant. Given the potentially numerous options, the group would favor a recommendation of a restaurant, which is consistent with the preferences of its members and does not make a member unhappy with respect to the rest of the group. Existing methods for group recommendations basically follow one of two paradigms. The first, the *single-user group* approach, is to explicitly construct a group profile by combining (aggregating) the profiles of individual members. In this way, the group can be treated as a pseudo user, and thus standard techniques can be employed to provide recommendations for the group. The second paradigm, the *multi-users group* approach, is to first compute recommendations for each member separately, and then employ an aggregation strategy across them to compile the group recommendations. Both of those paradigms suffer from some drawbacks. The first is that by using a fix aggregation strategy, assumptions should be made about the decision dynamics of the group. The second drawback is that the group members are treated as individuals users, without taking into account that the preferences of a user that belongs in a group, may be different than the preferences he may have when as an individual.

To encounter these drawbacks [47] attempts to explicitly learn the discrepancies between individual and group rating behavior. The authors base their approach on the understanding of the idea that people assume different roles (e.g., leaders or followers) when in groups, or even across groups (e.g., in work and in family), and thus may exhibit substantially different behavior compared to them acting individually. They propose two models. The first make use of the difference between the individual and group ratings averaged over all items in the group profile, and employ matrix factorization to compute any missing values. In this way they can take into account the different rating behavior of users within groups and across groups. The second model presupposes that group ratings are computed as a weighted average of member ratings. Therefore, the weight of a particular user captures her behavior change as a group member.

In most studies on group recommendation, the focus is on recommending one item at a time. The various aggregation methods are applied to find a consensus between the users on a single item. The final group recommendation list consists of items that are the more satisfying for the group. However even the best item in the group list, may still not satisfy all the members of the group. To compact this many studies have included the concept of fairness in group recommendations. In [31] the authors handle the fairness aspect by calculating the *user utility* of the members of the group. The utility is determined by how relevant the recommended items are to the user. For evaluating the overall satisfaction of the users estimating group recommendation quality, they make use of this utility function to consider the social welfare - the sum of user utilities inside the group - and the fairness - the balance of user utilities inside the group. To this direction, [5] formalize the concept of *group disagreement score*, which reflects the level at which group members disagree with each other. Furthermore they define the *group relevance score* as the degree to which an item is preferred by the members. The authors propose that the final relevance score of an item to the group should be the weighted sum of the two component scores.

Shuyao et al.[43] analyze the problem of recommending a package - or a bundle of items - to a group. For example, a group of friends wants to order food and then watch a movie. The package in this case will be a pair, that consists of the restaurant they will order food from and the movie they will decide to watch.

In accordance to the previous works mentioned, the package needs to be fair for all group members. The authors propose two probabilistic models for package recommendation that incorporate factors such as user impact, package viability, and fairness. The first, initially computes the probabilities that the group of users likes individual items, before deriving the probability that the group would select a package of items. The second, first forms item packages that are favored by the individual group members before identifying those that have high likelihood to be selected by the group.

In Stefanidis et al. [51] the authors consider a different aspect of group recommendations. Whereas in previous research, group recommendations focus on the relevance of the item to the group members, here, they study group recommendations when specific constraints apply to the members of the group. For example, a vacation package may seem more attractive to a user, if the other members of the group are of a similar age. Or a course may be recommended to a group of students that have similar or diverse backgrounds depending on the scope of the course. Those constrains are divided into three groups. First is the user-to-item constraints that express any limitation a user has for any given item. In order to ensure low disagreement between the group members, the user-to-group constraints are applied. Both user-to-item and user-to-group constraints describe limitations from the user, or customer, perspective. From the perspective of the company, group-to-group constraints refer to a set of properties that the group under construction must satisfy. These properties express the requirements of the company concerning the group that a product, or item, is targeting on.

In Ntoutsi et al. [37] the authors use a hierarchical agglomerative clustering algorithm to cluster users into groups with strong similarity. They use those clusters to efficiently locate similar users to a given one. The recommendations for users are produced with respect to the preferences of their cluster members without extensively searching for similar users in the whole database. Top-k group recommendations are computed by aggregating the personal recommendations of the individual users, while they are presented along with explanations on the reasons that the particular items are being suggested to the group. Finally, to deal with the sparsity of the explicitly defined user preferences, they introduce the notion of support in recommendations to model how confident the recommendation of an item for a user is. They define the notion of support as the percentage of the friends of a user that have expressed a preference for a given item.

2.3 Recommendations in the Health Domain

Nowadays, patients turn towards the Internet, in order to inform themselves about their diseases and their possible treatment. Although this is regarded as a good thing, it suffers from two main problems. Firstly the information found on the web is not always accurate and secondly it is very diverse. To face these problems a personalized recommender would allow the users to have a seamless, secure and consistent bi-directional linking of clinical research and clinical care systems, and thus empowering the patients to extract the relevant data out of the overwhelming large amounts of heterogeneous data and treatment information.

To help patients many studies have been made to accommodate the patients needs. Kim et al. [25] propose a context-aware item-based recommendation method to establish personalized healthcare services. Also, they propose a Context Hidden Markov Model (C-HMM) based collaborative filtering method that is able to effectively recommend items in the healthcare environment. In this C-HMM based collaborative filtering method, the preference of items can be estimated using the C-HMM in the case of the absence of item preferences. In addition, collaborative filtering is performed using the estimated preference and that recommends items to users.

Wiesner et al. [42] portray the requirements that a Health Recommender System (HRS) needs to fulfill. Firstly in order to successfully blend the HRS and the Personal Healthcare Record (PHR) such a system will need to successfully cope with issues like, imprecise terms, colloquial terms and misspellings. It also needs to detect whether clinical conditions mentioned in clinical reports are negated. For example if the PHR of a user includes the following: "Autoimmune retinopathy in the absence of cancer", then the term 'cancer' has to be excluded. Furthermore a HRS must also recognize expert vocabulary (i.e., common medical abbreviations) and classification system codes primarily used by physicians and other health professionals. Next they introduced a Health Graph - a graph-based data structure of health related concepts extracted from information included in Wikipedia. Given medical facts from PHR data entries, the HRS makes use of techniques like negation detection, spell-correction and semantic query expansion via the Health Graph. According to their evaluations this gives better results than other Information Retrieval (IR) methods.

Another example of such works, is the Personal Medical Information Recommender (PMIR) [27, 28], which is part of EURECA and iManageCancer EU research projects. The tool aims to empower patients to extract the relevant data out of the overwhelmingly large amounts of heterogeneous data and treatment information. Due to the cold start and sparsity problems mentioned before, PMIR is semantically annotating patient profiles and past queries using a modular ontology and then exploits them to identify relevant documents from a high-quality corpus. The documents are annotated as well using the same ontology and an adaptation of the vector space model from the information retrieval domain is employed in order to identify relevant documents that will be eventually recommended to each user.

The wide spread of mobile applications, has also influenced the health domain. There are many applications that urge the users into bettering their health. A part of such an application usually is a recommender systems that advocates to the users, to enhance their health either through suggestions (such as what to eat in a weight loosing app) or messages (reminders how smoking affect the health in a smoking cessation app). Agapito et al. [2] for example, propose a web-based recommender system called DIETOS (DIET-Organizer System). DIETOS provides individualized nutritional recommendations according to the user's health profile defined, by following the main guidelines furnished by the medical specialist. DIETOS also provides information about the benefits and side-effects of several foods for specific diseases or health conditions. A novelty of DIETOS is that it incorporates a catalog of typical/regional foods. The recommendation of the foods by DIETOS to the users, is based on criteria certified by the medical team as well as the nutritional information of the foods contained in the database.

Hors-Fraile et al. [21], is a study that was funded through the Project Smoke-FreeBrain "Multidisciplinary tools for improving the efficacy of public prevention measures against smoking" of the European Union's Horizon 2020 research and innovation program, describes the design of two recommender systems designed to support the smoking cessation process through a mobile application. The first recommender system utilizes the hybrid approach (content-based, utility-based, and demographic filtering), to select the messages of topics more relevant for the user, that actually help them change their behavior. The second system utilizes the content-based approach, and it is used mainly to schedule a timely delivery of the messages.

López-Nores et al. [35] propose a new strategy called *property-based collaborative filtering* (PBCF), that answers problems that the previous strategies (collaborative filtering, content-based and hybrid) suffer from, in regards to the health domain. As such this approach depends on having a semantic characterization of the items that may be recommended, which is not necessarily true for the rest. Whereas the other strategies rely on direct links between specific users and items, PBCF fully decouples users and their properties on the one hand, and items and their properties on the other. This way, it is possible to build a matrix of values representing how much one item feature influences (positively or negatively) the suitability of an item for someone with a certain user property, which helps solve persistent problems of other collaborative approaches like sparsity, latency and the unfair treatment given to people whose interests and needs are different from those of the majority.

In spite of the recent trend towards health recommender systems, there are some pitfalls that needs to be overcome. Schäfer et al. [49] analyze those pitfalls, and divide them into three main categories. The a) patient/user challenges, b) the recommender challenges and c) the evaluation challenges. On the patient perspective, recommender systems will need to collect data from a wide variety of sources, such as electronic health records or lifestyle trackers, interconnect and standardize these data entries, and assure their quality regarding missing or false information. From there the specific user needs and requirements need to be filtered out from the available datasets using intelligent user models. On the recommender perspective, those models should then be used to personalize the given recommendation to the user's health context, history and goals. Once those recommendations are achieved, a health recommender system needs to guide the user to accept and implement them. While doing so, the applications additionally need to conserve the user's privacy and be conform to local laws and restrictions. Finally, when being evaluated, there is a need to define multidimensional user satisfaction measures, test those in real life situations and prevent any harmful or unethical behavior using fallbacks and expert guidance.

2.4 Beyond Simple Recommendations in the Health Domain

Although, as mentioned above, recommender systems have gained popularity in the health domain, there are many limitations in the systems implemented so far. For example, to the best of our knowledge there has been no research done in the challenging field of group recommendations neither work has been performed on incorporating the notion of fairness in the health domain. This work demonstrates some of the benefits that the health experts may gain, in pursuit of these new challenges.

More specifically, we have used techniques that are principally found in pure group recommendations systems, i.e. we have utilized the *multi-users group* approach, in order to compose the group recommendation list. However, we have tailored our recommendations for the health domain, exploiting the semantically annotated PHR profile of the users. This directly allows us to endorse documents that are relevant to a user not only on the level of appreciation (meaning the ratings that each item has gained) but also on the level of his personal health profile (we recommend items that are relevant to him because of related health artifacts).

Furthermore, by introducing the concept of fairness in our approach, in the same vain as previous works, we make sure that the final product of the group recommendation process, remains fair and unbiased towards all group members. This is particularly important in our chosen domain, where we explicitly want all members of the group to be satisfied and content with the final proposed items.

With this work, we hope to open the gate for more in depth research to the concept of group recommendations, that have as a focal point the specifications and requirements of the health domain.

Chapter 3

Single User Recommendations

Assume a recommender system in the health domain, where I is a set of data items to be rated and U is the set of patients in the system. A patient, or user, $u \in U$ might rate an item $i \in I$ with a score $r(u, i)$, as in [1, 5]. Typically, the cardinality of the item set I is high and users rate only a few items. The subset of users that rated an item $i \in I$ is denoted by $U(i)$, while the subset of items rated by a user $u \in U$ is denoted by $I(u)$.

For the items unrated by the users, recommender systems estimate a relevance score, denoted as $relevance(u, i)$, $u \in U$, $i \in I$. There are different ways to estimate the relevance score of an item for a user. As already mentioned, in the content-based approach (e.g., [34]), the estimation of the rating of an item is based on the ratings that the user has assigned to similar items, whereas in collaborative filtering systems [48], this rating is predicted using previous ratings of the item by similar users. In this work, we follow the collaborative filtering approach. First, similar users are located via a *similarity function* that evaluates the proximity between two users (Section 3.1). Then, items relevance scores are computed for users taking into account their most similar users (Section 3.2). The novelty of our approach lies in the fact that instead of only using classical similarity notions we exploit the similarity in patient profiles (their diseases), improving as we shall show in the sequel, the quality of the recommendations.

3.1 User Similarities

The information that are available to us, in order to find similar users, are the ratings that each user has given to data items as well as some personal users information regarding their health problems. Because the knowledge that we gain from each source is distinct, we can define two different similarity functions.

3.1.1 Similarity based on ratings

We assume that two users are similar if they have rated data items in a similar way, i.e., they share the same interests. We calculate their similarity based on their ratings, by exploiting the Pearson correlation metric.

$$RatS(u, u') = \frac{\sum_{i \in X}(r(u,i) - \mu_u)(r(u',i) - \mu_{u'})}{\sqrt{\sum_{i \in X}(r(u,i) - \mu_u)^2}\sqrt{\sum_{i \in X}(r(u',i) - \mu_{u'})^2}} \tag{3.1}$$

where $X = I(u) \cap I(u')$, μ_u is the mean of the ratings in $I(u)$, i.e., the mean of the ratings of u. An example for such calculations follows in section 3.3

3.1.2 Similarity based on semantic information

In the health domain, usually two people have similar interests in health documents if they have similar health problems. The International Statistical Classification of Diseases and Related Health Problems[1] (ICD10) is the *international standard diagnostic tool for epidemiology, health management and clinical purposes* maintained by the World Health Organization, which we exploit to identify similarities between health problems and eventually between users.

3.1.2.1 Similarity between two health problems

The ICD10 taxonomy can be represented as a tree, with health problems as its nodes. In the 2017 version of ICD10, there are 4 levels in the corresponding tree, in addition to the root level. Because of the structure of the ontology (acyclic), there is only one path that connects two individual nodes. Another characteristic of the structure is that sibling nodes that belong to lower levels share greater similarity than siblings that belong to upper levels.

Table 3.1 presents an example of 4 pairs of sibling nodes from the ICD10 ontology, with their code id, their description and the level they belong. From their descriptions, we can identify that the siblings that reside in the forth level share a far greater similarity than the ones in the first level.

Because of this discrepancy of the similarity of the health problems in different levels, we assign different weights to nodes according to their level. These weights will help us differentiate between siblings nodes in the various levels. We want sibling nodes in the higher levels to share greater similarity than those in the lowest.

Definition 1 (weight)**.** *Let A be a node in the ontology tree. Then,*

$$weigth(A) = w * 2^{maxLevel - level(A)},$$

[1] `http://www.icd10data.com/`

Table 3.1: An instance of the ICD10 ontology.

Code id	**Description**	**Level**
S27	Injury of other and unspecified intrathoracic organs	1
S29	Other and unspecified injuries of thorax	1
S27.3	Other injury of bronchus, unilateral	2
S27.4	Injury of bronchus	2
S27.43	Laceration of bronchus	3
S27.49	Other injury of bronchus	3
S27.491	Other injury of bronchus, unilateral	4
S27.492	Other injury of bronchus, bilateral	4

where w is a constant, $maxLevel$ is the maximum level of the tree and $level(A)$ is a function that returns the level of each node.

In addition, let $anc(A)$ be the direct ancestor of A. Intuitively, we need a formula that not only takes into account the distance between two nodes, but also the level that those nodes belong. To achieve that we make use of the concept of the lowest common ancestor (LCA).

Definition 2 (LCA)**.** *Let T be a tree. The lowest common ancestor LCA(A,B) of two nodes A and B in T is the lowest node in T that has both A and B as descendants, where each node can be a descendant of itself.*

Then, for computing the distance between A and B, we compute their distance from $LCA(A, B)$. For doing so, we identify first the path that connects A (and respectively B) with $LCA(A, B)$.

Definition 3 (path)**.** *Let T be a tree, and A and B two nodes in T, with $LCA(A, B) = C$. Then, $path(A, C)$ returns a set of nodes including A, its direct ancestor $anc(A)$, its direct ancestor $anc(anc(A))$, and so on, until we reach C, without including C in the set.*

The distance between A and C is calculated by accumulating the weight of each node in the path, as follows:

$$dist(A, C) = \sum_{n \in path(A,C)} weight(n).$$

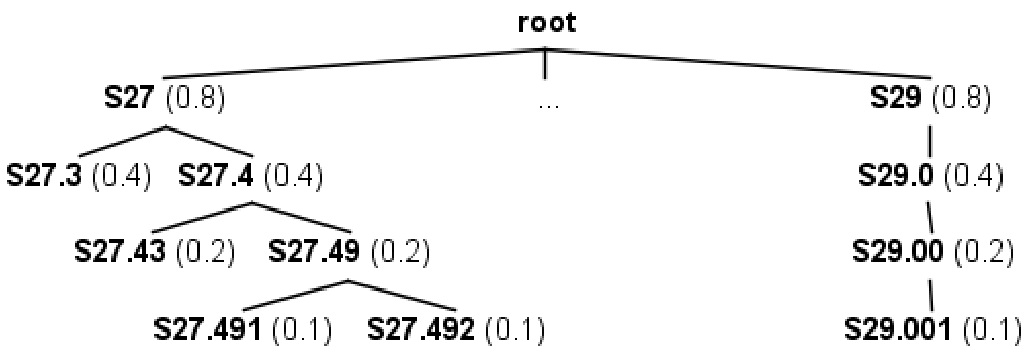

Figure 3.1: A snippet of the ontology tree along with the assigned weights in parenthesis.

In overall, for computing the similarity between two nodes A and B, we use the following formula.

Definition 4 (simN)**.** *Let T be a tree, and A and B two nodes in T, with* $LCA(A,B) = C$*. Then,*

$$simN(A,B) = 1 - \frac{dist(A,C) + dist(B,C)}{maxPath * 2}$$

Note that we divide the sum of the two distances with $maxPath * 2$, in order to normalize the overall similarity, so that the function $simN$, returns a value in the range of [0,1]. We define $maxPath$ as follows:

Definition 5 (maxPath)**.** *Let T be a tree, and A and B two nodes in T, with A being a node in the highest level and B the root. Then,*

$$maxPath = dist(A,B)$$

Figure 3.1 presents a snippet of the ICD10 ontology tree, where each node is associated with a weight (in this example, $w = 0.1$). Note that the root has not been assigned a weight, because when calculating the path that connects a node with its ancestor, we do not include the actual ancestor in the path. Table 3.2 presents various similarities between nodes from Figure 3.1.

3.1.2.2 Overall similarity between two users

Using the measures described above, we can calculate the similarity between two health problems. However, a user typically has more than one health problem in his/her profile.

Let $Problems(u)$ be the list of health problems of user $u \in U$. As such, given two users u and u', we calculate their overall similarity by taking into consideration all possible pairs of health problems between them. Specifically, we take one by one all the problems in $Problems(u)$ and calculate the similarity with all the problems

Table 3.2: Examples of similarities between nodes using figure 3.1.

Node A	**Node B**	**LCA(A,B)**	**simN(A,B)**
S27.43	S27.49	S27.4	$1-(0.2+0.2/3)=0.87$
S27	S29	root	$1-(0.8+0.8/3)=0.47$
S27.492	S27.49	S27.49	$1-(0+0.1/3)=0.97$
S27.3	S27.49	S27	$1-(0.4+0.6/3)=0.67$
S27.492	S29.001	root	$1-(1.5+1.5/3)=0$
S27.491	S27.492	S27.49	$1-(0.1+0.1/3)=0.93$

in $Problems(u')$. For each distinct problem from u, we take into account only the health problem of u' that has the maximum similarity.

Definition 6 (SemS)**.** *Let u and u' be two users in U. The similarity based on semantic information between u and u' is defined as:*

$$SemS(u,u') = \frac{\sum_{i\epsilon Problems(u)} ps(i,u')}{|Problems(u)|},$$

where

$$ps(i,u') = max(\forall_{j\epsilon Problems(u')}\{simN(i,j)\}).$$

Instead of the maximum function used in the above process, one can employ the average function. However, according to our experiments, such an approach leads to a big number of unrelated pairs of health problems.

3.2 Single User Rating Model

Let P_u denote the set of the most similar users to u, hereafter, referred to as the *peers* of u. Formally:

Definition 7 (Peers)**.** *Let U be a set of users. The peers P_u of a user $u \in U$ consists of all those users $u' \in U$ that are similar to u w.r.t. a similarity function $S(u,u')$ and a threshold δ, i.e., $P_u = \{u' \in U : S(u,u') \geq \delta\}$.*

S represents either the similarity based on ratings, $RatS$, or the one based on semantic information, $SemS$.

Given a user u and his peers P_u, if u has expressed no preference for an item i, the relevance of i for u is estimated as:

$$relevance(u,i) = \frac{\sum_{u' \in (P_u \cap U(i))} S(u,u')r(u',i)}{\sum_{u' \in (P_u \cap U(i))} S(u,u')} \tag{3.2}$$

Table 3.3: Patients with their corresponding health problems and ratings

#	**Name**	**Health Problem**	**Ratings** - DocID(score)
u_1	John Smith	S27.492	10(4), 15(5), 20(3), 30(3), 22(4), 23(2), 36(5)
u_2	Mary Jane	S27.4, S27.49	10(3), 30(5), 16(3), 19(4), 18(3), 17(4), 35(4)
u_3	Thomas Murphy	S27.3	16(5), 20(5), 30(2), 25(3), 22(3), 17(5), 36(5)
u_4	Scott Wilson	S29.00	25(5), 45(2), 19(2), 17(5), 31(5), 35(5)
u_5	Mia Brown	S29, S27.49	15(4), 20(4), 30(3), 45(4), 22(5), 36(5), 18(5)

Table 3.4: *RatS* scores for patient John Smith

Patient	**Pearson Score**
Mary Jane	-0.993
Thomas Murphy	0.389
Scott Wilson	0
Mia Brown	0.897

After estimating the relevance scores of all unrated user items for a user u, the items A_u with the top-k relevance scores are suggested to u.

3.3 Running Example

Assume that we have 5 patients in our health recommender system. Their health problems and the ratings that they have given are shown in Table 3.3.

Using the *similarity based on ratings* (Equation 3.1), *RatS*, for user u_1, we calculate the similarity scores shown in Table 3.4. As an example, let's consider the users u_1 and u_5. To calculate their *RatS* similarity, we find first the mean value of the users ratings. For user u_1 is 3.71 and for user u_5 is 4 (these patients have rated three common documents, i.e., the documents with id 15, 22 and 36). Applying these values to the similarity function, we came up with the value of 0.897. The same procedure is used for the rest of the patients.

We will again use u_1 and u_5 to calculate the similarity scores using the semantic similarity function. The patient u_5 suffers from two health problems. As such, we will compute two times the *simN* function. The first will be for S27.492 and S29. This gives a score of 0.234. The second will be for S27.492 and S27.49, with similarity score equal to 0.97. As the final similarity score of the two patients, we will take the maximum value. The scores using the semantic similarity function

Table 3.5: *SemS* scores for patient John Smith

Patient	Partial Scores	Similarity Score
Mary Jane	S27.492 - S27.4 = 0.9	0.967
	S27.492 - S27.49 = 0.967	
Thomas Murphy	S27.492 - S27.3 = 0.634	0.634
Scott Wilson	S27.492 - S29.01 = 0.034	0.034
Mia Brown	S27.492 - S29 = 0.234	0.967
	S27.492 - S27.49 = 0.967	

Table 3.6: The Peers and Recommendation Lists for patients John Smith and Mary Jane

Patient	Similarity function	Peers	Recommendation List - DocId(score)
John Smith	*RatS*	Thomas Murphy, Mia Brown	18(5), 16(5), 17(5), 28(3), 31(3), 25(3), 45(2)
	SemS	Mary Jane, Thomas Murphy, Mia Brown	18(5), 17(4.4), 35(4), 19(4), 16(3.8), 18(3), 25(3), 28(3), 31(3), 45(2)
Mary Jane	*RatS*	Scott Wilson	25(5), 31(5), 45(2)
	SemS	John Smith, Thomas Murphy, Mia Brown	15(5), 36(5), 22(4.1) 20(3.8), 25(3), 28(3), 31(3), 23(2), 45(2)

for u_1 are shown in Table 3.5.

After having found the similarity scores for a patient, the next step is to find his peers (Definition 7). Subsequently, by exploiting Equation 3.2, we compute the recommendation list for the patient (as threshold δ used in Definition 7, we use the value 0.3). Table 3.6 shows the peers and the recommendations for the users u_1 and u_2. Recommendations are ordered from the most to the least relevant. In case of ties, we resolve them arbitrarily.

Chapter 4

Group Recommendations

However, we are not only interested in recommending useful information to single users but to a whole group of patients. Our goal is actually to provide valuable suggestions to a caregiver who is responsible for a group of patients. We interpret valuable suggestions as suggestions that are both highly related and fair to the patients of the group. The overall process of computing a group recommendation list is shown in figure 4.1. In section 3.1 we discussed about the similarity functions and the relevance function was mentioned in section 3.2. In this section we will examine four different aggregation methods.

4.1 Group Rating Model

Most previous works focus on recommending items to individual users. Recently, group recommendations that make recommendations to groups of users instead of single users (e.g., [37, 45]), have received considerable attention. Commonly, a method for computing group recommendations first estimates the relevance scores of the unrated items for each user in the group, and then, aggregates these predictions to compute the suggestions for the group. Formally, the relevance of an item for a group is computed as follows:

Definition 8 (relevance)**.** *Let U be a set of users and I be a set of items. Given a group of users G, $G \subseteq U$, the group relevance of an item $i \in I$ for G, such that, $\forall u \in G$, $\nexists rating(u, i)$, is:*

$$relevanceG(G, i) = Aggr_{u \in G}(relevance(u, i)).$$

As in single user recommendations, the items with the top-k relevance scores for the group are recommended to the group.

4.2 Fairness in Group Recommendations

As already mentioned, in this work, we exploit the concept of group recommendations in order to provide valuable suggestions to a caregiver responsible for a

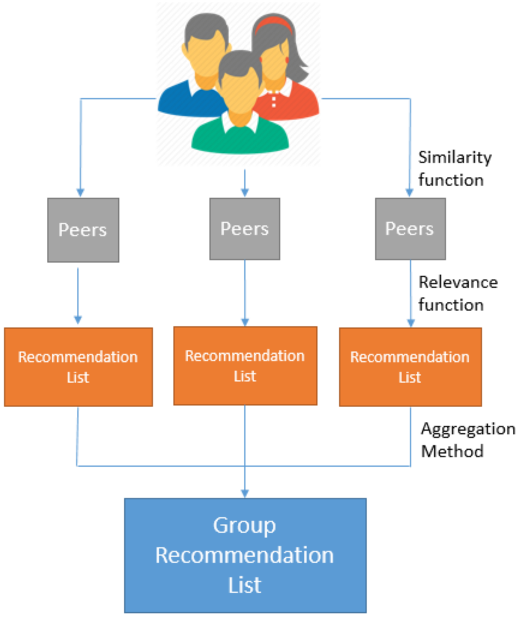

Figure 4.1: An overall example for the actions necessary to get a group recommendation list.

group of patients.

Specifically, given a particular set of recommendations for a caregiver, it is possible to have a user u that is the least satisfied user in the group for all items in the recommendations list, that is, all items are not related to u. Therefore, although the caregiver may like as a whole the set of recommendations, the package selection is not fair to u. In actual life, where the caregiver is concerned for the needs of all patients in his group, we should recommend items that are both strongly relevant and fair to the majority of the group members.

Motivated by this observation, we target at having insights into the properties of the produced recommendations in order to help making the algorithmic process transparent, non-discriminative and accountable [52]. In particular, to increase the quality of the recommendations for the caregiver, we consider, similar to [44], a fairness measure that evaluates the goodness of the recommendations as a set. This way, given a user u and a set of recommendations D, we define the degree of fairness of D for u as:

$$fairness(u, D) = \frac{|X|}{|D|}, \tag{4.1}$$

where $X = A_u \cap D$. Remember A_u are the items with the top-k relevance scores for u.

Intuitively, the fact that the group recommendations contain some highly relevant items to u, makes both u and his caregiver tolerant to the existence of other items that are not highly related to u, considering that there are other members in the group who may be related to these items.

Then, the fairness of a set of recommendations D for a set of users G is defined as follows.

Definition 9 (Fairness). *Given a group G and a set of recommendations D, the fairness of D for G is defined as:*

$$fairness(G, D) = \frac{\sum_{u \in G} fairness(u, D)}{|G|}.$$

After having defined the fairness of the group recommendations, we want to have a metric that will demonstrate the overall value – suitability – of these proposed items to the group. We want to take into account not only the above mentioned fairness, but also how "good" these recommendations are, namely their relevance score w.r.t. to the group list. Considering these, we define the fairness-aware value of D for G as follows:

$$value(G, D) = fairness(G, D) \cdot \sum_{i \in D} relevanceG(G, i). \tag{4.2}$$

4.3 Aggregation Designs

For the aggregation method $Aggr$, we employ 5 different designs, each one carrying different semantics. Specifically, we distinguish between the score-based and rank-based designs.

In a *score-based design*, the prediction for an item is computed taking into account the relevance of the item for the group members. Firstly, we consider that strong user preferences act as a veto; this way, the predicted relevance of an item for the group is equal to the minimum relevance of the item scores of the members of the group:

$$relevanceG(G, i) = \min_{u \in G}(relevance(u, i)). \tag{4.3}$$

Alternatively, we focus on satisfying the majority of the group members and return the average relevance for each item:

$$relevanceG(G, i) = \sum_{u \in G} relevance(u, i)/|G|. \tag{4.4}$$

In a *rank-based design*, we aggregate the group members recommendations lists by considering the ranks of their elements. Specifically, following the Borda count method [18], each data item gets 1 point for each last place received in the ranking, 2 points for each next to last place, and so on, all the way up to k points for each first place received in the ranking. The item with the largest point total gets the first position in the aggregated list, the item with the next most points takes the second position, and so forth, up to locate the best k items. Overall, the points of each item i for the group G is computed as follows:

$$points(G, i) = \sum_{u \in G} (k - (p_u(i) - 1)), \tag{4.5}$$

where $p_u(i)$ represents the position of item i in A_u.

Targeting at increasing the fairness of the resulting set of recommendations, we introduce also the $Fair$ method, which consists of two phases. In the first phase we consider pairs of users in the group, in order to identify what to suggest. In particular, according to this rank-based design, a data item i belongs to the top-k suggestions for a group G, if, for a pair of users $u_1, u_2 \in G$, $i \in A_{u_1} \bigcap A_{u_2}$, and i is the item with the maximum rank in A_{u_2}.

For locating fair suggestions in this design, initially, we consider an empty set D. Then, we incrementally construct D by selecting, for each pair of users u_x and u_y, the item in A_{u_x} with the maximum relevance score for u_y. The above process is shown in Algorithm 1.

If the value of k (i.e., the number of items that need to be provided by the group recommendation list) is greater than the items we found using the above method, then we construct the rest of D, by serially iterating the A_u lists of the group members and adding the item with the maximum rank that does not already exists in D.

As we have already said, the size of the set I, i.e., the number of items in the recommender system, is typically high. In many cases, regardless of how similar the group members are, the first phase of the algorithm may yield few items (i.e., less than out targeted k). In this case, regardless of what order we examine the members, the items we get are the same. Moving to the second phase, we can assume a pseudo hierarchy inside the group members, meaning that the members that will be checked first, will have more relevant to them items in the group list. So, by rearranging the order of the group members, we can influence the fairness achieved for each individual member. On the other hand, if we produce all the $top - k$ items from the first phase, then a number of items in the group list, may change accordingly to what order we examine the members. Again the vast majority of the items will be included, regardless of the members order. In both cases, the fairness of the list for the entire group (Definition 9) does not change.

4.4 Running Example

Continuing from the previous Section 3.3 , we define as a group, the users u_1 and u_2. Their individual recommendation lists are shown in Table 3.6. Using the aggregation methods mentioned above, Tables 4.1 and 4.2 present the produced group recommendation lists. For all methods, we take into account only the documents that exist in the individual recommendation list of all patient in the group.

In more detail, given the individual recommendation lists based on $RatS$, for the users u_1 and u_2, we can observe that only the documents 25, 31 and 45 are

Algorithm 1 Fair Group Recommendations Algorithm

Input: A group of users $G = \{u_1, \ldots, u_n\}$, and the sets of recommendations A_{u_x} for each user $u_x \in G$.

Output: The fairness-aware set of the z recommendations D for G.

```
begin
D = ∅;
while |D| < z do
    for x = 0; x < n; x++ do
        for y = 0; y < n; y++ do
            if x ≠ y then
                Find the item i ∈ A_{u_y} with the maximum relevance(u_x, i);
                D = D ∪ i;
end
```

Table 4.1: The Group Recommendation list using *RatS*

#	**Min**	**Average**	**Borda**	**Fair**
1	31(3)	31(4)	31 (5 pts)	31
2	25(3)	25(4)	25 (5 pts)	25
3	45(2)	45(2)	45 (2 pts)	45

Table 4.2: The Group Recommendation list using *SemS*

#	**Min**	**Average**	**Borda**	**Fair**
1	25(3)	25(3)	25(9 pts)	25
2	31(3)	31(3)	28 (7 pts)	28
3	28(3)	28(3)	31 (5 pts)	31
4	45(2)	45(2)	45 (2 pts)	45

common. The min and average methods, are straightforward. For the Borda count method, we will consider the document 25. For u_1, it is in the sixth position in his recommendation list, and so it takes 2 points $(7-(6-1))$. For u_2, the document is in the first position, so it takes 3 points for total of 5 points. Finally, for the Fair approach, for u_1, we take the document in the list of u_2 that has the maximum score, and that it also present in the list of u_1. That is the document 25, with value equal to 5.

Chapter 5

Experimental Evaluation

In this section, we initially describe the dataset and the evaluation metrics used. Then, we present the results of our experimental evaluation.

5.1 Dataset

Although it is really common for patients to look for health information and sometimes to rate such documents on the web, their profiles are usually not accessible, neither linked to those documents. Among others, legal and ethical constraints prohibit the collection and the exploitation of such a dataset.

In order to experiment with such a dataset we initially exploited 10.000 chimeric patient profiles. These profiles [23] contain the same characteristics that exist in a real medical database such as patients' admission details, demographics, socioeconomic details, labs, medications, etc. Additionally, the health problems for each patient, are described using the ICD10 ontology making this dataset ideal for our semantic similarity approach.

Then, based on these profiles, we synthetically generated a document corpus and user ratings as follows:

• **Create document corpus.** Initially, we generated a *numDocs* number of documents, for each first level category of the ICD10 ontology (i.e., for each node that belongs in the first level of the ontology tree). For their corresponding keywords, we randomly selected $numKeyWords$ words from the description of the nodes in each subsequent subtree.

• **Divide the patients into groups.** We have surmised that all patients have given a *numRatings* number of ratings. Specifically, we have divided the patients into three groups – *sparse*, *medium* and *dedicated*. The users in each group have given *few*, *average* and *a lot* of ratings, respectively.

• **Simulate a power law rating distribution.** When ranking items based on human preferences, they tend to follow the power law distribution. In order to depict this, we have randomly selected a *popularDocs* number of documents that will be the most popular.

• **Generate items to rate.** For each patient, we have divided the ratings, that he/she will give into two groups: *healthRelevant* and *nonRelevant*. Using the health problems data (noted in ICD10 nodes) the first group of rating will go to the documents belonging in the same subtree as their health problems. The second group will be given to the rest of the documents. Our assumption here is that the patients will be interested not only in documents regarding their health, but also to some extent in others as well.

• **Generate ratings.** Finally, for each rating generated in the previous step, we assigned randomly, *a value* in the range of 1 to 5.

A more detailed analyses of the parameters, for building the document corpus and the rating dataset, is given in Section 5.4.1

5.2 Aggregation Methods

As we have mentioned in Section 4.3, the aggregation methods we want to utilize are divided into two groups. The score-based group consists of the Minimum and Average aggregation designs, while the rank-based group is composed of the Borda and Fair designs. In order to have a baseline for which to compare these methods, we will employ a trivial aggregation method, namely the Round-Robin aggregation design. The way that this method constructs the group list is straightforward. We consider each member of the group individually, and for each one, we will take the item in his/her list with the highest score that does not already exists in the group list. Because we consider each member of the group as an isolated user, any notion of fairness that this design will offer, will be incidental and highly dependent on the similarity between the members and specifically, on them having similar items with high score. Given that during the aggregation process of Round-Robin, we only consider the ranks of the items, it will be included in the rank-based group of the aggregation methods.

5.3 Evaluation Measures

For our experiments, in order to calculate the semantic similarity function *SemS*, we used the value of 0.1 for constant w that is needed in Definition 1.

To evaluate the similarity functions, we used the Mean Absolute Error (MAE) and the Root Mean Square error (RMSE). These two are the most common metrics used to measure accuracy for continuous variables - such as the similarity scores [38]. MAE signifies the average of absolute errors of the predictions, compared to the actual given ratings for a user:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |predicted_i - actual_i| \tag{5.1}$$

RMSE expresses the average of squares of absolute errors of the prediction, compared to the existing rating for a user:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(predicted_i - actual_i)^2} \tag{5.2}$$

Intuitively, we can see that the smaller the values, the better the quality of recommendations.

To quantify the success of each aggregation method, we computed the distance of each user's $top - k$ recommendation list with that of the group's. Then to calculate the final score we took the average of those. For calculating the distance we used two measures. The Kendall tau (Equation 5.3) and the Spearman footrule distance (Equation 5.4). The Kendall tau distance is a metric that counts the number of pairwise disagreements between two ranking lists [24]. The Spearman footrule is the absolute difference between the ranks assigned to an item in each list [15].

The Kendall tau distance between two lists t_1 and t_2 is:

$$\begin{aligned} K(t_1, t_2) = &|\{(i, j) : i < j, (t_1(i) < t_1(j) \wedge t_2(i) > t_2(j)) \\ &\vee (t_1(i) > t_i(j) \wedge t_2(i) < t_2(j))\}| \end{aligned} \tag{5.3}$$

where $t_1(i)$ and $t_2(i)$ are the rankings of the element i in t_1 and t_2, respectively. In turn, the Spearman footrule for the same lists is:

$$S(t_1, t_2) = \sum_{i=1}^{n} |t_1(i) - t_2(i)| \tag{5.4}$$

5.4 Evaluation Results

5.4.1 Datasets Creation

In Section 5.1, we analyzed the proposed method of creating two datasets - documents and ratings - in lieu of real data. In Tables 5.1 and 5.2, we present the parameters needed, a brief explanation and the value given in order to build those datasets. After all the necessary steps were completed the number of items in the document corpus was 79.650 and the total number of ratings generated was 1.576.872.

In Figure 5.1, we can see the distribution of ratings in the documents. We have partition the ratings in groups of 500. Most of the documents (71%) have received ratings in the range of [50-100]. In the second place (21%) we have the document that have been rated 0 to 50 times, while if we accumulate all the documents with ratings more than 200 they are merely make up of the 1.12% of the corpus. As was intended, these results simulate a power law, where the prominent items are few and plethora of documents have very low popularity.

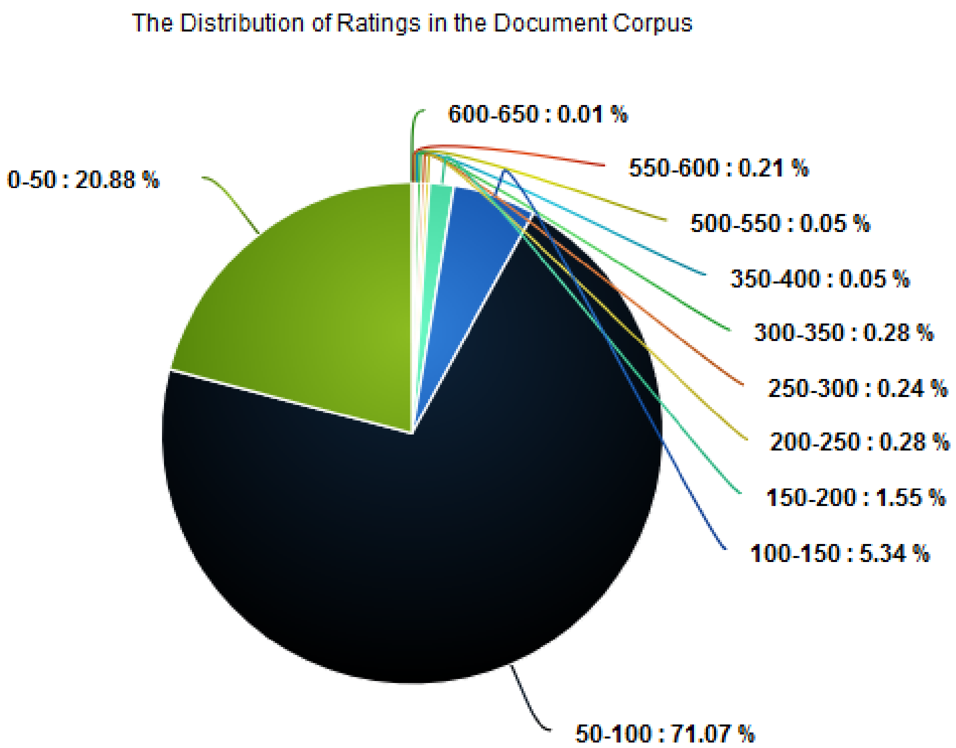

Figure 5.1: The distribution of the ratings in the document corpus, where we have partition the number of ratings in groups of 500

Table 5.1: The parameters needed to creating the document corpus.

Parameter Name	**Explanation**	**Value**
numDocs	The number of documents created for each different category of health problems, based on the ICD10 ontology tree.	270
numKeyWords	The number of randomly selected keywords, attached to each document.	10
popularDocs	The number of documents, that will be most popular in each category, in order to simulate a power law distribution.	70

Table 5.2: The parameters needed to create the ratings dataset.

Partitions	Parameter Name	Explanation	Value
Group Partition	Group *sparse*	The number of ratings given by patients in this group is 20 to 100	50% of all patients
	Group *medium*	The number of ratings given by patients in this group is 100 to 250	30% of all patients
	Group *dedicated*	The number of ratings given by patients in this group is 250 to 500	20% of all patients
Scores Partition	One	The number of ratings that have as value 1	20% of all ratings
	Two	The number of ratings that have as value 2	10% of all ratings
	Three	The number of ratings that have as value 3	30% of all ratings
	Four	The number of ratings that have as value 4	20% of all ratings
	Five	The number of ratings that have as value 5	20% of all ratings
Ratings Partition	healthRelevant	The number of documents each user will rate that are relevant to some health problem he/she suffers from	20% of ratings from each user
	nonRelevant	The number of documents that each user will rate that are not relevant to any of his/her health problems.	80% of ratings from each user

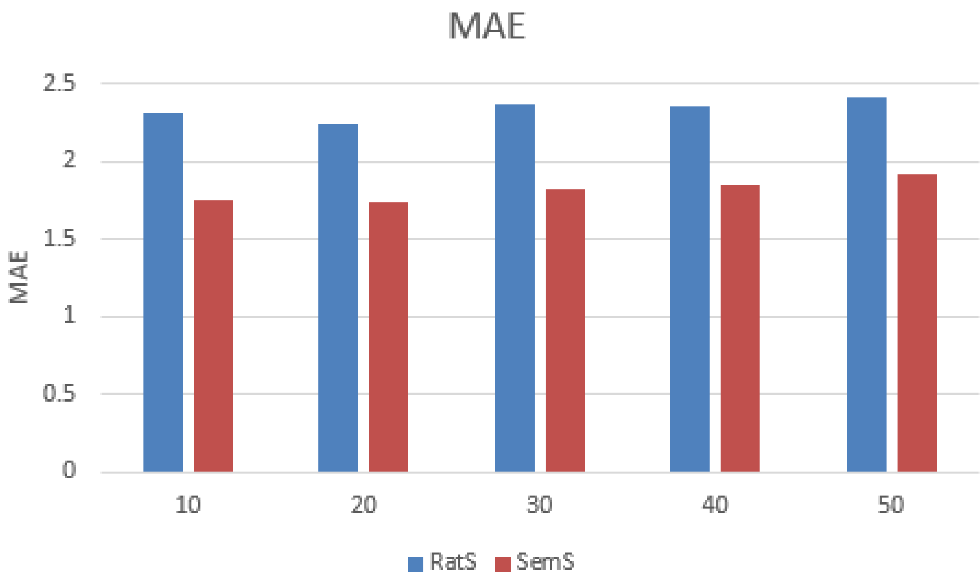

Figure 5.2: The MAE for different K

5.4.2 Evaluation of Similarity Functions

In order to compare the two proposed similarity functions, we focused on single users recommendations. We used 100 users, for which we hide their $top-k$ suggestions. In Figures 5.2 and 5.3, we can see the different values of the MAE and RMSE measures for several values for k. In all cases, we can observe that the semantic similarity function gave better results than the rating similarity function. This shows the added value of our solution on calculating effectively the similarity between users, incorporating for the health-related semantic information in its calculation.

In contrast, we can see in Figure 5.4 that the time needed to calculate the semantic similarity function is higher than the time needed for the rating similarity function. This might be considered as a drawback. However, as we can observe, calculating the semantic similarity is really fast (msecs for 10K patients), and is balanced out by the better quality of the results.

5.4.3 Evaluation of Aggregation Methods with Different Similarity Functions

In order to evaluate the effectiveness of each aggregation method, in regards to our construction of the final group recommendation list, we use the Kendall distance (Equation 5.3) and the Spearman footrule (Equation 5.4) measures. In more detail, we compute the distance - using either Kendall or Spearman - between the top-K list of each member of the group and the group recommendation list. In this way

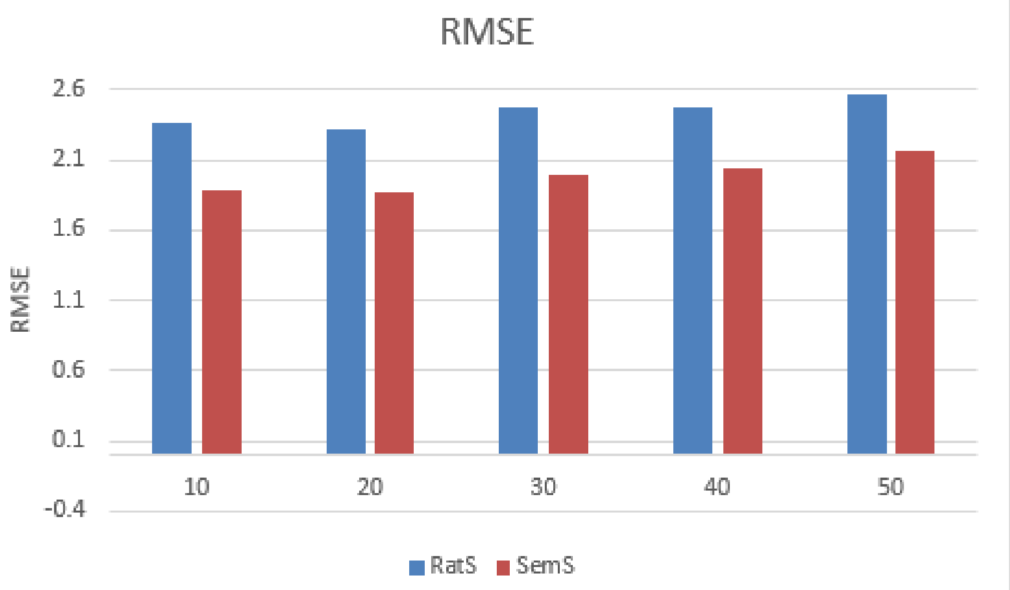

Figure 5.3: The RMSE for different K

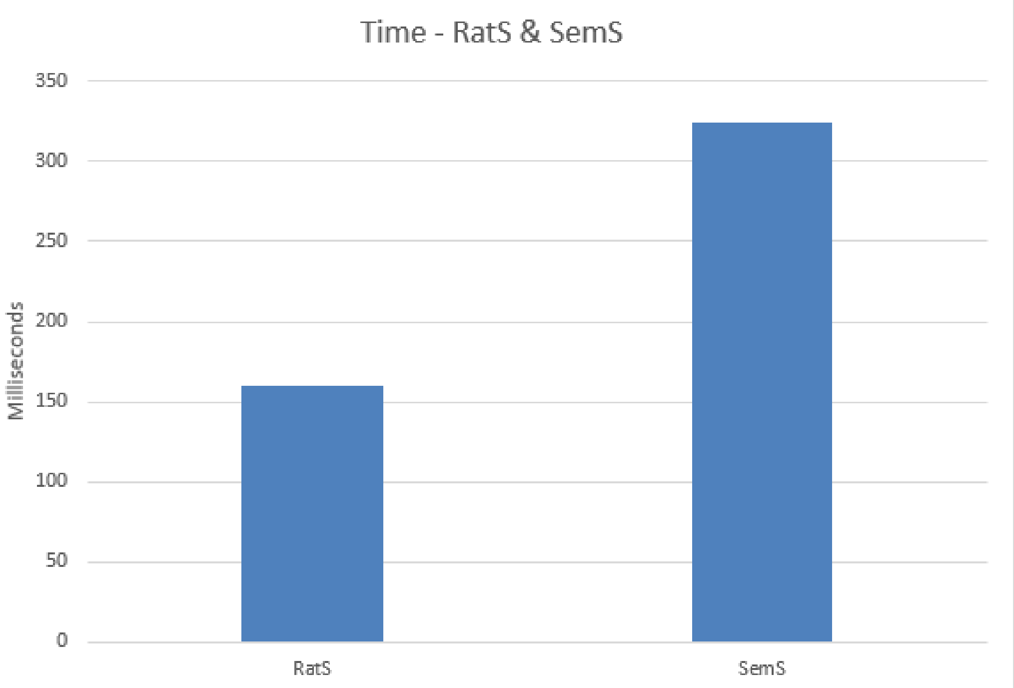

Figure 5.4: The time needed to calculate the Pearson and semantic similarity function. The time is averaged over 10 randomly selected users.

we can estimate the difference between the lists, and consequentially how many of the most highly recommended documents for each user, have been included in the group recommendations. These experiments, help us identify whether each aggregation method makes adequate use of the individual top-k list of the members of the group.

In order to produce these results, we randomly selected 10 different groups that share the same *group similarity*. Group similarity is the similarity for all pairs of users in the group, averaged over the number of those pairs. After generating the group recommendation list, we calculate for each member of the group the appropriate distance (i.e. either Kendall or Spearman). The distance score for the aggregation method is the averaged score of the summation of these distances over the number of group members. Finally, following the same procedure for all 10 groups, the overall score for each aggregation method is the mean of the previously calculated scores, over the number of different groups.

To further supplement our findings, that our proposed similarity function $SemS$ gives better results than the $RatS$, we compare the Kendall and Spearman distance for 10 different groups of size 5. The results of these experiments are shown in Figures 5.5 and 5.6, where we can see the Spearman and Kendall distance respectively for the same groups when using different similarity functions. We can observe that the $SemS$ function offers better results than those of $RatS$, regardless of which aggregation design we utilized.

As our proposed method gives better results, in the subsequent evaluation tests, we will only consider the $SemS$ similarity function.

5.4.4 Evaluation of Aggregation Methods with Different Group Size

In order to get more accurate results, we expanded on the experiment presented above, with different sized groups. The group sizes we examined are 5 and 7. The methodology for these experiments are the same as described above. In Figure 5.7 we report the Kendall distance and in Figure 5.8 the Spearman distance.

In Figure 5.7, we can observe that, using the Kendall distance, the rank-based methods give better results than those of the score-based methods. This is because the score-based methods consider the entirety of a user's list, while the rank-based methods consider only the $top-k$ items. Furthermore, the Minimum and Average designs, take into account the scores given to an item. For example, if given one item, if a member of the group has a radically different relevance score for it than the rest (i.e. the user has a relevance score of 1, while the rest of the group has above 4), in the case of Minimum aggregation design, his opinion will act as veto, and the item's group relevance score will be 1, while in the case of Average, its group relevance will be brought down and might not make it into the group list. In both cases, an item that has high relevance for all but one members of the group, will not be included in the group recommendation list, something that the rank-based aggregation methods can circumvent. Because of this, the rank-based

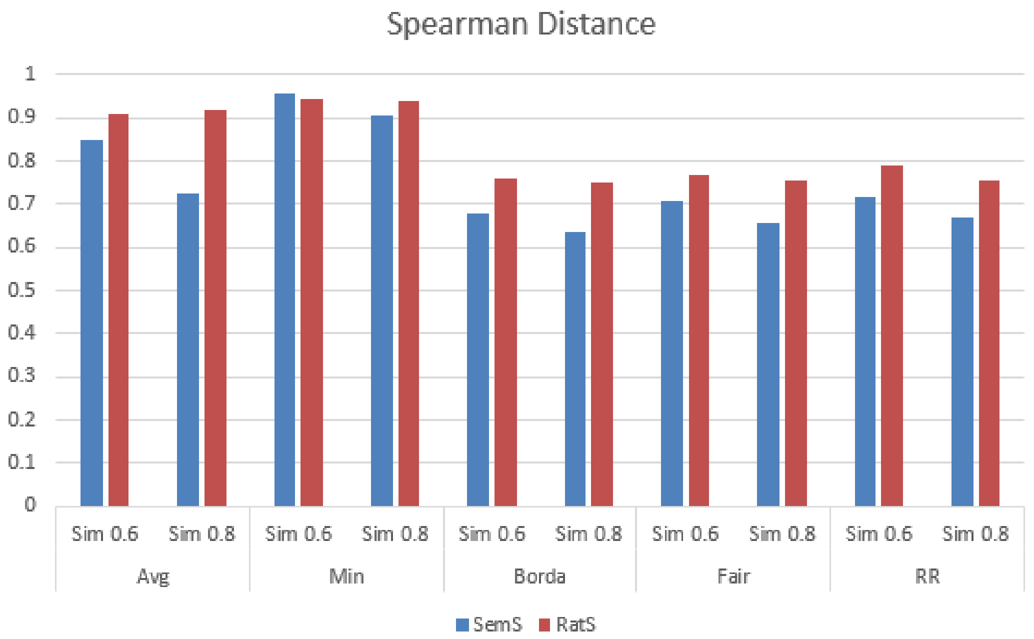

Figure 5.5: The Spearman distance averaged over 10 groups, for different aggregation methods when utilizing the *RatS* and *SemS* similarity functions

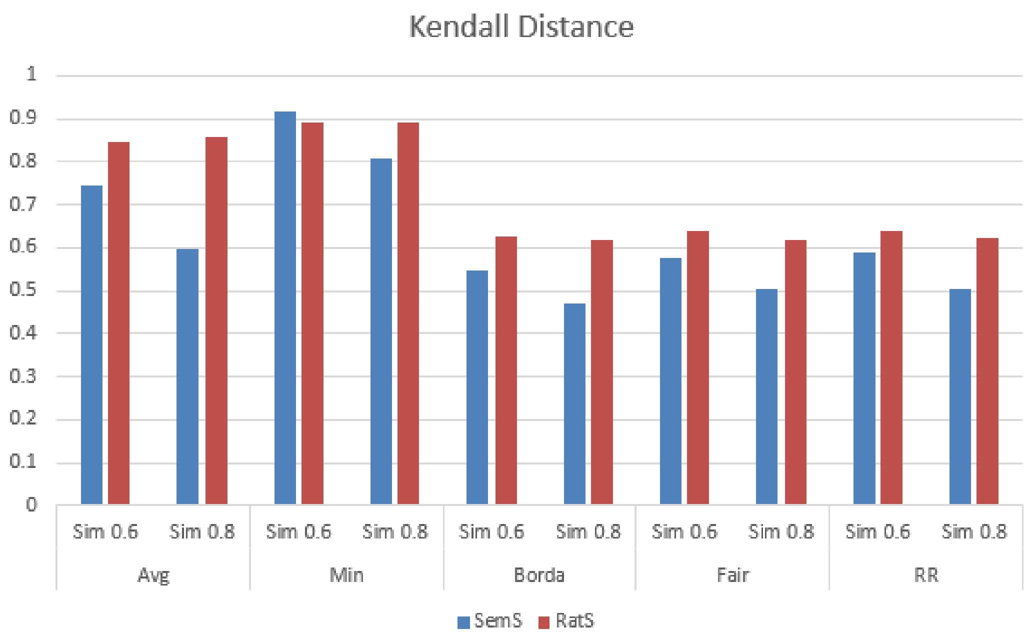

Figure 5.6: The Kendall distance averaged over 10 groups, for different aggregation methods when utilizing the *RatS* and *SemS* similarity functions

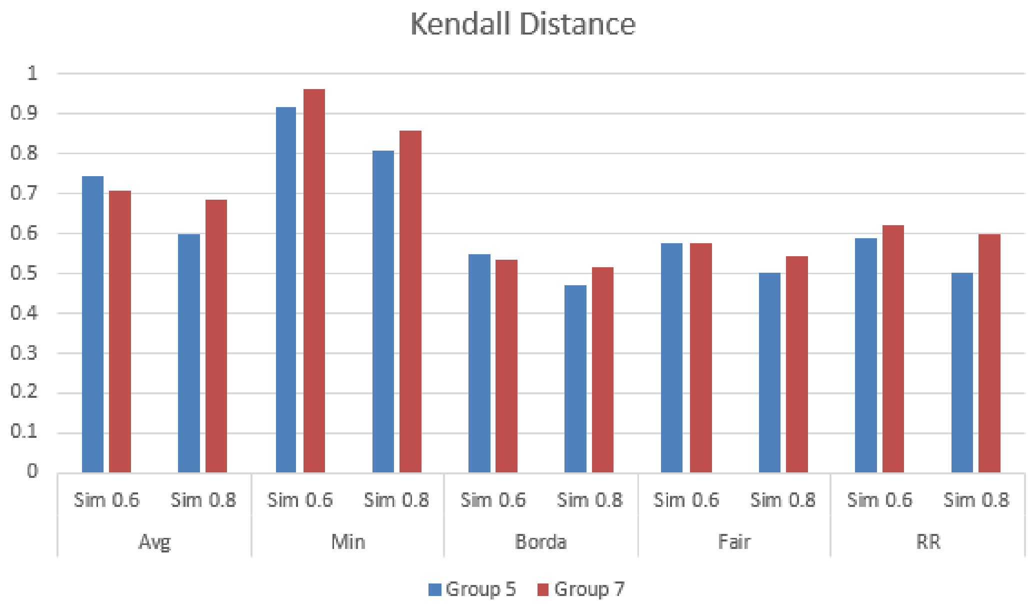

Figure 5.7: The Kendall distance for different group similarity, group size and aggregation method.

methods, are able to include more items from the members individual $top - k$ recommendation lists, and hence give lower distances.

From a closer look into the results of the rank-based methods, when making comparisons between them, we observe that Round-Robin is the worst, while Borda gives slightly better results than the Fair design. Since all rank-based methods focus on constructing the group list with as many items from the members $top - k$ lists as possible, a more constructive comparison between them will be given in Figure 5.9, where we compare the fairness of the designs.

In addition, as expected, when the group similarity gets higher, all algorithms provide better results. Finally, the size of the group, given that we consider groups sharing the same group similarity, slightly affects the quality of the results of the employed aggregation methods, and overall, the bigger the size of the group, the higher the Kendal tau distance. Figure 5.8 shows similar results for Spearman as well.

In order to calculate the fairness (Definition 9) of the aggregation methods, we follow the same methodology as the calculation of the distance. Since with fairness, we measure in essence how many of the items in an individual top-K list made it in the group recommendation list, which is inherently what we shown in Figures 5.7 and 5.8, we expect complementary results. In fact, the Minimum method that had the highest distance now produces the lowest fairness. What is intriguing are the results of Borda and Fair algorithms. Although Borda gave marginally lower distance, Fair now in turn gives better fairness. An explanation for these results

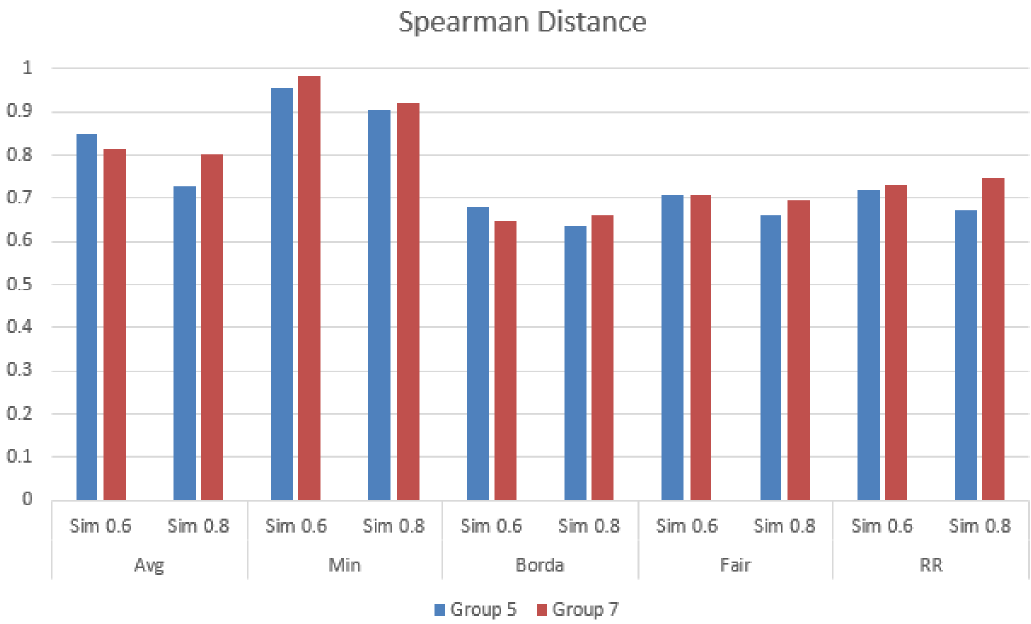

Figure 5.8: The Spearman distance for different group similarity, group size and aggregation method.

is that during the first phase of the Fair algorithm, we take into account items regardless of their relevance score. As an example, if two users share a common recommendation to a document, but its relevance score is low for both, so that item recites in the end of the two lists, the Fair algorithm will include it (since first we check if two users share a common recommendation, and only if there are many, we check their relevance scores), while the Borda will give it just 2 points. With Borda providing only 2 points to this item that is relevant to two users, it will probably not be included in the group list, in favor of a different item that has a higher relevance but only for one member. For example if the $top - k$ list consists of 20 items, an item that shows only in one of the recommendation lists of the users, but is in the first place, it will get 20 points. In these circumstances, because we want to include in the group list items that are relevant to most users, the Fair algorithm behaves better and offers higher fairness.

Finally, we can see that the Round-Robin method gives lower fairness than the rest of the rank-based methods. That is because, with Round-Robin, we do not consider the rest of the group members when constructing the recommendations list, but we consider each member individually from the others. That way, we may include items that are relevant to only one particular user, something that the Borda and Fair algorithms try to lessen.

As described in Section 4.3, we divide the aggregation methods into two groups. The score-based designs (i.e., Average, Minimum) and the rank-based designs (i.e., Borda, Fair, Round-Robin). Because of the inherent differences of these designs,

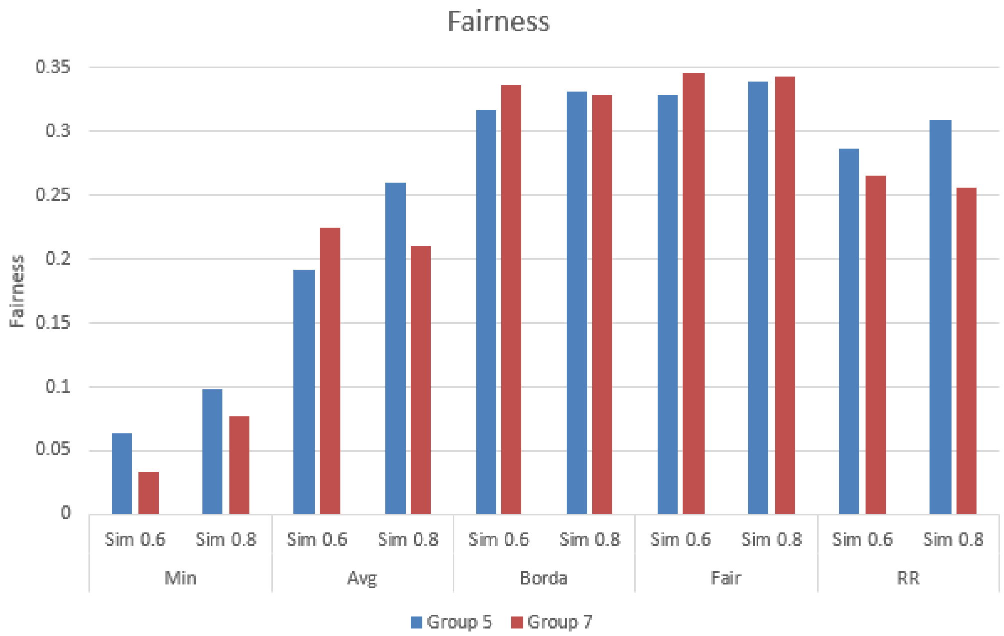

Figure 5.9: The fairness for different group similarity, group size and aggregation method.

we cannot directly compare the value (Equation 4.2) returned from the methods across different categorizations. To elaborate more, when we aggregate using the score-based techniques, for any given item we directly compute its corresponding relevance score in the final group recommendation list (i.e. using the average aggregation method, the final score of an item, is the summation of the scores of that item from all individual lists of the group members, averaged over the size of the group). On the other hand, with the rank-based techniques what we actually calculate is the rank that a specific item will have in the group recommendation list. But as we define in Equation 4.2, in order to find the value of a group recommendation list, we need the relevance score of all its items. Thus, we define the relevance score of an item in a group recommendation list, that is produced by a rank-based technique as the summation of its rank in each individual recommendation list of the group members. Similar to Borda if an item is not present in one of the list then the score it takes is equal to 0. As it is apparent, we cannot directly compare score-based and rank-based aggregation methods. The score-based methods give as relevance score to an item, in the range of [1,5] while the rank-based approaches, given that the group recommendation list consist of k items and the size of the group is s, give a relevance score in the range $[1, s * k]$.

In Figure 5.10 we can see the *value* for the score-based aggregation methods. As was expected, the Average method offers much better results. In Figure 5.11 we compare the rank-based methods. These results complement those in Figure 5.9.

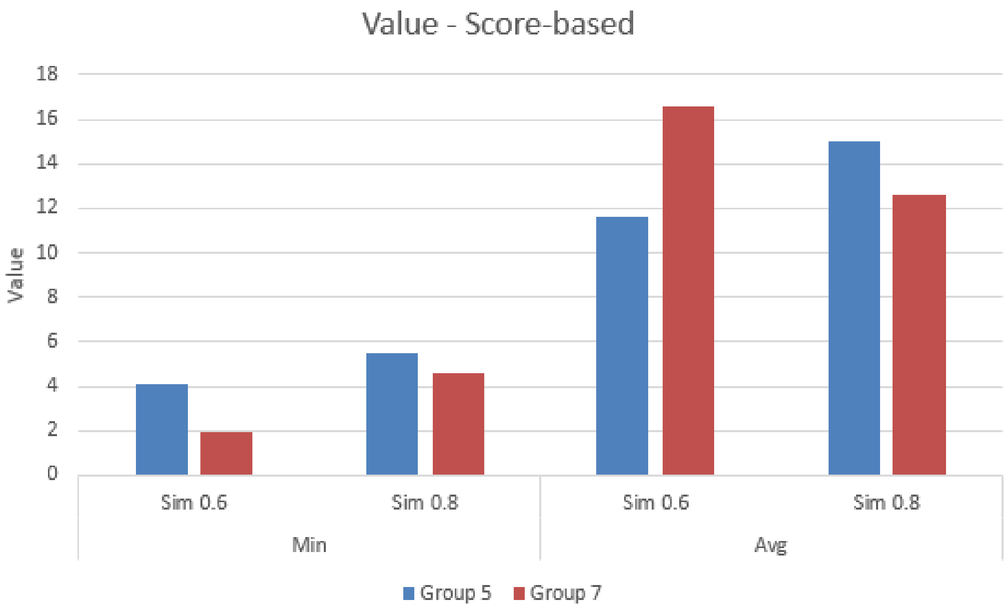

Figure 5.10: The value for the score based aggregation methods.

The Fair algorithm offers better overall value for the group recommendation list. As expected the worst results are presented by the Round-Robin. As mentioned above, the fairness that Round-Robin offers is highly incidental and the relevance scores for those are low, since most users do not share the same items with the same high scores.

Finally in Figure 5.12, we calculated the time needed to aggregate the individual lists for each method. For each group size we took randomly 10 different groups. All the groups shared the same group similarity. The most time costly method is the Average. The time needed for the Fair algorithm is marginally more, than the one needed for the Borda method. Nevertheless, the execution time is really small (at most 4 msecs) and since the Fair algorithm offers better results (as shown in Figure 5.11 where the overall performance of the methods is presented), this is a small drawback that we can afford. Finally, the Round-Robin is the fastest algorithm, since it requires little to none operations.

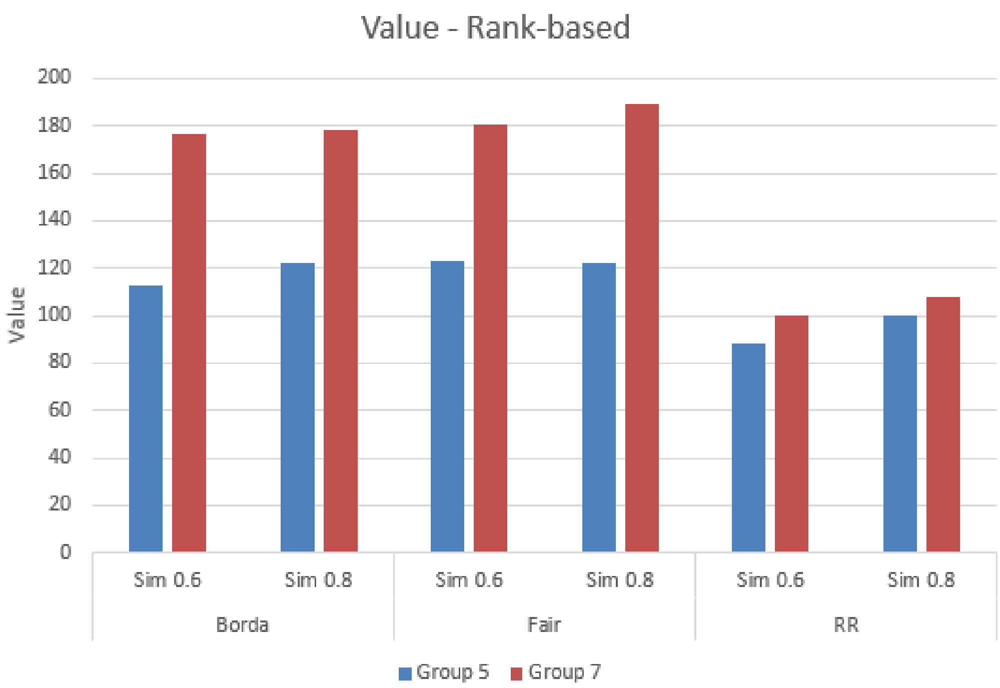

Figure 5.11: The value for the rank based aggregation methods.

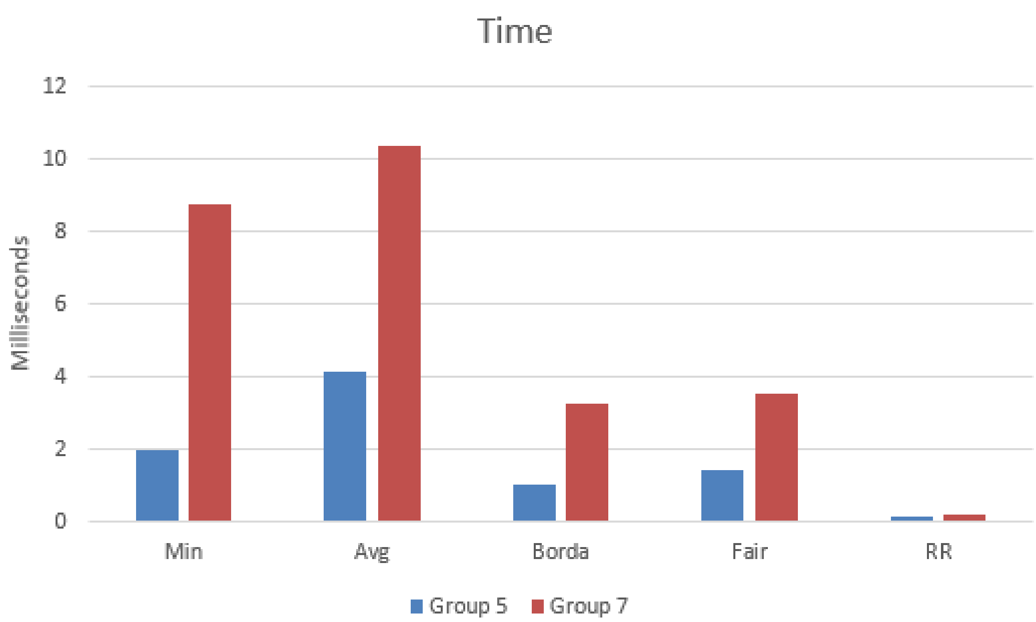

Figure 5.12: The execution time for each aggregation method. Calculated for group sizes: 5,7 and group similarity 0.6

Chapter 6

Conclusion and Future Work

6.1 Discussion and Conclusion

In this work, we investigate how fairness can be modeled in group recommendations in the health domain. Specifically, our focus was in producing recommendations for items related to the health domain. As such we proposed a new similarity function, named *SemS*, that takes into account information provided by a patient's profile. As in modern Personal Health Systems, patient information is represented using standard terminologies, in this work, as a proof of concept, we employee the ICD10 ontology. Our experiments confirm that indeed, for the health domain, our proposed similarity function gives better results than traditional similarity functions based on ratings, such as the Pearson correlation. It is also worth mentioning that because the *SemS* similarity function is ontology independent, it can be applied to different domains, according to which ontology we use.

We proceed even further, to explore and compare four different aggregation methods – Minimum, Average, Borda and our own contribution in the field, the Fair method. The experiments performed demonstrate the good behavior of our solution with respect to its target, i.e., to increase the fairness and utility of the suggested results. To achieve better results, we sacrifice execution time as our method requires slightly more complex calculations. However, still we remain into the same order of magnitude and the execution time is in the order of milliseconds.

A side-effect of our work is that we also generated the first chimeric synthetic dataset (and the corresponding data generator tool) in the area, a dataset combining chimeric patient profiles and interesting documents. We believe that such a dataset is completely missing from the field, paving the way for more work in the area.

6.2 Limitations of this work

The semantic similarity measure proposed, makes the assumption that the health information of a patient is captured using standard terminologies. Although this is common practice in nowadays, there is still a lot of textual information in such systems not always mapped to standard terminologies. Nevertheless, today there exist many tools that annotate effectively textual descriptions to terminological terms. As an example, the Bioportal Annotator[1] exposes programmatically an API for annotating textual information with multiple terminologies. An extension of our work could use this API to annotate textual descriptions as well. The same assumption hold for the interesting documents recommended to the patients.

6.3 Future Work

For the continuation of this work, we could focus on minimizing the execution time of the *SemS* algorithm. Also, further studies will be needed in the Fair algorithm, to make its results far more challenging in comparison to that of Borda.

Because, our proposed system will be used by people that probably suffer from severe health problems – like cancer, their recommended documents have to be of high quality, and not documents that their approval ratings were manufacture from malicious users (i.e., users that give ratings to documents that do not reflect their quality). To counter those users, further studies will be needed into the trustworthiness of the users (e.g., [40]). Although we acknowledge the fact that for our target audience, an abuse of the system will have serious backlash on them, an extra layer of filtering can be applied in our recommendation process, that will safeguard from malicious users. As such, our system will be able to warranty the best possible results.

Additionally, we can introduce constraints and context into the recommendation process. Constraints can pertain to the individual members of the group (e.g., a user does not want to see any recommendations about herpes), or the group as a singularity (e.g., the group wants to focus on documents about cancer). In turn, context in our approach may be temporal (e.g., take into account the time of year, so that we can offer recommendations on season allergies) or we can take into account the severity of the problems of a user (i.e., the user profile) and present a more tailored recommendations list, but without disadvantaging the rest of the group (the fairness of the recommendations list must be preserved).

Moreover, our implementation could be complemented with execution traces, showing details and explanations on exactly how and why an item was recommended. For instance, an item i was recommended because it was high on the x group member's individual list, and it was in that place because it was recommended by num peers of x.

Finally, we have to note that our approach will be incorporated in the iPHR system of the iManageCancer EU project [26] for enabling recommendations to high-quality web resources in the cancer domain. Those web resources are selected by domain experts and besides automatic recommendations, we will provide also a search engine for enabling querying over the available corpus. In this search

[1] `https://bioportal.bioontology.org/annotator`

engine, the recommendations provided will exploit besides patient profiles, input queries. Those will be annotated as well using ICD10, and as a consequence, the terms in user profile will be extended by adding query terms. Another idea we plan to explore in this direction is to enable data aging as problems move in the past, and consider only the latest and chronic ones.

6.4 Acknowledgements

This work was partially supported by the EU project iManageCancer (H2020, #643529).

Bibliography

[1] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, and Alexander Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.*, 23(1):103–145, 2005.

[2] Giuseppe Agapito, Barbara Calabrese, Pietro Hiram Guzzi, Mario Cannataro, Mariadelina Simeoni, Ilaria Care, Theodora Lamprinoudi, G. Fuiano, and A. Pujia. DIETOS: A recommender system for adaptive diet monitoring and personalized food suggestion. In *2th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2016, New York, NY, USA, October 17-19, 2016*, pages 1–8, 2016.

[3] Nitin Agarwal, Ehtesham Haque, Huan Liu, and Lance Parsons. Research paper recommender systems: A subspace clustering approach. In *Advances in Web-Age Information Management, 6th International Conference, WAIM 2005, Hangzhou, China, October 11-13, 2005, Proceedings*, pages 475–491, 2005.

[4] Charu C. Aggarwal. *Recommender Systems - The Textbook*. Springer, 2016.

[5] Sihem Amer-Yahia, Senjuti Basu Roy, Ashish Chawla, Gautam Das, and Cong Yu. Group recommendation: Semantics and efficiency. *PVLDB*, 2(1):754–765, 2009.

[6] Liliana Ardissono, Anna Goy, Giovanna Petrone, Marino Segnan, and Pietro Torasso. Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence*, 17(8-9):687–714, 2003.

[7] Marko Balabanović and Yoav Shoham. Fab: Content-based, collaborative recommendation. *Commun. ACM*, 40(3):66–72, March 1997.

[8] Linas Baltrunas and Xavier Amatriain. Towards time-dependant recommendation based on implicit feedback. In *In Workshop on context-aware recommender systems*, 2009.

[9] Anika Batenburg and Enny Das. Emotional approach coping and the effects of online peer-led support group participation among patients with breast cancer: A longitudinal study. *J Med Internet Res*, 16(11):e256, Nov 2014.

[10] Gina M. Berg, Ashley M. Hervey, Dusty Atterbury, Ryan Cook, Mark Mosley, Raymond Grundmeyer, and David Acuna. Evaluating the quality of online information about concussions. *Journal of the American Academy of PAs*, 27:1547–1896, 2014.

[11] John S. Breese, David Heckerman, and Carl Myers Kadie. Empirical analysis of predictive algorithms for collaborative filtering. *CoRR*, abs/1301.7363, 2013.

[12] Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors. *The Adaptive Web, Methods and Strategies of Web Personalization*, volume 4321 of *Lecture Notes in Computer Science*. Springer, 2007.

[13] Derek Yee Tak Cheung, Helen Ching Han Chan, Jonah Chi-Keung Lai, Vivian Wai Fung Chan, Ping Man Wang, William Ho Cheung Li, Chee Sophia Siu Chan, and Tai-Hing Lam. Using whatsapp and facebook online social groups for smoking relapse prevention for recent quitters: A pilot pragmatic cluster randomized controlled trial. *J Med Internet Res*, 17(10):e238, Oct 2015.

[14] Andrew Crossen, Jay Budzik, and Kristian J. Hammond. Flytrap: intelligent group music recommendation. In *Proceedings of the 7th International Conference on Intelligent User Interfaces, IUI 2002, San Francisco, California, USA, January 13-16, 2002*, pages 184–185, 2002.

[15] Persi Diaconis and R. L. Graham. Spearman's footrule as a measure of disarray. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(2):262–268, 1977.

[16] Yi Ding and Xue Li. Time weight collaborative filtering. In *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*, pages 485–492, 2005.

[17] Mehdi Elahi, Francesco Ricci, and Neil Rubens. A survey of active learning in collaborative filtering recommender systems. *Computer Science Review*, 20:29–50, 2016.

[18] Peter Emerson. The original borda count and partial voting. *Social Choice and Welfare*, 40(2):353–358, 2013.

[19] F. S. Gohari and M.J. Tarokh. Classification and comparison of the hybrid collaborative filtering systems. *International Journal of Research in Industrial Engineering*, 4(2):129–148, 2017.

[20] Sandy El Helou, Christophe Salzmann, and Denis Gillet. The 3a personalized, contextual and relation-based recommender system. *J. UCS*, 16(16):2179–2195, 2010.

[21] Santiago Hors-Fraile, Francisco J. Núñez-Benjumea, Laura Carrasco Hernández, Francisco Ortega Ruiz, and Luis Fernández-Luque. Design of two combined health recommender systems for tailoring messages in a smoking cessation app. *CoRR*, abs/1608.07192, 2016.

[22] Brandon Irwin, Daniel Kurz, Patrice Chalin, and Nicholas Thompson. Testing the efficacy of ourspace, a brief, group dynamics-based physical activity intervention: A randomized controlled trial. *J Med Internet Res*, 18(4):e87, May 2016.

[23] Uri Kartoun. A methodology to generate virtual patient repositories. *CoRR*, abs/1608.00570, 2016.

[24] Maurice G. Kendall. *Rank correlation methods.* Griffin, London, 1948.

[25] Jong-Hun Kim, Daesung Lee, and Kyung-Yong Chung. Item recommendation based on context-aware model for personalized u-healthcare service. *Multimedia Tools Appl.*, 71(2):855–872, 2014.

[26] Haridimos Kondylakis, Anca Bucur, Feng Dong, Chiara Renzi, Andrea Manfrinati, Norbert Graf, Stefan Hoffman, Lefteris Koumakis, Gabriella Pravettoni, Kostas Marias, Manolis Tsiknakis, and Stephan Kiefer. imanagecancer: Developing a platform for empowering patients and strengthening self-management in cancer diseases. In *IEEE CBMS*, 2017.

[27] Haridimos Kondylakis, Lefteris Koumakis, Eleni Kazantzaki, Maria Chatzimina, Maria Psaraki, Kostas Marias, and Manolis Tsiknakis. Patient empowerment through personal medical recommendations. In *MEDINFO 2015: eHealth-enabled Health - Proceedings of the 15th World Congress on Health and Biomedical Informatics, São Paulo, Brazil, 19-23 August 2015*, page 1117, 2015.

[28] Haridimos Kondylakis, Lefteris Koumakis, Maria Psaraki, Georgia Troullinou, Maria Chatzimina, Eleni Kazantzaki, Konstantinos Marias, and Manolis Tsiknakis. Semantically-enabled personal medical information recommender. In *Proceedings of the ISWC 2015 Posters & Demonstrations Track co-located with the 14th International Semantic Web Conference (ISWC-2015), Bethlehem, PA, USA, October 11, 2015.*, 2015.

[29] Joseph A. Konstan, Bradley N. Miller, David Maltz, Jonathan L. Herlocker, Lee R. Gordon, and John Riedl. Grouplens: Applying collaborative filtering to usenet news. *Commun. ACM*, 40(3):77–87, 1997.

[30] Xiaohui Li and Tomohiro Murata. Using multidimensional clustering based collaborative filtering approach improving recommendation diversity. In *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Macau, China, December 4-7, 2012*, pages 169–174, 2012.

[31] Xiao Lin, Min Zhang, Yongfeng Zhang, Zhaoquan Gu, Yiqun Liu, and Shaoping Ma. Fairness-aware group recommendation with pareto-efficiency. In *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017, Como, Italy, August 27-31, 2017*, pages 107–115, 2017.

[32] Miriam McMullan. Patients using the internet to obtain health information: How this affects the patient-health professional relationship. *Patient Education and Counseling*, 63(1):24 – 28, 2006.

[33] Jingbo Meng, Wei Peng, Yun Soo Shin, and Minwoong Chung. Online self-tracking groups to increase fruit and vegetable intake: A small-scale study on mechanisms of group effect on behavior change. *J Med Internet Res*, 19(3):e63, Mar 2017.

[34] Raymond J. Mooney and Loriene Roy. Content-based book recommending using learning for text categorization. In *ACM DL*, pages 195–204, 2000.

[35] Martín López Nores, Yolanda Blanco-Fernández, José Juan Pazos-Arias, and Alberto Gil-Solla. Property-based collaborative filtering for health-aware recommender systems. *Expert Syst. Appl.*, 39(8):7451–7457, 2012.

[36] Eirini Ntoutsi and Kostas Stefanidis. Recommendations beyond the ratings matrix. In *Proceedings of the Workshop on Data-Driven Innovation on the Web, DDI@WebSci 2016, Hannover, Germany, May 22-25, 2016*, pages 2:1–2:5, 2016.

[37] Eirini Ntoutsi, Kostas Stefanidis, Kjetil Nørvåg, and Hans-Peter Kriegel. Fast group recommendations by applying user clustering. In *Conceptual Modeling - 31st International Conference ER 2012, Florence, Italy, October 15-18, 2012. Proceedings*, pages 126–140, 2012.

[38] Eirini Ntoutsi, Kostas Stefanidis, Katharina Rausch, and Hans-Peter Kriegel. "strength lies in differences": Diversifying friends for recommendations through subspace clustering. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 729–738, 2014.

[39] Cosimo Palmisano, Alexander Tuzhilin, and Michele Gorgoglione. Using context to improve predictive modeling of customers in personalization applications. *IEEE Trans. Knowl. Data Eng.*, 20(11):1535–1549, 2008.

[40] Manos Papagelis, Dimitris Plexousakis, and Themistoklis Kutsuras. Alleviating the sparsity problem of collaborative filtering using trust inferences. In *Trust Management, Third International Conference, iTrust 2005, Paris, France, May 23-26, 2005, Proceedings*, pages 224–239, 2005.

[41] Michael J. Pazzani and Daniel Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331, 1997.

[42] Daniel Pfeifer. Health recommender systems: Concepts, requirements, technical basics and challenges. *International Journal of Environmental Research and Public Health*, 11(3):2580–2607, 2014.

[43] Shuyao Qi, Nikos Mamoulis, Evaggelia Pitoura, and Panayiotis Tsaparas. Recommending packages to groups. In *IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain*, pages 449–458, 2016.

[44] Shuyao Qi, Nikos Mamoulis, Evaggelia Pitoura, and Panayiotis Tsaparas. Recommending packages to groups. In *IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain*, pages 449–458, 2016.

[45] Senjuti Basu Roy, Sihem Amer-Yahia, Ashish Chawla, Gautam Das, and Cong Yu. Space efficiency in group recommendation. *VLDB J.*, 19(6):877–900, 2010.

[46] Neil Rubens, Mehdi Elahi, Masashi Sugiyama, and Dain Kaplan. Active learning in recommender systems. In *Recommender Systems Handbook*, pages 809–846. 2015.

[47] Dimitris Sacharidis. Group recommendations by learning rating behavior. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, UMAP 2017, Bratislava, Slovakia, July 09 - 12, 2017*, pages 174–182, 2017.

[48] Jeff J. Sandvig, Bamshad Mobasher, and Robin D. Burke. A survey of collaborative recommendation and the robustness of model-based algorithms. *IEEE Data Eng. Bull.*, 31(2):3–13, 2008.

[49] Hanna Schäfer, Santiago Hors-Fraile, Raghav Pavan Karumur, André Calero Valdez, Alan Said, Helma Torkamaan, Tom Ulmer, and Christoph Trattner. Towards health (aware) recommender systems. In *Proceedings of the 2017 International Conference on Digital Health, London, United Kingdom, July 2-5, 2017*, pages 157–161, 2017.

[50] Kostas Stefanidis, Eirini Ntoutsi, Mihalis Petropoulos, Kjetil Nørvåg, and Hans-Peter Kriegel. A framework for modeling, computing and presenting time-aware recommendations. *Trans. Large-Scale Data- and Knowledge-Centered Systems*, 10:146–172, 2013.

[51] Kostas Stefanidis and Evaggelia Pitoura. Finding the right set of users: Generalized constraints for group recommendations. *CoRR*, abs/1302.6580, 2013.

[52] Julia Stoyanovich, Serge Abiteboul, and Gerome Miklau. Data responsibly: Fairness, neutrality and transparency in data analysis. In *Proceedings of the 19th International Conference on Extending Database Technology, EDBT 2016, Bordeaux, France, March 15-16, 2016, Bordeaux, France, March 15-16, 2016.*, pages 718–719, 2016.

[53] Maria Stratigi, Haridimos Kondylakis, and Kostas Stefanidis. Fairness in group recommendations in the health domain. In *33rd IEEE International Conference on Data Engineering, ICDE 2017, San Diego, CA, USA, April 19-22, 2017*, pages 1481–1488, 2017.

[54] Katrien Verbert, Erik Duval, Stefanie N. Lindstaedt, and Denis Gillet. enisontext-aware recommender systems. *J. UCS*, 16(16):2175–2178, 2010.

[55] Martin Wiesner and Daniel Pfeifer. Adapting recommender systems to the requirements of personal health record systems. In *ACM International Health Informatics Symposium, IHI 2010, Arlington, VA, USA, November 11 - 12, 2010, Proceedings*, pages 410–414, 2010.

[56] Liang Xiang and Qing Yang. Time-dependent models in collaborative filtering based recommender system. In *2009 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2009, Milan, Italy, 15-18 September 2009, Main Conference Proceedings*, pages 450–457, 2009.

[57] Liang Xiang, Quan Yuan, Shiwan Zhao, Li Chen, Xiatian Zhang, Qing Yang, and Jimeng Sun. Temporal recommendation on graphs via long- and short-term preference fusion. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*, pages 723–732, 2010.