Aspect-based Recommendation

by

Maryam Mirzaei

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computing Science

University of Alberta

Abstract

The problem of aspect-based recommendation—recommending an 'item' to a 'recommendation recipient' based on "aspects", i.e., information about the characteristic features of the item that may be of interest to the recommendation recipient or what makes an item a good match for a recommendation recipient—is attracting substantial research attention recently.

In this work, we study two problems in this general area.

First, we consider the problem of assigning reviewers to papers submitted for publication to a conference. We cast this problem as the recommendation of a set of experts as appropriate reviewers for a paper. Papers in this case correspond to "recommendation recipient", and we consider the thematic areas or topics of a paper as their "aspects". Potential reviewers correspond to "items", and we consider the expertise areas of reviewers when considering the importance of the papers aspects . The paper aspects can be inferred from terms extracted from the paper description (title and abstract); the reviewer's expertise can similarly be extracted from the descriptions of the papers they have authored. Our reviewer-recommendation algorithm assigns to each submitted paper a set of reviewers who can evaluate all aspects of the paper, while at the same time, maximizing the relevant expertise of the reviewers and balancing their workload.

Next, we consider the problem of personalized and explainable aspect-based recommendations of products and services based on online reviews. Our algorithm recommends items to users by capturing the dependencies between the

sentiments that reviews express towards different item aspects, and using the importance of these aspects for each target user. In this scenario, the algorithm effectively predicts the user's sentiments toward candidate item aspects, and uses these predicted sentiments as de-facto explanations for the items it selects to recommend.

In all stages of our work we experimentally validate our methods on a variety of datasets from different domains and we experimentally demonstrate its superior performance relative to other state-of-the-art approaches.

Preface

This research included in this thesis is my original work. Chapter 2 has been published as the following journal paper:

- Mirzaei Maryam, Jörg Sander, and Eleni Stroulia. "Multi-aspect review-team assignment using latent research areas." Information Processing & Management 56.3 (2019): 858-878.

Chapter 3 has been submitted as the journal paper and currently under review.

Dedicated to Mohammad, Ali, My Dad and My Mom

For teaching me a great many things about the life and for helping me to finish my PhD.

Yesterday I was clever, so I wanted to change the world. Today I am wise, so I am changing myself.

– Jalaleddeen Muhammad Balkhi (Rumi)

Acknowledgements

I would like to express my gratitude

- To Jörg and Eleni: For their help and guidance during my years as a PhD student; I learned so much from you, and can't thank you enough.
- To Mohammad: For companionship, support and the gift of love and for giving me renewed energy and propose when I felt depleted along this way, reminding me of what life is about;
- To Ali: For all his moments of patience, for bearing with my inadequate presence and for giving me reason to go on;
- To my mom: For supporting me always and for my independence;
- To my dad: For caring for me, for teaching me what to fight for and when to leave, for helping me build my world and for understanding that world in the true sense of the word understanding;
- To my friends especially Elaheh: For supporting me from afar, for hearing me and for their love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recently, there has been a great deal of interest in aspect-based recommendations. In aspect-based recommendation, "items" and "recommendation recipients" (in a broad sense) are represented via aspects. Item aspects represent characteristic features of an item, and "recommendation recipient" aspects help characterise what makes an item a good match for a recommendation recipient (e.g., a recommendation recipient's interests, preferences, expertise, etc. w.r.t. those aspects). With such a representation, it is possible to rank, recommend, or assign items or set of items to recommendation recipients in a way that takes into account (1) item aspects, (2) the importance of item aspects to recommendation recipients, and (3) possible constraints that can be expressed in terms of item aspects and their importance. Using aspect information in this way can make recommendations, rankings, and assignments more effective, more targeted, and potentially more explainable.

In this thesis, we develop and study two different scenarios for aspect-based recommendation.

In the first scenario, we use textual artifacts of reviewers such as their published papers to recommend a group of reviewers to each submitted paper in research-related activities (such as conference organization and grant-proposal adjudication) based on different aspects of the submitted paper. We cast this

problem as the recommendation of a set of experts as the appropriate reviewers for the paper, and we consider as "aspects" the thematic areas of the paper. These aspects are the research areas of the paper that reviewers may have different expertise on them.

The paper aspects can be inferred from terms extracted from the paper description (title and abstract); the reviewer's research areas can similarly be extracted from the descriptions of the papers they have authored. For example, a paper proposing a 'cluster-based information-retrieval algorithm for email texts' is talking about two aspects of 'clustering' and 'information retrieval'. And a good assignment for this paper would be a group of reviewers which have expertise on both aspects.

In the second scenario, we show how to leverage reviews from recommendation sites to recommend items to users based on item aspects expressed in the user reviews, i.e., characteristic features of items that users may weigh differently in their judgement of items. In some review systems, item aspects may be explicitly pre-defined and users are asked to rate those aspects individually. For instance, in case of hotel recommendations, a user may be asked to rate a hotel explicitly by rating the quality of pre-defined aspects such as "Room", "Service" and "Affordability".

On other sites item aspects not predefined and typically only free-form review text (and an overall rating) are available. In these cases, aspects can be extracted from the review texts as part of the recommendation process, by extracting sequences of words which describe specific attributes of items, which are mentioned in many reviews. For example, in reviews of mobile phones, aspects such as "Battery Life", "Screen Quality", "Noise" and "Affordability" may typically be mentioned.

1.1 Multi-aspect Paper-Reviewer Team Assignment

First, in Chapter 2, we examine the problem of assigning a team of reviewers to papers according to their research areas. We cast this problem as the recommendation of a set of reviewers for a paper. We consider as "aspects" the thematic areas of the paper that reviewers may have different expertise on them. The thematic areas of a paper, effectively the themes of the paper, can be inferred from terms extracted from the paper's description, including its title and abstract. A reviewer's research areas, i.e., their area of expertise, can similarly be extracted from the descriptions of the papers they have authored.

Papers typically bring together several different aspects, i.e., ideas from different bodies of knowledge, to develop their contributions. Moreover, the importance of different aspects of a paper can vary depending on the research area to which the paper belongs. Thus, the problem of "Multi-aspect paper-reviewer assignment" has been formulated aiming at optimizing three properties: (a) The expertise of each individual reviewer assigned to a paper should cover as many knowledge aspects of the paper as possible (coverage), (b) The overall expertise of the team of reviewers assigned to each paper should cover as many knowledge aspects of the paper as possible (confidence), and (c) Each reviewer should be assigned at most a defined number of papers to review (reviewer's quota).

Previously, Karimzadehgan et al. [25] proposed an algorithm based on *integer linear programming (ILP)* to maximize the number of assigned reviewers that can cover a paper's topics, for each paper, subject to paper quota and reviewer quota constraints. Neshati et al. [36] also cast the constrained multi-aspect reviewer assignment problem into a *capacitated facility-location analysis problem (CFLA)*. To solve this problem, they propose an integer linear-

programming formulation to choose the reviewer for a paper which simultaneously satisfies confidence and coverage maximization.

Both of the above methods consider each paper–reviewer assignment independently, ignoring the synthesis of the team of reviewers. In the independent reviewer assignment approach, one reviewer is chosen for a paper without considering the similarities and differences between the new reviewer and reviewers already assigned to this paper.

More recently, Kou et al. [26] proposed a formulation of the problem to consider the coverage of a paper's topics by *the expertise of the group of reviewers* assigned to it. They also assume that each topic has a different importance and propose a Weighted-coverage Group-based Reviewer Assignment.

All the previous works mentioned above use different topic modeling methods to represent the expertise of reviewers and the content of the papers. However, the accuracy of topic modeling methods based on the small collection of short documents is not reliable. Moreover, the full text of publications of reviewers is also not easily accessible.

To improve the shortcomings of the previous works, we propose a group-based reviewer assignment method that uses a term space representation for reviewers and papers. Since all terms are not equally important across different research areas, we estimate the importance of each term within a specific area of research. To that end, we present an approximation algorithm called *Multi-Aspect Review-Team Assignment using Latent Research Areas (MARTA-LRA)* using a greedy forward-selection strategy to select a group of reviewers for a paper. MARTA-LRA introduces a single efficient framework to solve both unconstrained and constrained problem variants. This framework defines a new objective function for group-based paper–reviewer assignment in term space. It also considers different important scores for the terms of each submitted paper and for the terms of each reviewer's expertise, when computing the

relevance of a particular reviewer for a paper.

We demonstrate the effectiveness of our method by applying it to two datasets. The results demonstrate that our objective function, MARTA-LRA, considerably improves the performance of multi-aspect reviewer assignment, in both constrained and unconstrained settings, over the state-of-the-art related works.

1.2 Personalized and Explainable Aspect-based Recommendation of an item to a user

The second question we study in this thesis, in Chapter 3 is personalized and explainable aspect-based recommendation of an item to a user. Recommender systems learn the users preferences and entice them with more offerings that are potentially to their liking. To make a recommendation more trustworthy, and potentially more persuasive, users can be provided with a reason *why* a particular item is recommended to them.

An explanation for a recommendation gives "reasons" for why the recommender system offers a specific item. Different categories of "reasons" can be distinguished to explain recommending an item to a user. The reasoning can be provided (1) based on the similarity of other users to the target user (*similar-user-based*), (2) based on the similarity to items rated previously by the target user (*similar-item-based*), (3) based on the quality of objective attributes of the recommended item such as the lens of a camera (*feature-based*), or (4) based on the rating of item aspects such as "affordability" of a hotel (*aspect-based*).

Similar-user-based explanations tend to not be very convincing since the users receiving the recommendations often know nothing about users that are "similar" to them [20]. Furthermore, similar-user-based explanations [57] tend to weigh past reviews based on the similarity between the review authors

and the target user; reviews by different users that have the same degree of similarity to the target user are likely to receive similar weight, even though the reviews for a particular item may discuss different aspects of the item. Similar-item-based explanations are usually more intuitively understandable for a user, but do not necessarily give details about *how* the recommended item is similar to what the user liked previously [57]. Feature-based explanations (like the similar-item-based explanations) do not use any previous reviews of the user to provide explanations for recommendation, which can limit their persuasiveness.

To improve on these shortcomings, we focus on aspect-based explanations, taking into account the user's preferences as exemplified by their previous item reviews, to explain why the item is recommended based on the recommended item's aspects.

Recently, a number of aspect-based recommendation methods [6, 22, 32] have been proposed. These methods use user-generated content, which expresses opinions about items and their various aspects, to predict the overall rating of an item for a user.

However, several problems in aspect-based recommendation problem still remain open:(a) Improve the identification of users preferences based on their reviews, by capturing semantic similarities between the different words that reviewers may use when talking about an aspect. (b) Providing any explanation for a recommendation in terms of aspects (instead of only using aspect information to predict an overall rating of an item for a user). (c) Considering the fact that not all users pay attention to all aspects in the same way. (d) Considering the dependency between the sentiments towards item aspects.

To move toward a solution to the problems above, we first predict the sentiments that a user expresses towards an item's aspects and then combine these predicted sentiments to develop an overall rating for each item. The top-rated

candidate items are then recommended and the predicted sentiment toward each of their aspects constitute the explanation for these recommendations.

To do so, we first enhance item reviews by adding the aspects and sentiments mentioned in these "raw" reviews, either using a state-of-the-art aspect-extraction method, the opinion parser [38] or explicitly provided aspect ratings.

Next, these enhanced reviews are used to construct word-embedded representations for aspects, users, and items. These representations are used alongside the explicit aspect ratings to provide the basis for estimating the importance score of different aspects for a user for an item. Furthermore, using word-embedded representations to estimate these importance scores enables our method to handle a variety of semantically similar words that reviewers may use when talking about an item's aspect.

We also group reviews of an item into groups that have expressed the same sentiments towards a *set* of aspects in order to better address the dependencies between sentiments expressed towards different aspects. This information about groups of reviews together with the aspect importance scores calculated for a user for an item are used to predict the sentiments expressed toward an item's aspects for a user, as well as the overall rating of a user for an item.

We have evaluated the recommendation performance and the quality of explanation of our method on a variety of datasets from different domains and we experimentally demonstrate its superior performance relative to other state-of-the-art approaches.

Chapter 2

Multi-Aspect Review-Team Assignment using Latent Research areas

Abstract

Reviewer assignment is an important task in many research-related activities, such as conference organization and grant-proposal adjudication. The goal is to assign each submitted artifact to a set of reviewers who can thoroughly evaluate all aspects of the artifact's content, while, at the same time, balancing the workload of the reviewers. In this paper, we focus on textual artifacts such as conference papers, where both (aspects of) the submitted papers and (expertise areas of) the reviewers can be described with terms and/or topics extracted from the text. We propose a method for automatically assigning a team of reviewers to each submitted paper, based on the clusters of the reviewers' publications as latent research areas. Our method extends the definition of the relevance score between reviewers and papers using the latent research areas information to find a team of reviewers for each paper, such that each individual reviewer and the team as a whole cover as many paper aspects as possible. To solve the constrained problem where each reviewer has a limited reviewing capacity, we utilize a greedy algorithm that starts with a group of

reviewers for each paper and iteratively evolves it to improve the coverage of the papers' topics by the reviewers' expertise. We experimentally demonstrate that our method outperforms state-of-the-art approaches w.r.t several standard quality measures.

2.1 Introduction

Reviewing is a key activity in scholarly work and the assignment of expert reviewers is a critical task of conference organizers, journal editors, and grant-proposal adjudication committees. For most conferences today, organizers assign reviewers manually, typically matching conference-defined keywords, assigned by authors to their papers at submission time, and selected by reviewers to characterize their expertise profile in the conference-management system. This process is inefficient, especially when the number of submissions is large. Even more importantly, such a process is fraught with challenges and can result in poor assignments. The keywords with which the conference-management system is configured are unlikely to cover the topics of all submissions and the expertise of all reviewers: the less meaningful these keywords are for a submission, the more likely it becomes selected by the reviewers less capable of assessing its contributions.

These challenges represent a research opportunity: if the reviewer-assignment process could be supported through the use of an automated system, assignment quality might improve considerably. Indeed, automatic reviewer assignment has been investigated in previous works such as [14, 5, 21, 34]. Most of these methods consider the paper–reviewer assignment as an information-retrieval problem: each paper is considered as a query that should be matched with the most similar documents, representing the reviewers' expertise based on evidence found in the reviewers' publications.

Papers typically bring together several different knowledge aspects, i.e.,

ideas from different bodies of knowledge, to develop their contributions. Consider, for example, a paper proposing a 'cluster-based information-retrieval algorithm for email texts': a good assignment should include reviewers with expertise on 'clustering' and 'information retrieval'. If the program committee does not include any reviewers with expertise on both topics, it is preferable to assign the paper to an expert on 'clustering' and a second expert on 'information retrieval', as opposed to two experts on 'clustering'. If, on the other hand, there are multiple reviewers with expertise on both terms, it is preferable to select two among them with diverse expertise profiles. Finally, we claim that the importance of different knowledge aspects of a paper can vary, depending on the research area to which the paper belongs. In case of a paper about a cluster-based information-retrieval algorithm for email texts, the knowledge aspect 'information retrieval' is likely more important than 'clustering' in describing the content of the paper, and should thus have a higher weight than 'clustering' when trying to match the paper representation with reviewer profiles.

Considering a paper as a single cohesive unit, without paying attention to its multiple knowledge aspects, as Mimno and McCallum [34] do, is bound to lead to poor reviewer assignments.

Recognizing this pitfall, Karimzadehgan *et al.* [25] formulated the problem as *multi-aspect* paper-reviewer assignment and proposed a method to address it, aiming at optimizing three properties.

1. *Confidence*: The expertise of each individual reviewer assigned to a paper should cover as many knowledge aspects of the paper as possible.

2. *Coverage*: The overall expertise of the team of reviewers assigned to each paper should cover as many knowledge aspects of the paper as possible.

3. *Load balancing*: Each paper should receive a predefined number of re-

views (paper's quota), and each reviewer should be assigned at most a defined number of papers (reviewer's quota) to review.

Two different problem variants can be formulated based on these challenges: the *Unconstrained Multi-Aspect Review-Team Assignment* (Unconstrained MARTA) variant considers only coverage and confidence, and the *Constrained Multi-Aspect Review-Team Assignment* (Constrained MARTA) variant considers all of the above objectives.

Karimzadehgan *et al.* [25] and Neshati *et al.* [36] consider each paper-reviewer assignment independently, ignoring the synthesis of the team of reviewers, and, as competent reviewers reach their quota, papers may end up being assigned to non-knowledgeable reviewers (low confidence of expertise). In the independent reviewer assignment approach, one reviewer is chosen for a paper without considering the similarities and differences between the new reviewer and reviewers already assigned to this paper.

More recently, Kou *et al.* [26] proposed yet another formulation of the problem to consider the coverage of a paper's topics by *the expertise of the group of reviewers* assigned to it as Weighted-coverage Group-based Reviewer Assignment.

In all the previous works mentioned above, the expertise of a reviewer and the contents of a paper are represented by a set of topics using different topic modeling methods. However, the accuracy of topic modeling methods based on the small collection of short documents is not reliable (only the title and the abstract of the papers and publications of reviewers are available). The full text of publications of reviewers is also not easily accessible. To improve these aspects of previous works, we introduce in this paper a group-based reviewer assignment method that uses a term space representation for reviewers and papers. However, not all terms are equally meaningful across

a scientific domain: some terms are more important than others, and their relative importance changes across different research areas and communities. The importance of each term, within a specific area of research, can be a valuable source of information in the paper-reviewer assignment problem.

To use this valuable source of information, our method clusters the reviewers' publications into what could be intuitively conceived as "research areas". Technically, each latent research area is represented using the concatenation of the terms of the publications that belong to the cluster. Our method uses these research areas to weigh the terms of each submitted paper and the terms of each reviewer's expertise, when computing the relevance of a particular reviewer for a paper; we call this quality score function "Multi-Aspect Review-Team Assignment using Latent Research Areas (MARTA-LRA)". As finding the exact solution for this problem is infeasible due to its large search space, we use a greedy forward-selection strategy as an approximation algorithm that selects reviewers for each paper in order to maximize the MARTA-LRA objective function, while examining the number of reviewers per paper and the reviewers' work load constraint in each step.

We demonstrate the effectiveness of our method by applying it to two datasets: (a) the dataset created by Karimzadehgan *et al.* [25] using ACM SIGIR publications from years 1971–2007 and (b) the dataset created by Neshati *et al.* [36] using the PubMed database. Our method outperforms the methods by Karimzadehgan *et al.*, Neshati *et al.* and Kou *et al.*. In summary, our work makes two key contributions.

- It demonstrates how information about the latent research areas of the program-committee's publications improves the quality of the multi-aspect review-team assignment, in both the unconstrained and constrained variants of the problem.

- It defines a new objective function for group-based paper-reviewer assignment in term space. This objective function considers the expertise of each reviewer, the overall expertise of the review team, as well as the diversity of reviewers' expertise.

- It introduces a single efficient framework to solve both unconstrained and constrained problem variants.

The remainder of this paper is organized as follows. In Section 3.2, we review the relevant previous works on the problem of multi-aspect review paper assignment. Then, our proposed method is introduced in detail in Section 3.3. Section 3.4 is devoted to reporting experimental designs and results. Conclusions are drawn in Section 3.5.

2.2 Related work

Several methods have been proposed for automating or supporting the *paper-reviewer assignment problem*. The methods can be categorized by how they approach the problem.

2.2.1 Single-aspect Independent Paper-reviewer Assignment in Term Space

Most of these methods cast the problem as an information retrieval (IR) problem, with the objective of retrieving the documents (reviewers) that best match a set of given queries (submitted papers).

Basu *et al.* [5] construct the expertise profile of each reviewer by concatenating her publications, and determining the similarity of the reviewer's profile to the query paper according to the TF–IDF score. Hettich and Pazzani [21] examine the problem of recommending panels of reviewers for NSF grant proposals, and use the reviewer's past proposals to construct her profile. The

grant proposals and reviewers are represented in the standard TF–IDF vector space [10]. A greedy approach is used to find the best reviewer for each paper based on the similarity between the TF–IDF scores of the terms of the paper and the selected reviewers in each iteration. Biswas and Hasan [7] represent papers and reviewers using topics from a domain ontology, and rank the reviewers with respect to papers using their TF–IDF similarity. Mimno and McCallum [34] propose the author-persona-topic (APT) model which divides the authors' papers into many "personas". Each persona is represented as a mixture of hidden topics learned using statistical topic-modeling methods. The language-modeling method is used to match then reviewers with papers.

In addition to all the specific differences mentioned above, none of the above methods consider covering multiple aspects of a paper in the process of choosing reviewers.

2.2.2 Multi-aspect Independent Paper-reviewer Assignment in Topic Space

Karimzadehgan *et al.* [25] proposed three strategies to address the unconstrained multi-aspect paper-review assignment problem: redundancy removal, reviewer aspect modeling, and, paper aspect modeling. The *redundancy removal strategy* involves a greedy algorithm that incrementally substitutes reviewers of a paper with new ones that are more relevant to the paper and less redundant relative to the other reviewers assigned to the paper.

The *reviewer aspect modeling* method focuses on extracting different aspects of the reviewers' expertise, using probabilistic latent semantic analysis (PLSA). Then, the reviewers are matched to papers, based on these aspects (i.e., latent-topic representations), one at a time. In this greedy approach, a reviewer that minimizes the following objective function is added in iteration

k to form an optimal skill-covering group:

$$D(\theta_p||(\frac{\sigma}{n-1}\sum_{i=1}^{n-1}\theta_i + (1-\sigma)\theta_n))\,, \tag{2.1}$$

where θ_i indicates the topic representation for the ith chosen reviewer; θ_p is the topic vector of the paper; $D(\cdot||\cdot)$ is the KL-divergence between two distributions p and q; and, σ is a parameter that controls how much the set of previously selected reviewers is relied upon to cover all aspects of the paper.

In *paper aspect modeling*, the aspects extracted from a paper are used as queries to retrieve relevant reviewers. If a paper needs k reviewers, the paper is partitioned into k aspects. For each aspect, the top reviewers are retrieved using the query likelihood model with Dirichlet smoothing [56]. To partition each paper, two methods are proposed:

- Word clustering, based on mutual information between each pair of words (MIC); and,
- Text segmentation, based on cosine similarity of the TF–IDF weight vectors of each sentence in the paper (SDA).

Of the three methods proposed in [25], the reviewer aspect modeling method is the most effective.

Neshati *et al.* [36] formulated the unconstrained multi-aspect paper-reviewer assignment problem as an *uncapacitated facility location analysis* (UFLA). In this framework, reviewers are considered as facilities and paper topics are considered as customers. The goal of UFLA is to minimize the sum of the cost of building k facilities plus the communication cost, namely the sum of the distances of the customer locations from their closest facilities, weighted by customer demand. A building cost function for reviewers is defined, based on the similarity between reviewers and papers in the topic space. The communication cost between the topics of each paper and a reviewer is estimated

based on the ability of the reviewer to cover the topics. A greedy approach is used to select a group of reviewers for each paper, based on the ability of all k members of the assigned group to cover the required expertise fields of the paper. Equation 2.2 shows the objective function of this greedy algorithm as a combination of the building and communication cost functions.

$$Cost(S, p_j) = \lambda \sum_{i=1}^{k} D(e_i||p_j) + (1-\lambda) \sum_{a=1}^{T} \tau_{aj} min_{s \in S} D(s||v_a)\,, \qquad (2.2)$$

where S is the set of selected reviewers for paper p_j; τ_{aj} indicates the weight of topic a in the topic vector representation of paper p_j; e_i is the ith reviewer and T is the total number of topics for reviewers and papers. Also, v_a is the unit vector with all zero elements except for topic a, and $D(\cdot||\cdot)$ denotes the KL-divergence. The parameter λ balances the building cost and communication cost in this framework. In the algorithm, first, k random reviewers are chosen for each paper. Then, in each iteration, the team of reviewers S for a paper is refined by swapping a reviewer in S with an available non-selected reviewer, according to the cost function of (2.2), until S does not change anymore.

Karimzadehgan *et al.* [24] introduced the problem of constrained multi-aspect paper-reviewer assignment, for which they proposed a solution based on integer linear programming (ILP). The objective is to maximize the number of assigned reviewers that can cover a paper's topics, for each paper, subject to paper quota and reviewer quota constraints. In the same vein, Neshati *et al.* [36] cast the constrained multi-aspect reviewer assignment problem into a *capacitated facility-location analysis problem (CFLA)* and propose an integer linear-programming formulation to solve it, minimizing the following objective function:

$$\sum_{j=1}^{N} \sum_{i=1}^{M} (\lambda U_{ij} BCost(i,j) + (1-\lambda) \sum_{a=1}^{T} CCost(i,j,a) X(i,j,a))\,, \quad (2.3)$$

where N, M and T are the number of papers, reviewers and topics, respectively; U_{ij} is an element of an $N \times M$ binary decision matrix that indicates

the assignment of reviewers to papers; and $X(i, j, a)$ indicates the assignment of topic a of paper p_j to reviewer r_i. The building cost ($BCost$) is the fraction of uncovered topics by the reviewer r_i. $CCost(i, j, a)$ indicates with 0 or 1 whether the reviewer r_i is able to cover the topic a for the paper p_j. Neshati *et al.* [36] also prove that simultaneously minimizing the building-cost and communication-cost functions can satisfy confidence and coverage maximization. The reviewer and paper quotas are defined as constraints in this optimization framework. CFLA outperforms ILP in both coverage and average confidence measures.

Unlike the methods proposed by Karimzadegan *et al.* and Neshati *et al.* that assign reviewers to papers independently, some other methods define a group-based quality measure for the assignment of reviewers to papers [31, 26].

2.2.3 Multi-aspect Group-based Paper-reviewer Assignment in Topic Space

In the independent reviewer assignment approach, one reviewer is chosen for a paper without considering the similarities and differences between the new reviewer and the reviewers already assigned to this paper.

Long *et al.* [31] represent reviewers and papers using a set of topics. Then, the quality of assigning a group of reviewers to a paper is evaluated by the set-coverage ratio, i.e., the ratio of the number of the paper topics covered by the group of reviewers to the number of all of the paper topics. Going a step further, Kou *et al.* [26] assume that each topic has a different importance and propose a weighted coverage ratio, $c(g, p)$ to estimate the quality of an assignment of a group of reviewers g to the paper p:

$$c(g, p) = \frac{\sum_{t=1}^{T} min\{g[t], p[t]\}}{\sum_{t=1}^{T} p[t]}, \tag{2.4}$$

where $g[t]$ and $p[t]$ are the weights of topic t for a group g and paper p accordingly. The weights of topic t for a group of reviewers g is defined, using the

weight of topic t for reviewers r, as follows:

$$g[t] = max_{r \in g}(r[t])\,, \tag{2.5}$$

As mentioned in the introduction, this representation of the aggregate expertise of the group of reviewers assigned to a paper as the maximum coverage by a reviewer in the group can lead to anomalies where if the first chosen reviewer happens to be an expert in all aspects of the paper, the rest of the reviewers in the group can be randomly chosen. Kou *et al.* also propose an approximation algorithm to maximize the weighted coverage score of a group of reviewers chosen for each paper. In this algorithm, they assign exactly one reviewer to each paper in each step, using a minimum-cost flow assignment algorithm [4], called *stage deepening greedy algorithm* (SDGA). To improve the quality of the assigned group of reviewers to each paper, they refine their assignments by substituting a reviewer with lower relevance probability with a new reviewer, identified by the minimum-cost flow assignment algorithm. The relevance probability of a reviewer r and paper p is estimated based on the coverage score of reviewers r and paper p. The *stochastic refinement* method is iteratively applied on the result of SDGA, until the quality of assignment doesn't change for the k rounds. This method is called SDGA-SRA.

2.2.4 Other Similar Information Retrieval Problem

A similar problem is that of *expert finding*, where candidate experts are ranked based on their expertise on a given textual query with a topic of interest. Many methods have been proposed to solve this problem, using language-modeling approaches.

The closest work to what we do is the work done by Deng *et al.* [13]. Their methods use the set of experts' documents and predefined collections of documents called "communities" to identify the most relevant document for

a given query. To estimate the relevance score of documents to the queries, they proposed to smooth the probability of occurrence of query terms in the documents, using communities instead of the whole collection of documents. Afterwards, the relevance scores of expert's documents to the query are aggregated to estimate the relevance score of the expert to that query. There are some important differences between our method and the one proposed by Deng *et al.*:

- Our method uses latent research areas to update the weights of terms in both papers and reviewer profiles, while Deng *et al.*'s method only smoothes the experts' document models to identify the most relevant documents to a query;
- Our model uses latent research areas to directly identify the most relevant reviewers;
- Deng *et al.*'s formulation does not avoid zero probabilities when a query term does not exist in a (community of) document(s);
- Our method infers latent research areas of the reviewers' publications instead of relying on predefined communities, and, our model allows for a document to be associated with multiple research areas;
- Finally, our objective is to improve coverage and confidence of the team of reviewers assigned to papers, while Deng *et al.* aim to increase the *mean average precision (MAP) of the ranked list of candidates, given queries.*

Moreira and Wichert [35] combine multiple resources of expertise based on a multi-sensor framework together to find the relevance score of the candidates to the query. Three sensors are defined using the textual content, the

graph structure of the citation patterns, and profile information about academic experts in heterogeneous information network, respectively. However, the textual content is the only available resource in many cases such as ours. Moreira and Wichert also are ranking the individual experts given a query rather than building a team of experts for the query.

In Hashemi et al. [18], a discriminative learning algorithm is used for assigning non-equal expertise scores to authors of a paper in order to recognize the leading author in the paper, which is a different problem than our paper-assignment problem. Their method assigns a higher relevancy score to the author who is more knowledgeable in the topic of the paper and also considers multiple aspects of the paper in topic space (whereas we use term space).

Garcia and Sebastia [16] introduced a model to recommend a list of items for a group of people, based on the tastes and preferences of all the users in the group. This model is called a group recommender system. The problem of group recommendations can be seen as a "reverse" version of our problem: Its problem is to find a suitable item for a group of people, while we are building a group of reviewers for a specific paper.

Recently, a new language-modeling approach has been proposed by Liang and de Rijke [28, 29] to rank a group of experts (as opposed to an individual expert) for a query topic. Their method addresses a problem different from the one we address with our work. Our problem is the review team formation for papers, while their problem is ranking predefined expert groups for different queries. (Group building vs. Group ranking). Although the Liang and de Rijke's method is similar to our method in using a language modeling approach, it uses a simple smoothing method like other language modeling approaches. However, we propose to use latent research areas to update the language models of the reviewers and the papers.

2.3 The MARTA-LRA Method

2.3.1 Problem Definition

Let $\mathcal{P} = \{p_1, p_2, p_3, \ldots, p_N\}$ be a set of N papers to be reviewed, and let $\mathcal{R} = \{r_1, r_2, r_3, \ldots, r_M\}$ be a set of M reviewers, each represented by a set of texts (e.g., by their publications). The reviewers' profiles are constructed as the concatenation of their publications. Each paper is assumed to involve different knowledge aspects; each reviewer is also assumed to be an expert on a number of areas. The knowledge aspects of papers and the areas of expertise of reviewers are assumed to belong to a set of K topics, $\tau = \{\tau_1, \tau_2, \tau_3, \ldots, \tau_K\}$.

Under these assumptions, we consider the two variants of the multi-aspect paper-reviewers assignment problem:

Unconstrained Multi-Aspect Review-Team Assignment The objective is to assign each paper to a group g of m reviewers with diverse expertise, who, together, maximally covers the paper's aspects.

Constrained Multi-Aspect Review-Team Assignment The objective subsumes the above objective, while, at the same time, respecting the constraint that each reviewer can review at most rq (reviewer's quota) papers.

The set τ of all topics, as well as the actual aspects of papers and reviewers, are typically not available in practice and are difficult to obtain even under the best of circumstances.

To address the above problem in a realistic setting, we adopt the typical approach of extracting τ topics implicit in the textual contents of papers and reviewers' profiles, using a topic-modeling method, such as Probabilistic Latent Semantic Analysis (PLSA).

The constrained version of the problem is, in practice, more relevant for reviewer assignments in settings like scientific conferences with peer-reviewed

papers, where the number of papers to be reviewed in a short amount of time is relatively large and the number of reviewers in a program committee is limited. The unconstrained version is in practice more relevant in settings where the number of submissions is relatively small compared to a large pool of potential reviewers, as is the case of reviewing journal submissions.

2.3.2 Adjusting the Language Models of Reviewers and Papers using Latent Research Areas

In the multi-aspect review-team assignment problem, the group of reviewers assigned to a paper p_j has to be chosen to cover as many of p_j's topics as possible, while, at the same time, ensuring that each individual reviewer covers many of p_j's topics. To achieve such an assignment, let us first consider the *Kullback-Leibler* (KL) divergence measure, $D(\cdot||\cdot)$, that can be used to compute the relevance score of a reviewer r to a paper p, $R(r,p)$, as in equation (2.6) [27]:

$$R(r,p) = -D(\theta_r||\theta_p)\,, \tag{2.6}$$

where θ_r and θ_p are the language models of the reviewer r and the paper p respectively, which model the probabilities of the presence of different terms in the reviewer's profile and the paper. KL divergence is an asymmetric measure of the difference between two language models and can be estimated as in Equation (2.7):

$$D(\theta_r||\theta_p) = \sum_{t\in V} P(t|\theta_p)\log\frac{P(t|\theta_p)}{P(t|\theta_r)} = \sum_{t\in V} P(t|\theta_p)\log P(t|\theta_r) - P(t|\theta_p)\log P(t|\theta_p)\,, \tag{2.7}$$

where $\{t \in V\}$ is the set of terms in the vocabulary of the whole collection, V; $P(t|\theta_p)$ denotes the probability of a term t given the paper language model θ_p; and $P(t|\theta_r)$ is the probability of a term t given the reviewer language model θ_r of reviewer r. Since the second part of Equation (2.7) doesn't change the

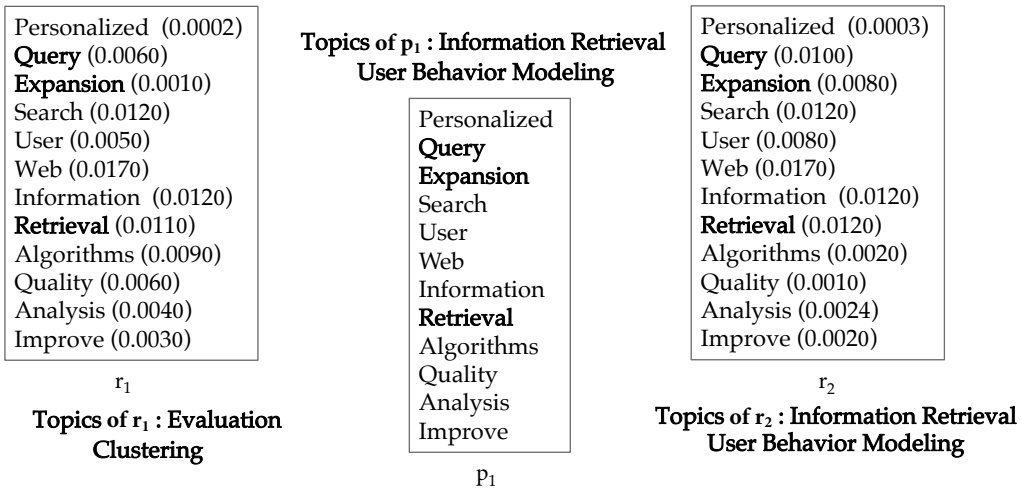


Figure 2.1: Example to illustrate reviewer assignment w.r.t. a paper (see main text for details).

ranking of reviewers for each paper, it can be ignored; our method considers the first term of this formula to estimate the relevance of reviewers to papers.

Note that, if a term does not appear in the profile of a reviewer, then $\log P(t|\theta_r)$ will be undefined ($P(t|\theta_r) = 0$), and the relevance score for the reviewer cannot be computed. This problem can be addressed with smoothing [56], which ensures that all terms have probabilities greater than zero. The smoothed reviewer model is defined in Equation (2.8):

$$P(t|\theta_r) = \lambda P(t|r) + (1 - \lambda)P(t|C)\,, \tag{2.8}$$

$P(t|r)$ can be, in a simple case, a maximum-likelihood estimate, i.e., the relative frequency of t in r's profile; and $P(t|C)$ is the probability of term t in the collection of reviewers, C. The coefficient $\lambda \in [0, 1]$ determines the relative strength of the contribution from the reviewer r's profile and from the whole collection of reviewers to the probability of the term t; In this manner, all term probabilities are smoothed equally for each reviewer, based on the collection of all reviewers profiles.

However, the same terms or keywords can be used in the characterization of different research (sub-)areas may not be equally important in these

different areas; thus, treating the terms differently, based on contextual information about the research areas in which the reviewers' publications belong, can potentially improve reviewer assignment.

To illustrate this intuition, consider, for example, the simple case illustrated in Figure 2.1. The terms and the topics of a paper p_1 are shown in the middle of the figure. The expertise topics of two reviewers, r_1 and r_2, and the probabilities of the paper's terms occurring in their respective profiles are shown to the left and right. With these values, which only consider the reviewers' profiles smoothed over the whole collection of reviewers as in Equation (2.8), r_1 is chosen to review p. Intuitively, however, r_2 appears to be more relevant to p. In relation to 'information retrieval'—one of the topics of paper p—terms like 'query', 'expansion' and 'retrieval' are more important than other less relevant but more widely and frequently occurring general terms. While the terms 'query', 'expansion' and 'retrieval' do have a higher probability for reviewer r_2, less relevant terms of the paper have a higher probability for reviewer r_1 and dominate the calculation of the reviewers' relevance. To increase r_2's relevance score, the weight of the important terms should be increased.

Using clusters of reviewers' publications as latent research areas can help achieve this objective, by assigning more weight to terms that are more central to those clusters. In a cluster about 'information retrieval', the terms 'query' and 'expansion' occur more often in the publications of the cluster, and this higher frequency can be used to adjust the probability of a term for a reviewer whose publications belong to that cluster.

To utilize latent research area information, two sub-problems need to be addressed:

- identifying the latent research areas of the reviewers' publications; and,
- improving the matching between reviewers and papers, based on these

latent research areas.

To obtain the latent research areas, we cluster the reviewers' publications. In this paper, we use a k-means with cosine distance for clustering all publications of all reviewers. We also use Latent Semantic Indexing to reduce the dimensionality of these documents in term space to improve the quality of clustering. The resulting clusters are called as latent research areas. Given these latent research areas derived from reviewers' publications, probabilities of term occurrences can be adjusted by adapting cluster-based information retrieval models [30], introducing a second factor to update the weights of the terms, as follows:

$$P(t|\theta_r) = \lambda P(t|r) + (1-\lambda)[\beta P(t|C_r) + (1-\beta)P(t|C)]\,, \qquad (2.9)$$

Each reviewer is associated with a number of latent research areas, which their publications belong to. C_r is the latent research area of the reviewer r that is most similar to the query paper. β is a coefficient indicating the relative importance of the latent research areas of the publications of reviewers in comparison to the whole collection. Equation (2.9) gives more weight to terms that appear more frequently in the latent research area of the reviewer which is most similar to the query paper.

Information of the latent research areas can also be used to update the language model of the query paper. First, the query paper is assigned to the most similar latent research area. Then, the contextual information of this latent research area is used to update the weights of the terms in the language model of the query paper, in order to give more weight to those terms of the paper that are more relevant in its latent research area. The term-weight adjustment is performed with a model-based feedback strategy [55] shown in Equation (2.10):

$$P(t|\theta_p) = \alpha P(t|p) + (1-\alpha)p(t|C_p)\,, \tag{2.10}$$

where $P(t|p)$ is the relative frequency of term t in the paper p, C_p is the latent research area assigned to the paper p, and α controls the effect of the latent research area model C_p.

To measure the quality of assigning a group of reviewers g to the paper p, $S(g,p)$, we adopt a *Maximal Marginal Relevance*(MMR) ranking strategy [8] in Equation (2.11), to simultaneously maximize coverage and confidence. The MMR strategy synthesizes the relevance of the reviewers to the paper with the diversity of the reviewers as a group.

$$S(g,p) = v\sum_{r\in g} R(r,p) - (1-v)[\frac{1}{|g|}\sum_{r\in g}\sum_{\substack{r'\in g\\ r\neq r'}} RD(r,r')]\,, \tag{2.11}$$

where $RD(r,r')$ is a measure of the similarity of the expertise of reviewers r and r' using the KL-divergence score. The formula for the review-team quality in Equation (2.11) reflects our intuition that the optimal group of reviewers for a paper should have two characteristics: (a) reviewers in the group should have expertise that is highly relevant for the paper, and (b) reviewers should have diverse expertise. The first term in Equation (2.11) reflects the first characteristic, by computing the sum of the relevance scores of all reviewers in the group to the paper. The second term reflects the second characteristic by computing the sum of pairwise similarities of the reviewers' expertise profiles; reviewers with very similar expertise profiles are unlikely to contribute much additional information value with their reviews. v controls the balance between these two features of a group of reviewers.

Algorithm 1: A Greedy Algorithm for CMARTA-LRA

Input : Set of N papers $\mathcal{P}$, set of M reviewers $\mathcal{R}$, set of reviewer capacities: $\mathcal{C} = \{C_{r_1}, \ldots, C_{r_M}\}$, papers' quota limit k.

Output: set of N groups of k reviewers assigned to the paper, G.

```
for p ∈ P do
    sum_of_relevance_scores_p ← 0;
    for r ∈ R do
        sum_of_relevance_scores_p ← R(r, p)
    end
end
Sort the set of papers P in ascending order of their sum of relevance scores;
for p ∈ P do
    g_p ⇐ ∅;
    repeat
        r ← argmax_{r∈R} S(g_p ∪ {r}, p);
        g_p ← g_p ∪ {r};
        C_r = C_r − 1;
        if C_i = 0 then
            Remove r from R;
        end
    until |g_p| = k;
end
```

2.3.3 Multi-Aspect Review-Team Assignment using Latent Research Areas

Unconstrained Multi-Aspect Paper-Review Assignment using Latent Research Areas (UMARTA-LRA) is a special case of *Constrained Multi-Aspect Paper-Review Assignment using Latent Research Areas* (CMARTA-LRA), by considering the reviewers' capacities in UMARTA-LRA equal to the number of all papers. We introduce a general framework to solve both problems. The goal is to find the best group of reviewers for each paper, i.e., the group that maximizes the quality score of Equation (2.11), given the reviewers' capacities.

We examine a heuristic forward-selection algorithm to optimize the defined objective function.

Our method first computes, for each paper p, the sum of relevance scores

of all reviewers to p, and then sorts the papers in ascending order of this sum of relevance scores, and considers papers with lower scores first. Thus, our method prioritizes papers with, in general, fewer relevant reviewers. This reduces the chance that the capacity of the few relevant reviewers for such a paper is exhausted before the paper is considered. Karimzadegan et al. [24] prioritize the papers with more topics—based on the intuition that there are more opportunities to assign the best reviewers to such papers. However, having more topics in a paper does not mean the number of reviewers who can cover these topics is low.

In our method, we iteratively optimize the team of reviewers assigned to each paper. For each paper, the next best reviewer —i.e., the one that maximizes the estimated quality score of the group assigned to the paper according to Equation (2.11)— is chosen in each step. Once a reviewer's quota (or paper's quota) is reached, that reviewer (respectively paper) is removed from the reviewer pool (respectively the paper list). The process continues until the reviewer teams for all the papers are selected. Algorithm 1 shows the pseudo-code.

2.4 Experimental Results

We conducted a set of experiments to address the following research questions:

Q1 *How sensitive are our methods to their parameters? What is the impact of λ, β, v and α on the coverage and confidence measures of our methods? How does the number of extracted aspects affect the performance of our method?*

Q2 *How useful is latent research area information for unconstrained and constrained multi-aspect reviewer assignment?*

Q3 *How do the group-based multi-aspect paper-review assignment methods, including ours, perform in comparison to the various baseline algorithms proposed by Karimzadegan* et al. *[25] and Neshati* et al. *[36] , which assign reviewers to a paper independently of the team of previously selected reviewers for that paper (group-based vs. independent reviewer–paper assignment), using topic space in both constrained and unconstrained settings?*

Q4 *How does our group-based objective function —which uses latent research areas— perform against the baseline group-based objective function proposed by Kou* et al. *[26] (term space vs. topic space)?*

We implemented our algorithm using the Lemur toolkit[1]. We extract topics from the union of all reviewer profiles using the PLSA topic-modeling method for the baseline methods. To obtain latent research areas, we use LSA (Latent Semantic Analysis) to reduce the dimensionality of reviewers' publications in term space, and, we applied *K-means* to cluster the reviewers' publications in this lower dimensional space. We use the cosine distance for clustering the reviewers' publications and for identifying the most similar latent research area for the query papers and the reviewers. The number of clusters was set to 25. We used the implementation of SDGA-SRA available through Kou *et al.*'s project homepage[2] [26]. We implement the methods of Karimzadegan *et al.* [25] and Neshati *et al.* [36] using a commercial CP solver, IBM ILOG CPLEX Optimizer 12.6(CPLEX)[3].

[1] `http://www.lemurproject.org/`

[2] `http://degroup.cis.umac.mo/reviewerassignment`

[3] `https://www-01.ibm.com/software/websphere/products/optimization/cplex-studio-community-edition/`

2.4.1 The Datasets

We used two datasets in our experiments. The *SIGIR dataset*[4] [25] includes 73 papers and 189 expert reviewers. In this dataset, the papers and the reviewers are manually annotated with ground-truth topics, selected among the ones mentioned in the call for papers (CfP) of the *ACM SIGIR* conference. Each paper is associated with at least two topics and each reviewer is associated with 25 topics. A second, larger dataset, the *PubMed dataset*[5], was constructed by Neshati *et al.* [36]. It includes 231 papers, retrieved by a crawler from the PubMed[6] database. PubMed papers are indexed by three medical subject headings (MeSH), a standard controlled vocabulary in life sciences. The 2009 version, which we used, contains 25,186 subject headings, each of which is associated with a number of more specific subheadings. Each paper in the PubMed dataset is associated with at least two breast-cancer subheadings; on average, the papers in this collection are associated with 3.6 subject headings. For each of the subheadings associated with a paper in this collection, the crawler identified authors with at least 10 papers on this subheading; the collection of these authors —98 in total— constitute the pool of potential reviewers. The profile of each reviewer was created using titles and abstracts of their publications.

The two datasets exhibit some interesting differences. On average, papers in PubMed are associated with more ground-truth subject headings than the papers in SIGIR. This implies that, in order to cover them, the expertise of the review teams assigned to PubMed papers has to be broader than that of the SIGIR review teams. However, PubMed reviewers have, typically, narrower expertise than SIGIR reviewers because PubMed reviewers are associated with

[4] `http://timan.cs.uiuc.edu/data/review.html`
[5] `http://isl.ce.sharif.edu/pubmed/dataset/`
[6] `http://www.ncbi.nlm.nih.gov/pubmed`

fewer ground-truth subject headings than SIGIR reviewers, who are associated with 25 ground-truth topics. Furthermore, the reviewers in the PubMed dataset are more similar to each other than SIGIR reviewers, since they are all associated with breast-cancer subject headings. Using KL-divergence, the average dis-similarity score between reviewers is 1.60 and 2.35 for PubMed and SIGIR, respectively. Therefore, since the reviewers of the PubMed dataset are more similar to each other than the SIGIR reviewers, we expect that using latent research areas (the clusters of the reviewers' publications) should improve the quality of the review teams assigned to SIGIR papers more than those of the PubMed papers.

2.4.2 Evaluation Measures

We evaluate the effectiveness of our methods using the *coverage* and *average confidence* measures, defined in [25], using the *ground truth* topics assigned to papers and reviewers. The coverage of a paper by a set of reviewers is defined as the percentage of its topics that are covered by the reviewers topics of expertise, as shown in Equation (2.12):

$$\text{Coverage} = \frac{n_r}{n_A}\,, \tag{2.12}$$

where n_A is the number of *all* topics of the paper and n_r is the number of paper topics that are also topics of expertise of the reviewers assigned to the paper.

A second indicator of a good assignment is the degree to which individual reviewers cover more than one of the paper topics. The average confidence measure, shown in Equation (2.13), captures this indicator by measuring how redundant the reviewers are in covering the various topics of the paper:

$$\text{Average confidence} = \frac{1}{n_A}\left(\sum_{i=1}^{n_A} \frac{n_{A_i}}{n}\right). \tag{2.13}$$

where n_{A_i} is the number of reviewers assigned to the paper that can cover topic A_i and n is the total number of reviewers assigned to the paper.

2.4.3 Baseline Methods

To find out how effective using *latent research area information* is, we define two baseline methods for the unconstrained and constrained versions of the problem by using the *standard* KL-divergence in Equation (2.11) instead of our proposed extended version (as shown in Equations (2.9) and (2.10)). These baseline methods are called UMARTA-KL and CMARTA-KL, respectively. $\lambda = 0.9$ is used in both methods for both the PubMed and SIGIR datasets. The comparisons between our methods and these simplified baseline methods should show if latent research area information, as we propose, does indeed contribute positively to the performance.

To find out how effective the *group information* is, we compare UMARTA-LRA against Karimzadegan's review aspect modeling (RAM) [25] and Neshati's uncapacitated facility-location analysis (UFLA) [36]; and we compare CMARTA-LRA, against Nesthati's capacitated facility location analysis (CFLA) [36] and Karimzadegan's integer linear programming (ILP) [24].

The parameters of the comparison methods are set to the values that were recommended in the respective publication. As recommended in [36], λ is set to 0.1 and 0.2 for the UFLA method on the SIGIR and PubMed datasets, respectively. As recommended in Karimzadegan *et al.* σ is set to 0.1 and 0.2 for the RAM method on the SIGIR and PubMed datasets, respectively. The number of extracted topics from the papers and reviewers' profiles is set to 25. For CFLA, the value of λ is set to 0.5. The program committee size for SIGIR is 189, and the reviewing capacity of each reviewer and the number of reviewers assigned to each paper is set to 5 and 3, respectively. The program committee size for PubMed is 98, and the reviewing capacity of each reviewer and the

number of reviewers assigned to each paper is set to 10 and 3, respectively. Karimzadegan *et al.* and Neshati *et al.* also set low probability values of extracted topics to zero to improve the performance of their methods.

We also compare our group-based objective function in *term space* with the weighted-coverage group based reviewer assignment objective function in *topic space*, WCGRA, proposed by Kou *et al.* [26], in order to show which space is more informative and appropriate for the reviewer-paper assignment. Moreover, Kou *et al.* use the Author-Topic Model (ATM) [42] to extract the expertise aspects of reviewers and papers. To make the comparison fair, we use the same topic vectors and constraints in CFLA, ILP and WCGRA. As topic vector generation may affect the performance of the methods in topic space, We also use both ATM and PLSA to generate the topic vectors. In reporting the results, we designate the different cases by putting the name of topic modeling method used alongside the name of the baseline methods, for example, a CFLA-PLSA designation specifies the case when the PLSA topic modeling method is used in CFLA.

2.4.4 Parameter Setting for UMARTA-LRA and CMARTA-LRA

For CMARTA-LRA, the reviewer quota and paper quota are set to the same values as for CFLA and ILP, i.e., 5 and 3 for SIGIR, and 10 and 3 for PubMed. The remaining parameters are studied in more detail in this section.

We study the effect of each parameter on both datasets, but we use the SIGIR dataset to set the default values for each parameter, used then on both datasets. Table 2.1 summarizes different parameters of our method, and their selected values have been put in bold face. In all following experiments, we use these values for the parameters of our methods if we don't mention anything further.

Table 2.1: The summarization of different parameters of our method

Parameters	Definition	Equation	Value range
λ	Reviewer's Profile Smoothing Parameter	(2.8)	0, 0.1,. . . ,. . . ,**0.9**, 1
β	Reviewer's Latent Research Area Balance Factor	(2.9)	0, 0.1,. . . ,**0.5**,. . . , 1
α	Query's Latent Research Area Balance Factor	(2.10)	0, 0.1,. . . ,**0.6**,. . . , 1
v	Diversity factor	(2.11)	0, 0.1,. . . ,**0.6**,. . . , 1
p	Number of LSA components		15, 20, **25**, 30
k	Number of clusters		10, 15, **25**, 30, 35, 40

The authors of the comparison methods [24, 36] did perform a fine–tuning on the same data sets and reported their best values in their papers. Therefore, since we use the same datasets, we think it is fair to use their reported best parameter values.

Figure 2.2 shows the coverage and average confidence of UMARTA-LRA using different numbers of aspects extracted by LSA; note that, subsequently, the reviewers' publications and the query papers are represented as vectors of these latent aspects and clustered to discover the latent research areas required by our method. The method seems to be very robust with respect to the number of latent aspects, with little variation in coverage and average confidence over a wide range of numbers of aspects. Using 25 aspects yields slightly better results than the other tested values in the SIGIR data set, which is why we fix the value of extracted aspects to 25 for all methods in both datasets.

Figures 2.3, 2.4, 2.5 and 2.6 illustrate the sensitivity of UMARTA-LRA to the combination of coefficients λ, β, α and v, on both datasets. Recall that λ in (2.9) and α in (2.10) determine how important the latent research areas are in the reviewers' profiles and the query papers, respectively, in adjusting the term frequencies; β in (2.9) is used to include a smoothing term, considering

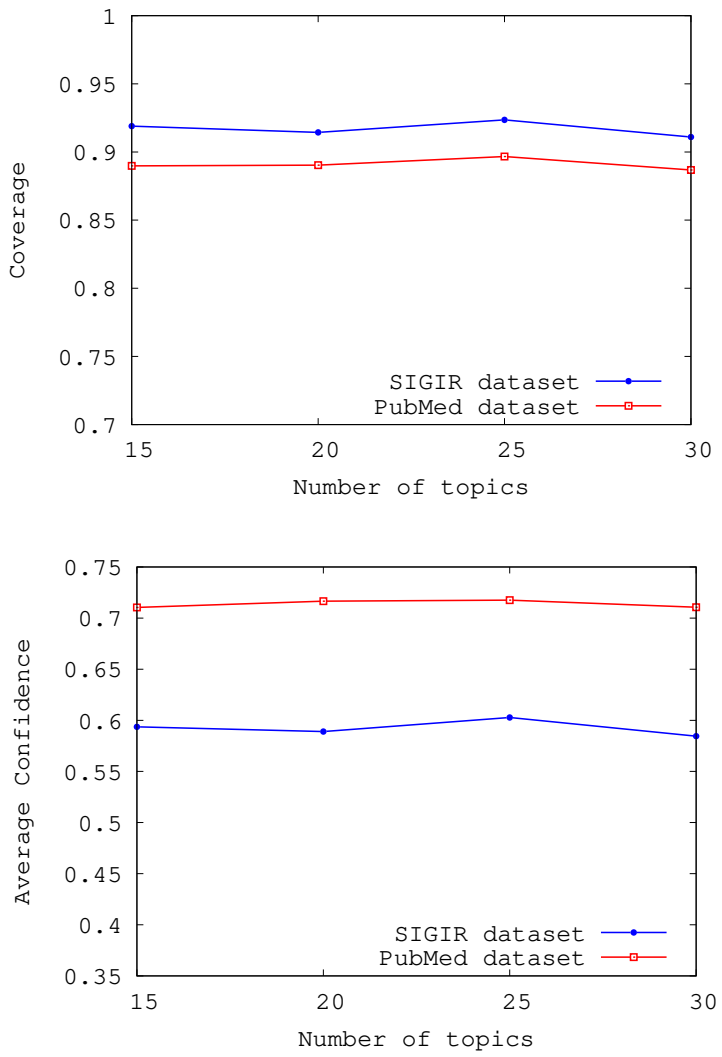


Figure 2.2: Coverage and average confidence of UMARTA-LRA using various numbers of aspects in LSA for $\lambda = 0.9$, $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

the whole collection, in case a term does not occur in the reviewer's profile nor in the latent research areas of her papers; and v in (2.11) balances the review-team expertise for the query paper against the diversity of the profiles of the reviewers in the team.

In both datasets, increasing the value of λ (without changing the chosen values for other parameters) improves coverage and average confidence, indicating that the reviewer profiles are the main source of information for identifying the reviewers' expertise aspects. Based on these results, we set

$\lambda = 0.9$ in all other experiments with UMARTA-LRA.

Increasing the value of β improves the coverage in the SIGIR dataset, indicating that latent research area information can improve the reviewer-assignment process, but the effect is less pronounced on the PubMed dataset, in which the latent research area are more similar to each other than in the SIGIR dataset. Since $\beta = 0.5$ shows better coverage and average confidence in the SIGIR dataset, We use a value of $\beta = 0.5$ for both datasets in all other experiments. This gives equal importance to the whole collection and the latent research areas of reviewers.

As Figure 2.5 shows, increasing the value of α improves the coverage for SIGIR dataset and average confidence for both datasets to some degree and up to about a value of $\alpha = 0.5$, indicating that using the closest latent research area to a query paper to adjust the language model of the paper improves the result. We set α to 0.6 for both datasets in all other experiments because $\alpha = 0.6$ yields maximum coverage and average confidence in the SIGIR dataset.

Figure 2.6 shows increasing the value of v improves, as can be expected, the average confidence in both datasets since for low values of v the assignment of a group of reviewer is mainly based on the diversity of the group without considering the relevance of the reviewers with respect to the paper. Although reviewers in a group should be diverse, they should also cover as many topics as possible, i.e., have sufficient knowledge about these covered topics, as measured by the relevance score. When multiple reviewers are available that can cover all the aspects of a submission, it is preferable that the team consists of reviewers who have diverse expertise beyond the paper. Since $v = 0.6$ results in maximum coverage and average confidence in the SIGIR dataset, this value of v is chosen as a default value for both datasets in all other experiments.

To validate the cluster structure, the silhouette coefficient [43] is used.

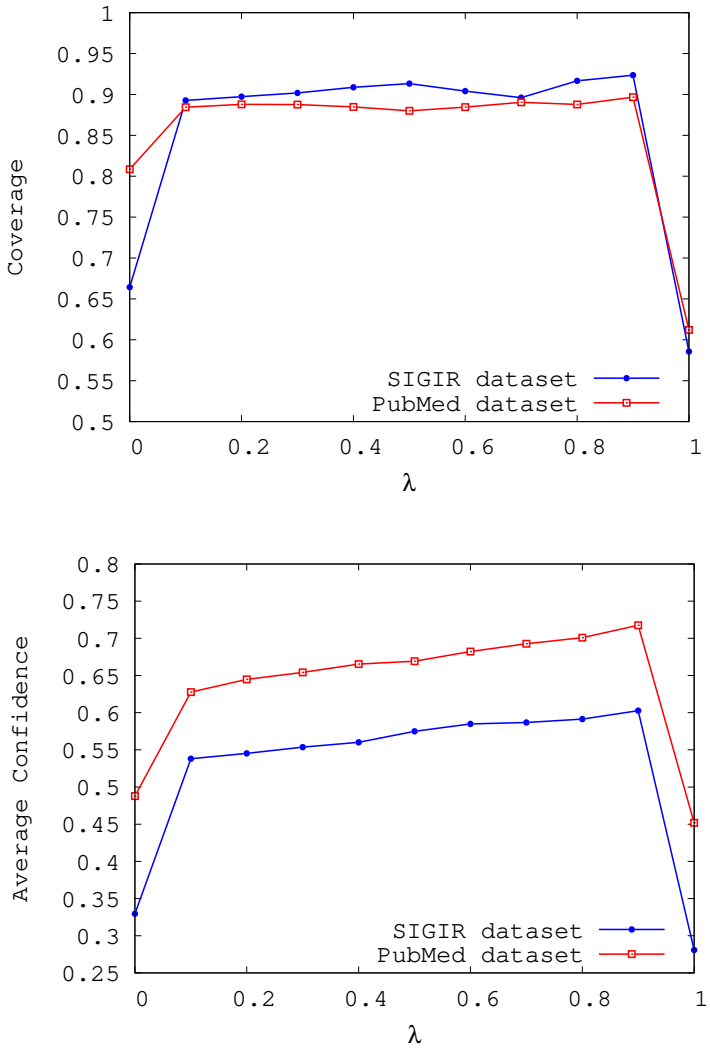


Figure 2.3: Sensitivity of UMARTA-LRA w.r.t λ for $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

This measure shows the similarity of an object to its own cluster (cohesion) in comparison to other clusters (separation). In summary, the higher the silhouette coefficient, the better the quality of the clustering.

Figure 2.7 shows that the highest silhouette value for both datasets occurs when the number of clusters is 25. Hence, the number of clusters is set to 25 for both dataset. It is also worth noting that this value of k is the same as the number of ground truth topics. It shows that our intuition to consider these clusters as the latent research areas seems reasonable.

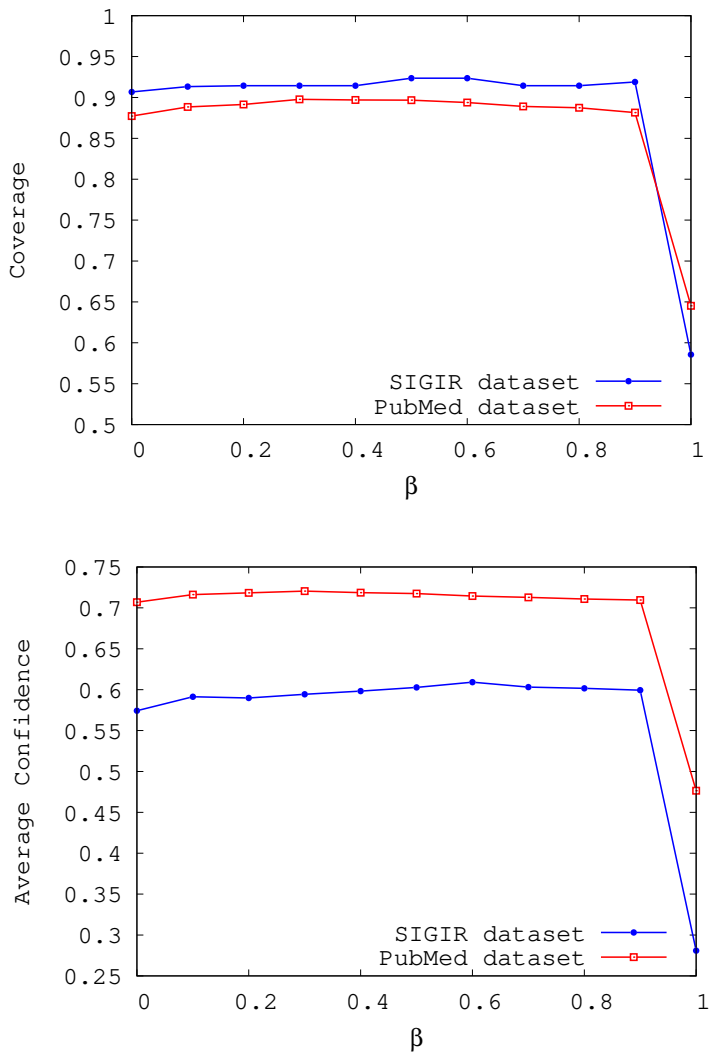


Figure 2.4: Sensitivity of UMARTA-LRA w.r.t β for $\lambda = 0.9$, $\alpha = 0.6$ and $v = 0.6$

We analyzed the effect of different number of clusters on the result of the methods evaluation measures. As shown in Figure 2.8, although the performance of the CMARTA-LRA is very robust when the default values of other parameters are used, for average confidence and coverage we can see a slightly different performances (at a still high level), which have their maximum values at 25 for both datasets. As shown in Figure 2.9 ,the performance of UMART-LRA is also robust when using various number of clusters. However, the value of 25 for the number of clusters yields to the best coverage and average confi-

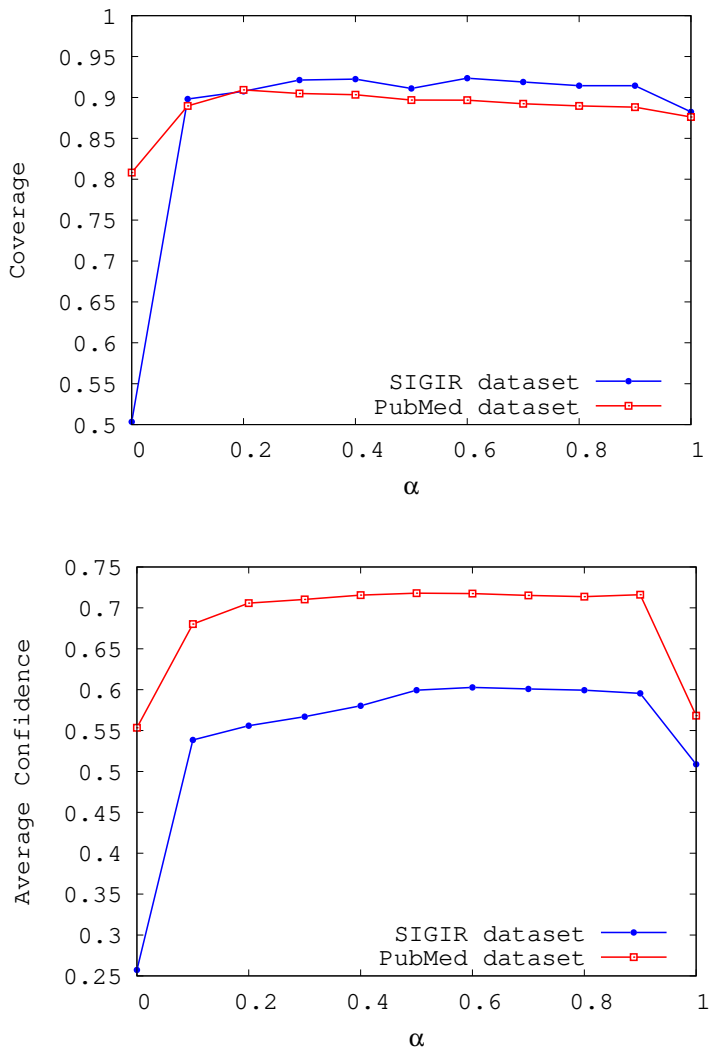


Figure 2.5: Sensitivity of UMARTA-LRA w.r.t α for $\lambda = 0.9$, $\beta = 0.5$ and $v = 0.6$

dence for both datasets.

Similar to UMARTA-LRA, Figure 2.10 shows the average confidence of CMARTA-LRA does not vary substantially with the number of latent aspects used. The maximum value of coverage for CMARTA-LRA is obtained if the number of latent aspects are 20 and 25 for the SIGIR and PubMed datasets, respectively.

Figures 2.11, 2.12, 2.13 and 2.14 illustrate the effects of different parameter values on the performance of the CMARTA-LRA, which are similar to

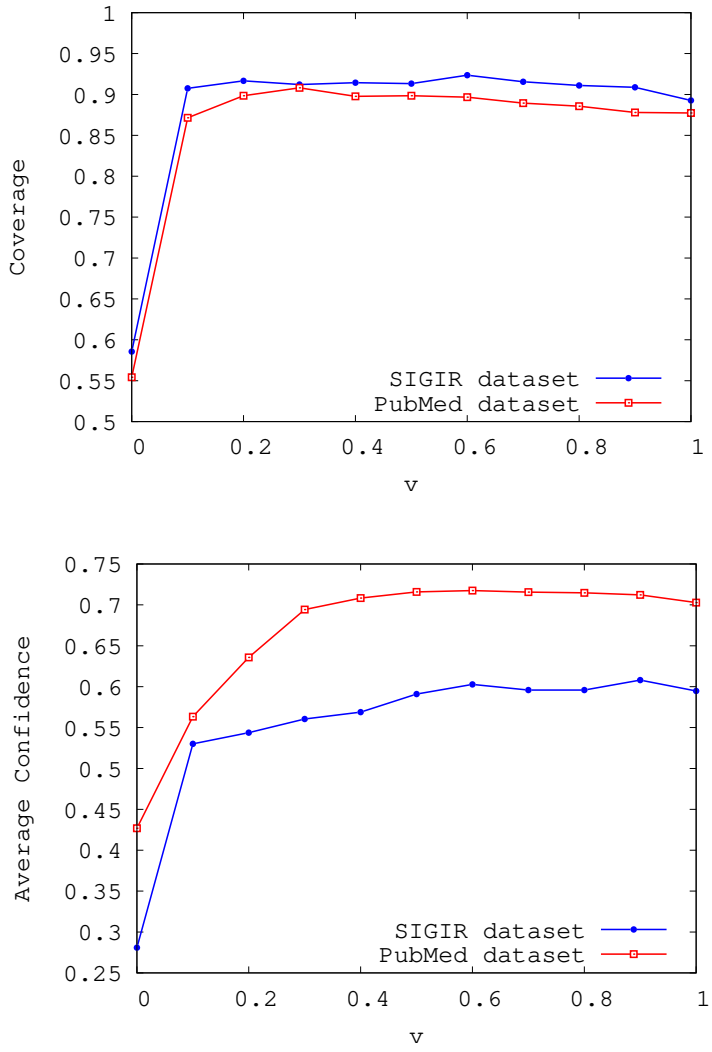


Figure 2.6: Sensitivity of UMARTA-LRA w.r.t v for $\lambda = 0.9$, $\beta = 0.5$ and $\alpha = 0.6$

their effects on UMARTA-LRA, showing an overall very robust behavior of a parameter when the others are fixed at their default values.

2.4.5 Evaluation of UMARTA-LRA

Tables 2.2 and 2.3 report the coverage and average confidence measures of different methods using their recommended parameters. The parameters for our method are shown in Table 2.1.

To show how effective the group information is, we compare the result of

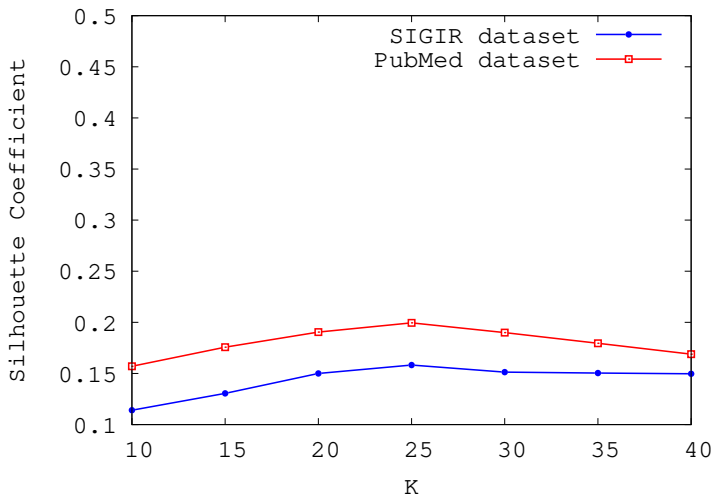


Figure 2.7: The value of silhouette coefficient using various numbers of clusters

Table 2.2: Comparison of the UMARTA-LRA with other baseline algorithms

Dataset	Method	Coverage	Average confidence
SIGIR	RAM	0.869	0.501
	UFLA	0.900	0.564
	UMARTA-LRA	**0.923**	**0.602**
PubMed	RAM	0.798	0.424
	UFLA	0.826	0.683
	UMARTA-LRA	**0.896**	**0.717**

UMARTA-LRA against results of the RAM and UFLA methods. As Table 2.2 shows, our method outperforms these baseline algorithms in both coverage and average confidence.

In Figures 2.15 and 2.16, we can see more clearly how the performance of UFLA, RAM and UMARTA-LRA are sensitive to the number of aspects in PLSA and LSA. Our method, UMARTA-LRA is more robust to the number of aspects than both UFLA and RAM.

It is important, however, to note a key difference in the role that this number plays in the methods we compare: UMARTA-LRA uses the number

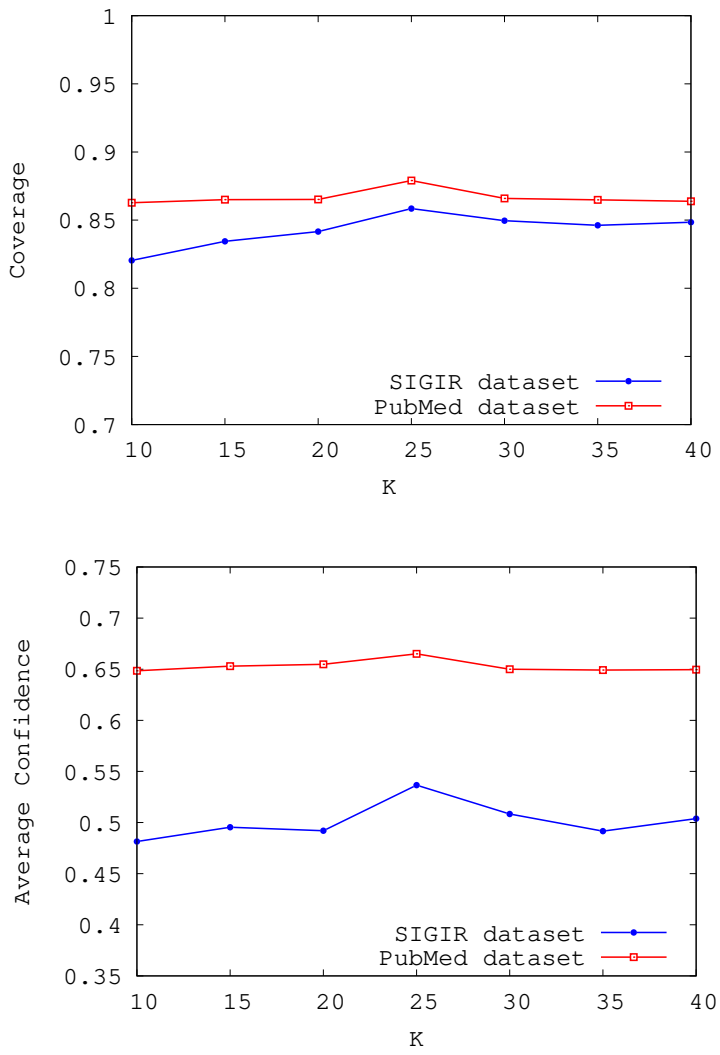


Figure 2.8: Coverage and average confidence of CMARTA-LRA using various numbers of clusters (k) for $\lambda = 0.9$, $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

of aspects only for dimensionality reduction in clustering the documents; in contrast, the competitor baseline methods use the number of aspects as the number of topics in terms of which to represent reviewers and papers.

Considering the quality of a team of reviewers as a whole, in comparison to independent review-paper assignment, increases the coverage and average confidence of the assignments.(Q3)

Table 2.3 shows that using the latent research area information in UMARTA-

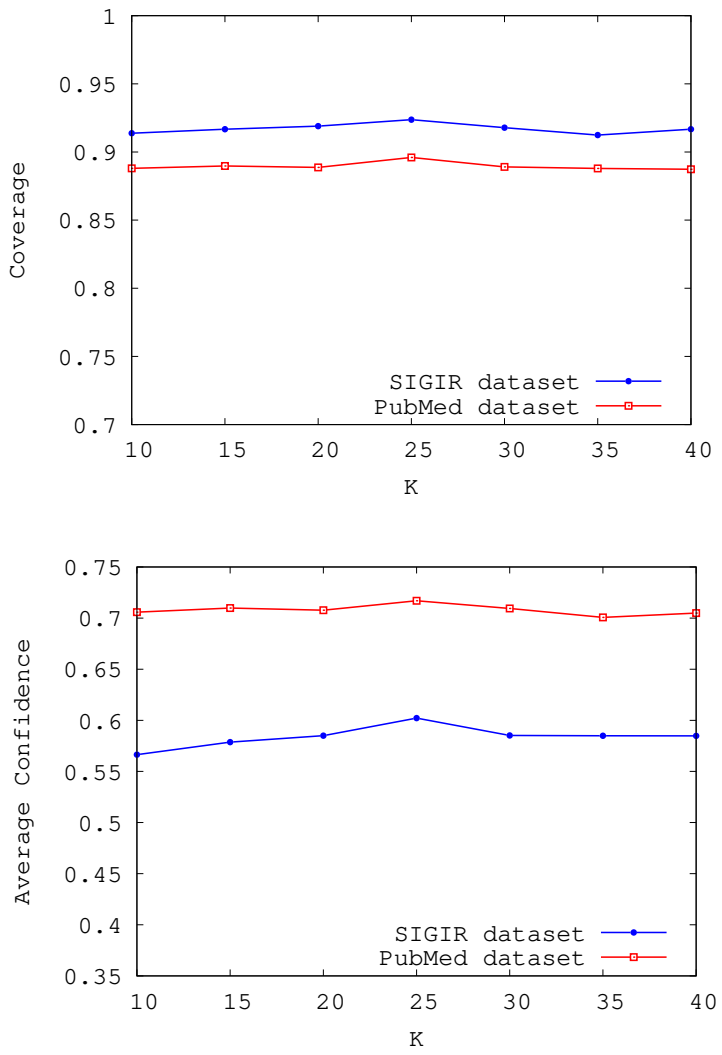


Figure 2.9: Coverage and average confidence of UMARTA-LRA using various numbers of clusters (k) for $\lambda = 0.9$, $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

Table 2.3: Comparison of the UMARTA-LRA with UMARTA-KL

Dataset	Method	Coverage	Average confidence
SIGIR	UMARTA-KL	0.698	0.395
	UMARTA-LRA	**0.923**	**0.602**
PubMed	UMARTA-KL	0.824	0.646
	UMARTA-LRA	**0.896**	**0.717**

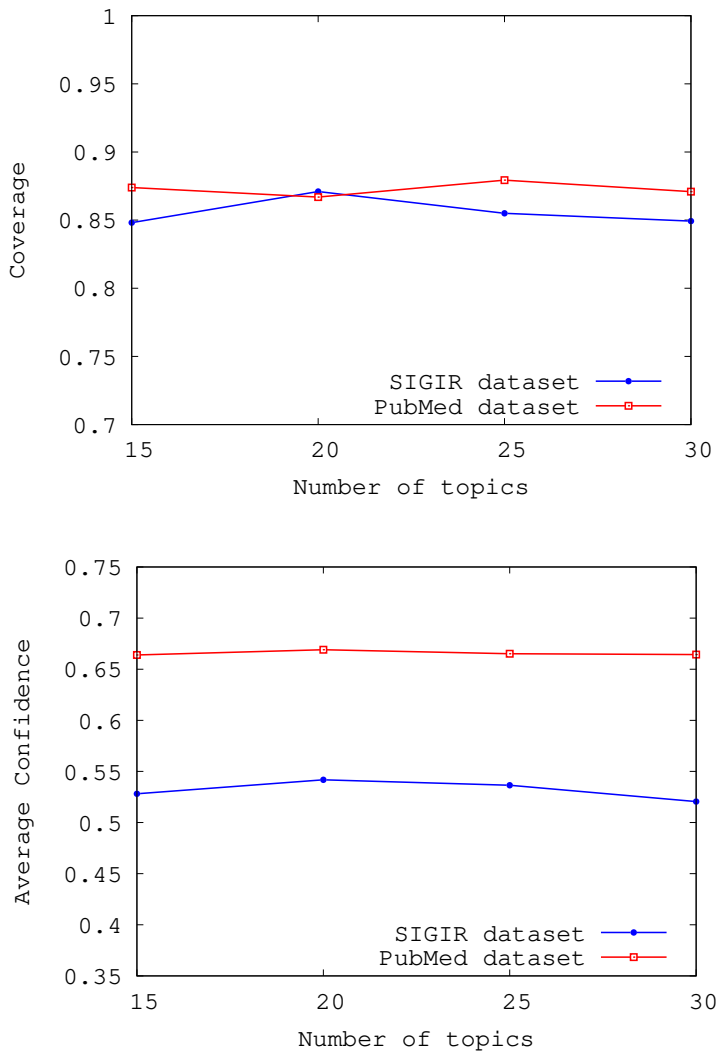


Figure 2.10: Coverage and average confidence of CMARTA-LRA using various numbers of aspects in LSA for $\lambda = 0.9$, $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

LRA improves performance in all measures, compared to not using this information as in the baseline method UMARTA-KL.

Our method is very effective in maximizing the coverage of the paper's topics by the assigned reviewers, and implicitly validates the usefulness of latent research areas in improving multi-aspect reviewer assignment.(Q2)

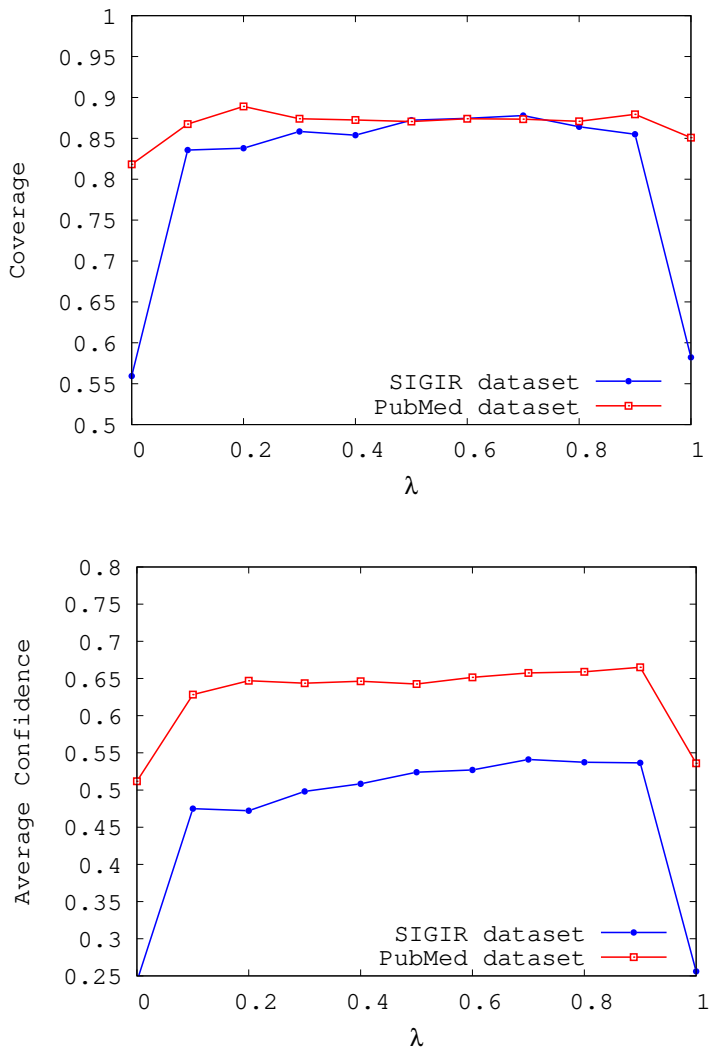


Figure 2.11: Sensitivity of CMARTA-LRA w.r.t λ for $\beta = 0.5$, $\alpha = 0.6$ and $v = 0.6$

2.4.6 Evaluation of CMARTA-LRA

Table 2.4 and 2.5 summarizes the results of comparing the group-based methods WCGRA and our method CMARTA-LRA against the baseline methods that assign reviewers to each paper independently (CFLA and ILP). We used PLSA to extract topics for CFLA, ILP and WCGRA, the results of which are reported in Table 2.4. Table 2.5 reports the results for when we use ATM instead of PLSA to generate topic vectors for CFLA,ILP and WCGRA. An interesting observation is that Kou *et al.* [26] used ATM to extract the topic

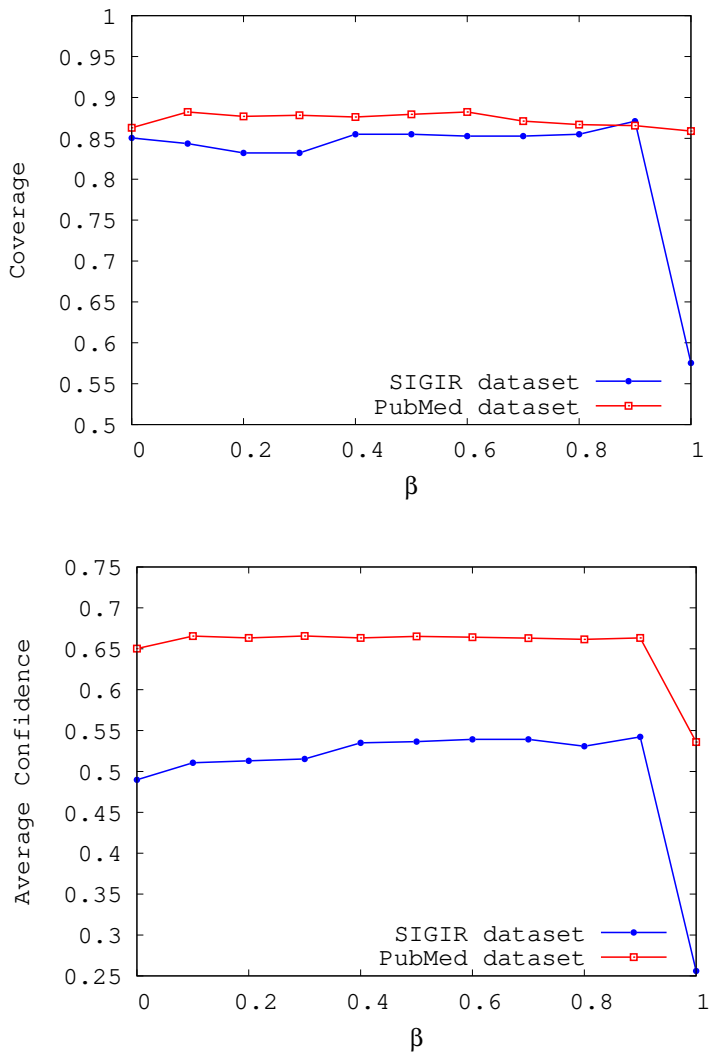


Figure 2.12: Sensitivity of CMARTA-LRA w.r.t β for $\lambda = 0.9$, $\alpha = 0.6$ and $v = 0.6$

vectors; However, using PLSA in WCGRA, CFLA and ILP improves the performance of all these methods in both coverage and average confidence for both SIGIR and PubMed datasets.

Similar to previous experiments with UMARTA-LRA, the default values of the parameters for CMARTA-LRA are used. As Tables 2.4 and 2.5 show, our proposed method clearly outperforms other methods in all measures.

Although CFLA and ILP use the Integer Linear Programming to optimize their solutions, our simple, greedy approach performs better with the infor-

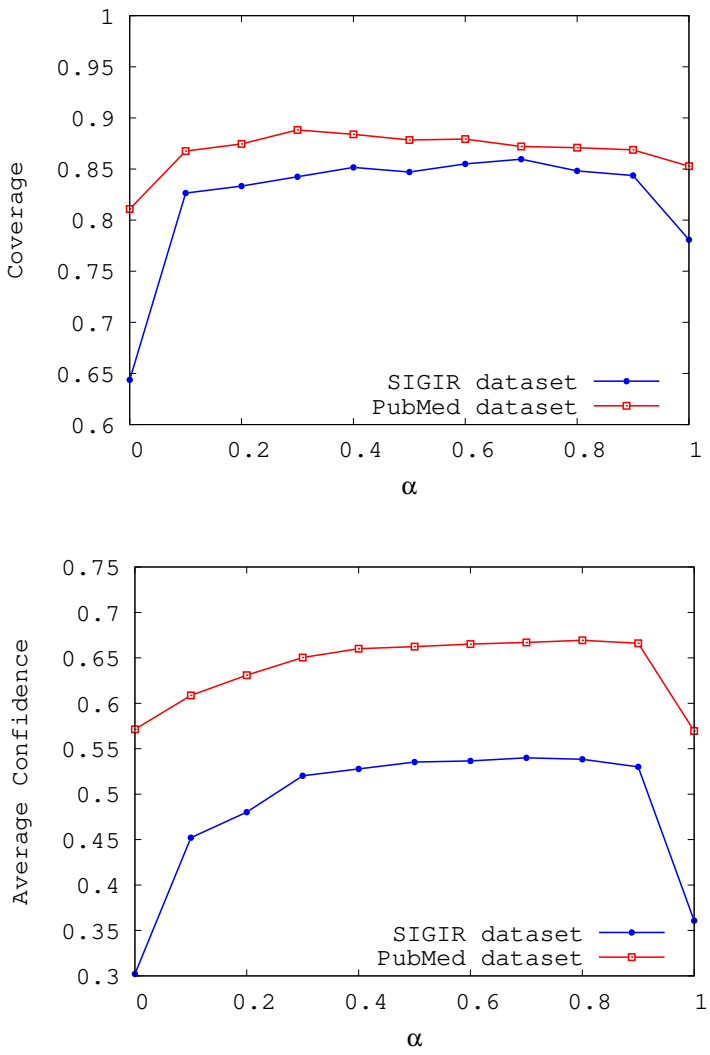


Figure 2.13: Sensitivity of CMARTA-LRA w.r.t α for $\lambda = 0.9$, $\beta = 0.5$ and $v = 0.6$

mation of latent research areas which better captures the relevance of reviewers to papers. Furthermore, our method solves the constrained multi-aspect review paper assignment problem using an objective function that takes reviewer diversity into account instead of choosing reviewers independently only to maximize coverage of paper topics.

Both CFLA and ILP define new similarity measures between reviewers and papers in topic space. They use the Integer Linear Programming framework to optimize the quality of the assigned group of reviewers for each paper using

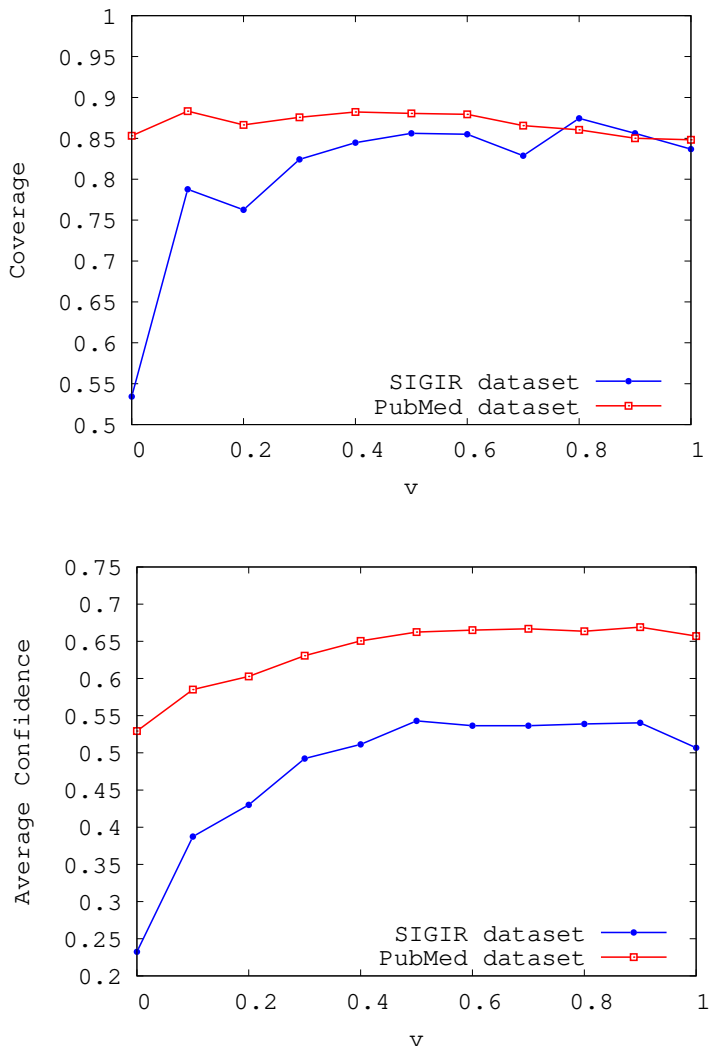


Figure 2.14: Sensitivity of CMARTA-LRA w.r.t v for $\lambda = 0.9$, $\beta = 0.5$ and $\alpha = 0.6$

their similarity measures. For the sake of completeness, we also use the Integer Linear Programming framework to optimize the group of assigned reviewers for each paper using our similarity measure between groups of reviewers and papers in term space (see equation (2.11)). We call the latter, CMARTA-LRA-ILP. As shown in the table 2.4, when we adapt our similarity measure to use the Integer Linear Programming, our method outperforms both CFLA and ILP in all measures.

Moreover, WCGRA with PLSA, as a group-based method in topic space,

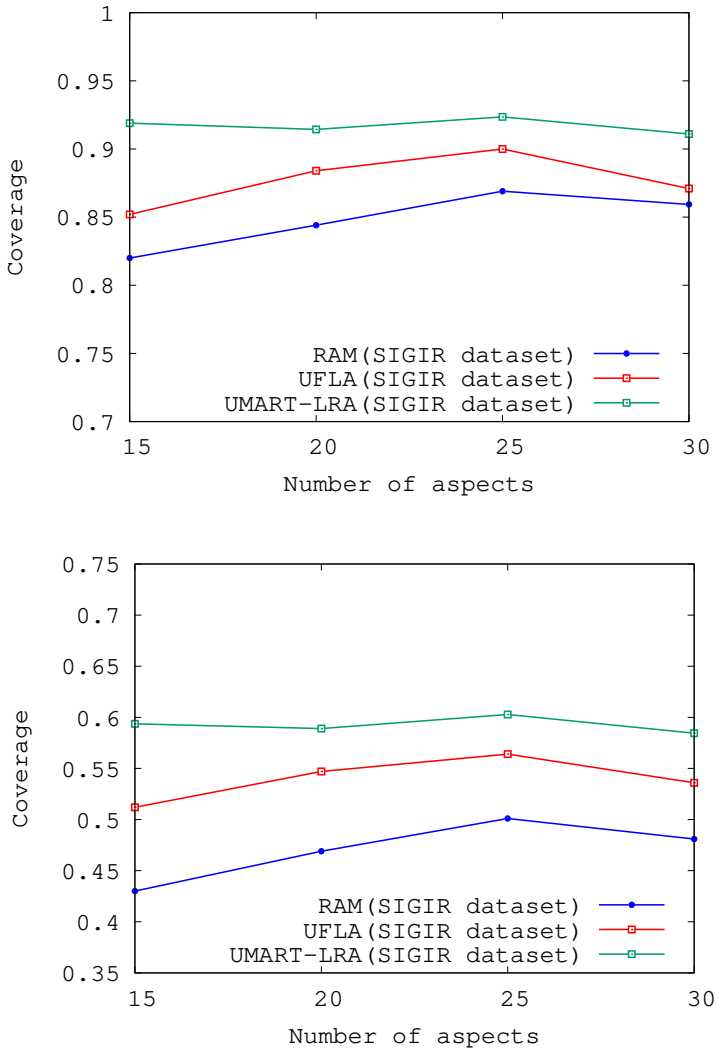


Figure 2.15: Coverage and average confidence of UMARTA-LRA, RAM and UFLA using various numbers of aspects in LSA and PLSA respectively, in SIGIR dataset

performs better, in terms of both coverage and average confidence, than CFLA and ILP using PLSA, which are doing independent reviewer-paper assignments in topic space. Therefore, optimizing the team of reviewers for each paper instead of considering each reviewer independently can improve the quality of paper-reviewer assignments significantly.

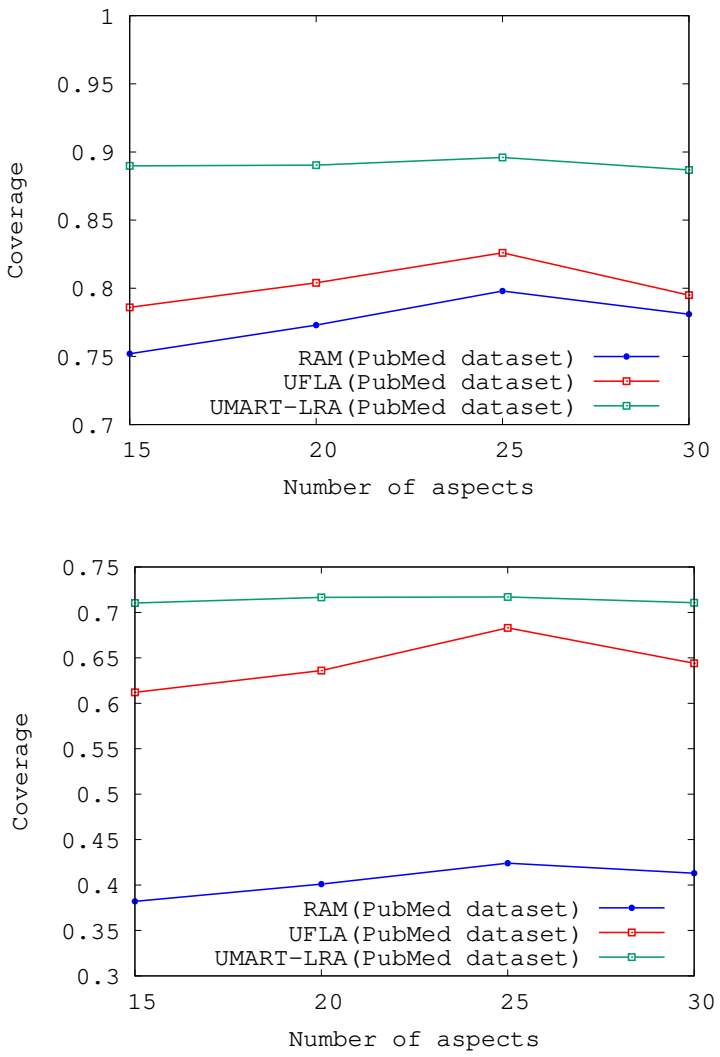


Figure 2.16: Coverage and average confidence of UMARTA-LRA, RAM and UFLA using various numbers of aspects in LSA and PLSA respectively, in PubMed dataset.

Using a group-based objective function improves multi-aspect reviewer assignment in the constrained setup as well (Q3),

As Table 2.4 and 2.5 show, our objective function outperforms the WCGRA objective function, when measured against the ground truth, by a large margin, in both coverage and average confidence of the resulting assignment, regardless of different topic vector generation methods used. WCGRA focuses solely

Table 2.4: Comparison of group-based reviewer-paper assignment methods with other baseline algorithms using the PLSA topic modeling method

Dataset	Method	Coverage	Average confidence
SIGIR	ILP-PLSA	0.501	0.211
	CFLA-PLSA	0.526	0.227
	WCGRA-PLSA	0.574	0.24
	CMARTA-LRA-ILP	0.762	0.417
	CMARTA-LRA	**0.855**	**0.536**
PubMed	ILP-PLSA	0.774	0.568
	CFLA-PLSA	0.739	0.528
	WCGRA-PLSA	0.791	0.587
	CMARTA-LRA-ILP	0.85	**0.675**
	CMARTA-LRA	**0.879**	0.665

Table 2.5: Comparison of group-based reviewer-paper assignment methods with other baseline algorithms using ATM topic modeling method

Dataset	Method	Coverage	Average confidence
SIGIR	ILP-ATM	0.374	0.168
	CFLA-ATM	0.502	0.203
	WCGRA-ATM	0.461	0.181
	CMARTA-LRA	**0.855**	**0.536**
PubMed	ILP-ATM	0.69	0.483
	CFLA-ATM	0.751	0.495
	WCGRA-ATM	0.792	0.524
	CMARTA-LRA	**0.879**	**0.665**

on maximizing the coverage score of the review team assigned to a paper and exhibits much lower average confidence than our group-based objective function that tries to maximize both relevance and diversity of the review team. Moreover, CMARTA-LRA represent the reviewers and papers expertise in term space using the latent research information rather than in topic space.

To compare the performance of our group-based objective function using latent research areas with the performance of the group-based objective function WCGRA-PLSA [26], we use three approximation algorithms: our greedy

Table 2.6: Comparison of the CMARTA-LRA with WCGRA using three different approximation algorithms

Dataset	Method	Coverage		Average confidence	
		WCGRA-PLSA	CMARTA-LRA	WCGRA-PLSA	CMARTA-LRA
SIGIR	SDGA	0.542	**0.845**	0.232	**0.519**
	SDGA-SRA	0.574	**0.857**	0.24	**0.511**
	Greedy	0.553	**0.855**	0.235	**0.536**
PubMed	SDGA	0.773	**0.875**	0.514	**0.674**
	SDGA-SRA	0.791	**0.877**	0.527	**0.670**
	Greedy	0.736	**0.879**	0.501	**0.665**

forward selection and two algorithms proposed by Kou *et al.* [26], SDGA, and, SDGA-SRA. The results for coverage and average confidence are shown in Table 2.6. As Table 2.6 shows, our objective function outperforms the WCGRA objective function by a large margin in both coverage and average confidence of the resulting assignment, regardless of the different approximation algorithms used. The performance of our similarity measure using different approximation algorithms in reviewer paper assignment shows that our method is robust and efficient in comparison to their competitors. Moreover, our proposed greedy approach combined with our introduced similarity measure performs best among all cases considered.

Matching reviewers' expertise and papers has performed better in term space using the latent research information (CMARTA-LRA) rather than in topic space (WCGRA) (Q4).

To show how effective using the latent research area information is in constrained multi-aspect expertise matching, we compare CMARTA-LRA with CMARTA-KL. Table 2.7 shows the results of CMARTA-LRA and CMARTA-KL. In both datasets, using the latent research area information helps to estimate the relevance score of reviewers and papers more effectively. As we

Table 2.7: Comparison of CMARTA-LRA and CMARTA-KL

Dataset	Method	Coverage	Average confidence
SIGIR	CMARTA-KL	0.60	0.26
	CMARTA-LRA	**0.855**	**0.536**
PubMed	CMARTA-KL	0.84	0.52
	CMARTA-LRA	**0.879**	**0.665**

explain in section 2.4.1, the effect of the latent research areas is more pronounced in SIGIR rather than PubMed datasets.

Our experimental results demonstrate that using the latent research areas of the reviewers' publications, when assigning teams of reviewers to papers, results in improved paper-reviewer assignments over all baseline and state-of-the-art methods, both in terms of coverage and average confidence, and in both the unconstrained and constraint problem variants (Q2).

To illustrate the effectiveness of CMARTA-LRA even further, we show some example cases in which we compare the reviewers assigned using our method with the assignments of the baseline methods.

In Table 2.8, the information about four query papers and assigned reviewers, along with their ground truth topics—which are used to evaluate the quality of the assignments—are reported. For each reviewer, the covered topics of the paper are shown in bold face. For query papers 2 and 57 in PubMed and papers 5 and 54 in SIGIR, the team of reviewers assigned using CMARTA-LRA covers a higher number of topics. Moreover, most of the assignments add a new knowledge aspect to the group of reviewers. This is due to the consideration of other reviewers' coverage when the next one is being chosen.

We also prioritize papers with lower values of the sum of relevance scores (i.e., fewer relevant reviewers, in general), in order to avoid improper assign-

Table 2.8: The reviewers assignment of CMARTA-LRA, ILP and CFLA for some queries in PubMed and SIGIR datasets

PubMed Dataset			
Paper	Method	Team of reviewers	Covered Topics
$P_2 : (T_1, T_3, T_6, T_8, T_{12}, T_{13})$	ILP-PLSA	$R_{30} : (\mathbf{T_1}, T_7)$ $R_{41} : (\mathbf{T_1}, T_4, T_7)$ $R_{42} : (\mathbf{T_1}, T_4, \mathbf{T_6}, T_7, T_{10}, T_{15})$	$(\mathbf{T_1}, \mathbf{T_6})$
	CFLA-PLSA	$R_{17} : (\mathbf{T_1}, T_2, \mathbf{T_3}, T_5)$ $R_{29} : (\mathbf{T_1}, T_4, \mathbf{T_6}, T_7)$ $R_{50} : (\mathbf{T_1}, T_4, \mathbf{T_6}, T_{16})$	$(\mathbf{T_1}, \mathbf{T_3}, \mathbf{T_6})$
	CMARTA-LRA	$R_{57} : (\mathbf{T_1}, T_2, T_4, T_5, \mathbf{T_6}, T_7, T_{10}, \mathbf{T_{13}})$ $R_{53} : (\mathbf{T_1}, T_7, \mathbf{T_8}, \mathbf{T_{12}}, T_{17})$ $R_{50} : (\mathbf{T_1}, T_4, \mathbf{T_6}, T_{16})$	$(\mathbf{T_1}, \mathbf{T_6}, \mathbf{T_8}, \mathbf{T_{12}}, \mathbf{T_{13}})$
$P_{57} : (T_1, T_2, T_3)$	ILP-PLSA	$R_{30} : (\mathbf{T_1}, T_7)$ $R_{59} : (\mathbf{T_1}, T_7)$ $R_{71} : (\mathbf{T_1}, T_6, T_7)$	$(\mathbf{T_1})$
	CFLA-PLSA	$R_{19} : (\mathbf{T_1}, \mathbf{T_2}, T_5, T_{11})$ $R_{23} : (\mathbf{T_1})$ $R_{76} : (\mathbf{T_1}, \mathbf{T_2}, T_5, T_6, T_7, T_9, T_{10}, T_{11}, T_{19})$	$(\mathbf{T_1}, \mathbf{T_2})$
	CMARTA-LRA	$R_{55} : (\mathbf{T_1})$ $R_{79} : (\mathbf{T_1}, \mathbf{T_2}, T_5, T_9, T_{10})$ $R_{15} : (\mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3}, T_4, T_5, T_6, T_7)$	$(\mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3})$
SIGIR Dataset			
Paper	Method	Team of reviewers	Covered Topics
$P_5 : (T_3, T_{12}, T_{23})$	ILP-PLSA	$R_{143} : (T_7, T_{24})$ $R_{152} : (T_1, T_2, T_4, T_9, T_{11}, \mathbf{T_{12}}, T_{13}, T_{22})$ $R_{176} : (T_7, T_{19}, T_{24}, T_{25})$	$(\mathbf{T_{12}})$
	CFLA-PLSA	$R_5 : (T_9, T_{11}, T_{14})$ $R_{16} : (T_1, T_2, \mathbf{T_3}, T_7, T_{10}, T_{11}, T_{19}, T_{20}, T_{24})$ $R_{133} : (T_2, T_5, T_{25})$	$(\mathbf{T_3})$
	CMARTA-LRA	$R_{106} : (T_{16}, \mathbf{T_{23}}, T_{24})$ $R_{16} : (T_1, T_2, \mathbf{T_3}, T_7, T_{10}, T_{11}, T_{19}, T_{20}, T24)$ $R_{62} : (T_1, T_4, T_9, T_8, \mathbf{T_{12}}, T_{13}, T_{15}, T_{21}, T_{25})$	$(\mathbf{T_3}, \mathbf{T_{12}}, \mathbf{T_{23}})$
$P_{54} : (T_1, T_3, T_{10})$	ILP-PLSA	$R_{128} : (T_4, T_5, T25)$ $R_{140} : (T_4, T_9, T_{11}, T_{12}, T_{13})$ $R_{150} : (T_7, T_9, \mathbf{T_{10}}, T_{11}, T_{19}, T_{21}, T_{25})$	$(\mathbf{T_{10}})$
	CFLA-PLSA	$R_{59} : (T_9, \mathbf{T_{10}}, T_{23}, T_{24})$ $R_{64} : (T_2, \mathbf{T_3}, T_7, \mathbf{T_{10}}, T_{11}, T_{17}, T_{19}, T_{24})$ $R_{124} : (T_4, \mathbf{T_{10}})$	$(\mathbf{T_3}, \mathbf{T_{10}})$
	CMARTA-LRA	$R_{182} : (\mathbf{T_1}, T_2, T_7, T_{11})$ $R_{170} : (T_2, T_7, T_8, \mathbf{T_{10}}, T_{11}, T_{15}, T_{19}, T_{22}, T_{24})$ $R_{107} : (\mathbf{T_3}, \mathbf{T_{10}}, T_{12}, T_{14}, T_{15}, T_{16}, T_{18}, T_{19}, T_{20}, T_{24})$	$(\mathbf{T_1}, \mathbf{T_3}, \mathbf{T_{10}})$

ments due to the possible unavailability of the few relevant reviewers later in the process. Consider two query papers in the PubMed dataset: query 159 with topics T_1, T_2 and T_9 and query 207 with topics $T_1, T_4, T_5, T_6, T_7, T_8, T_{10}, T_{12}$ and T_{13}). In Karimzadegan *et al.*'s method, paper 207 has higher priority than paper 159, since paper 207 has 9 topics and paper 159 has only 3. However, the order of paper in our method is reverse, since the topics of paper 159 has a lower value of the sum of relevance scores than the paper 207. Therefore, we get better coverage and average confidence than Karimzadegan *et al.*'s method in both queries.

Table 2.9: The concrete example of reviewers assignment of CMARTA-LRA, ILP and CFLA for one query in PubMed dataset

PubMed Dataset			
Paper Title	Method		
Carcinoma of the Male Breast	ILP-PLSA		
	R30	R59	R71
Pathology Surgery Mortality	**Pathology** Genetics	**Pathology** Genetics Metabolism	**Pathology** Genetics
	CFLA-PLSA		
	R19	R23	R76
	Pathology **Surgery** Diagnosis Radiography	**Pathology**	**Pathology** **Surgery** Diagnosis Metabolism Radiotherapy Therapy Radiography Chemistry
	CMARTA-LRA		
	R55	R79	R15
	Pathology	**Pathology** **Surgery** Diagnosis Radiotherapy Therapy	**Pathology** **Surgery** **Mortality** Drug Therapy Diagnosis Metabolism Genetics

Tables 2.9 and 2.10 also show the concrete examples with those topic words and the paper names in order to demonstrate how well the quality of our assignments are.These examples illustrate the effectiveness of our assignments in both SIGIR and PubMed datasets in comparison to the other baseline methods.

Table 2.10: The concrete example of reviewers assignment of CMARTA-LRA, ILP and CFLA for one query in SIGIR dataset

SIGIR Dataset			
Paper Title	Method		
Combining Content and Link for Classification using Matrix Factorization	ILP-PLSA		
	R128	R140	R150
Text Categorization Machine Learning Web Structures	Evaluation Efficiency Other IR applications	Evaluation Web IR Other Web Interactive IR and Feedback User Studies (behavior and modeling)	Summarization Web IR **Web Structures** Other Web Text mining Distributed IR Other IR applications
	CFLA-PLSA		
	R59	R64	R124
	Web IR **Web Structures** Collaborative Filtering Document representation	Clustering Machine Learning Summarization Other Web Other IR theory Text mining Document representation	Evaluation **Web Structures**
	CMARTA-LRA		
	R182	R170	R107
	Text categorization Clustering Summarization Other Web	Clustering Summarization Multimedia IR **Web Structures** Other Web Adaptive Information filtering	**Machine Learning** **Web Structures** Interactive IR and Feedback XML and structured data retrieval Other retrieval models Text mining

2.5 Conclusions and Future Work

Automating the process of assigning reviewers to papers can potentially improve the quality of assignments –and consequently that of the reviews– in conference organization and journal-submission reviewing. To that end, the overall objective of the process is, given a set of submissions, to identify, for each one of them, an expert team of reviewers, who can cover all or most of the different knowledge aspects of the paper in a complementary manner.

In this paper, we introduced the idea of using latent research areas in multi-aspect review-team assignment. Our method uses LSA to reduce the dimensionality of the collection of papers (authored by the potential reviewers) in term space and, subsequently, clusters the papers in this new space to infer "latent research areas". The latent research areas are then used to enhance multi-aspect review-team assignment, through adjusting the importance of the terms more relevant to the reviewers' research areas. Our formulation of the "multi-aspect review-team assignment" problem using latent research areas (MARTA-LRA) aims at identifying the best group of reviewers for each paper, optimizing the individual and group coverage of the paper's aspects, with preference to review teams that bring together diverse individual perspectives. We experimented with a greedy forward-selection approximation algorithms for optimizing our objective function to solve the constrained and unconstrained multi-aspect reviewer assignment. The results demonstrate that our objective function, MARTA-LRA, considerably improves the performance of multi-aspect reviewer assignment, in both constrained and unconstrained settings, over the state-of-the-art related works.

In the future, we plan to investigate additional functions for estimating the similarity between submissions and reviewers' expertise profiles, to examine the various parameters of our method and the relations between them in order

to develop a systematic method for automatically configuring these parameters for a given data set, and, to empirically validate the usefulness of our method for conference organizers in the context of a real conference.

Also, we can apply other identification methods for latent research areas along with a more comprehensive computational model for real world conferences which may include, e.g., the co-citation network structure besides the textual information

Chapter 3

Personalized and Explainable Aspect-based Recommendation using Latent Opinion Groups

Abstract

The problem of explainable recommendation—supporting the recommendation of a product or service with an explanation of why the item is a good choice for the user—is attracting substantial research attention recently. Recommendations associated with an explanation of how the aspects of the chosen item may meet the needs and preferences of the user can improve the transparency and trustworthiness of consumer-oriented applications, which is the motivation driving this research area.

Current methods are far from ideal because they do not necessarily consider the following issues: (i) users' opinions are influenced not only by individual aspects but also by the dependency between sentiments towards aspect; (ii) not all users place the same value on all aspects; and, (iii) any explanation are not provided for how the item aspects have led to the recommendation.

We introduce a personalized explainable aspect-based recommendation method that can address these challenges. To identify the aspects that a user cares about, our semantics-aware method learns the likelihood of an aspect being

mentioned in a user's review. To capture dependency between the users' sentiments towards an aspect, reviews that express opinions with similar polarities towards sets of aspects are clustered together in latent opinion groups. To construct aspect-based explanations, item aspects are rated according to their importance based on these latent opinion groups and the preferences of the target user. Finally, to provide a user with a (set of) useful recommendation(s) of an item, our method selects and synthesizes the aspects important for the target user.

We evaluate our method over two datasets from (a) Yelp and (b) Tripadvisor. Our results demonstrate that our method outperforms previous methods in both recommendation performance and explainability.

3.1 Introduction

Recommendation of products and services to users has long been a key feature of e-commerce websites. As users interact with the offerings of the website, recommender systems learn their preferences and entice them with more offerings that are potentially to their liking. However, most traditional recommendation methods are not transparent enough to be able to indicate to the users *why* a particular product or service is recommended to them [57]. Intuitively, such an explanation can make the recommendation more understandable, more trustworthy, and potentially more persuasive, which is the motivation driving current research on recommendation explanation.

The explanation associated with a recommendation should describe the reasons why the recommender system offers a specific item[1], among a catalog of potential offerings, to the particular user. In principle, four different categories of 'reasons' can be distinguished: *similar-user-based* explanations, *similar-item-based* explanations, *feature-based* explanations, and *aspect-based*

[1]Henceforth, we use the term item to refer to the products or services being recommended.

explanations. In similar-user-based explanations, the reasoning is that a group of users, considered similar to the target user, have assigned good ratings to the recommended item [11, 41, 54]. These explanations tend to not be very convincing since the users receiving the recommendations often know nothing about their "similar" users [20]. In similar-item-based explanations, the reasoning is that the recommended item is considered similar to at least one other item that the target user has liked previously [46]. Explanations based on item similarity are usually more intuitive for users, but do not necessarily give details about how the recommended item is similar to what the user liked previously [57]. The last two types of explanations are based on two different types of properties of the recommended item. In feature-based explanations, the term "feature" refers to some objective attribute of the recommended item, such as, for example, the lens of a camera; everything else, except the lens, being equal, any two users would agree on which camera is better [15, 59]. In contrast, in aspect-based explanations, the term *aspect* refers to some subjective attribute, such as, the affordability of the camera; even if two users agree on the quality of the camera, one may find it "affordable" and the other may find it "too expensive". This is because, these two example users have different personal preferences that lead them to different ratings of this subjective aspect of "affordability". Like the similar-item-based explanations, feature-based explanations do not use any previous reviews of the user to provide explanations for recommendation, which limits their persuasiveness. In our work, we focus on aspect-based explanations, aiming to use the users' preferences, as exemplified by their previous item reviews, to explain why the item is recommended based on the recommended item's aspects.

Recently, a number of aspect-based recommendation methods have been proposed [6, 22, 32]; however, none of them address our research objective of *explainable aspect-based recommendations*.

- Existing aspect-based recommendation methods [6, 22, 32, 51] typically describe aspects of an item using fixed sets of words. However, different users may use different words to describe the same aspect of an item. Current aspect-based explanation methods do not deal with the identification of semantic similarities between the different words that reviewers use when talking about the same aspects of an item [44]. For example, "view", "neighbourhood" and "place" are different words that users may use in their reviews to refer to the aspect "quality of location" of a hotel.

- Most existing methods [22, 32] use information about an item aspects to predict an overall rating of an item for a user, but do not provide any explanation for how these aspects have led to the recommendation.

- Some aspect-based recommendation methods [58, 52] assume that all users pay similar attention to all aspects. However, the interests, objectives, and lifestyles of users influence the aspects on which they choose to focus their reviews, and their sentiments about them. For example, a user may give the highest rating and strongly positive review to a clean and inexpensive hotel room, when another user may rate this room lower than another room that is similarly clean but is more expensive and more centrally located and easily accessible.

- Existing aspect-based recommendation methods [6, 22, 32] tend to estimate each aspect rating based on how the majority of the item reviews talk about this aspect. They ignore the possibility that a user's experience with an item may be influenced by a combination of aspects. For example, although the average rating of the "affordability" of a hotel is negative, some users may find the hotel appropriately priced because of its easily accessible location.

In this paper, we propose PRLOG, a method for *Personalized Recommendations, based on Latent Opinion Groups*, to address the above shortcomings. PRLOG first predicts the sentiments that the user is likely to express towards the items' aspects, and then combines these ratings to develop an overall rating for each candidate item. The higher-rated candidate items are then recommended and the predicted sentiments toward their aspects constitute the explanation for these recommendations.

More specifically, PRLOG starts by analyzing a set of input reviews to identify the aspects mentioned in these "raw" reviews, and the reviewers' sentiments about them. In some cases, the input reviews include explicit ratings for a predefined set of aspects, which makes this task fairly straightforward; in yet other cases, a state-of-the-art aspect-extraction method, such as the opinion parser [38], can be used to extract aspects and sentiments toward them from the review text. Next, word-embedded representations for aspects, users, and items are constructed from these aspect-annotated reviews. These representations enable our method to handle a variety of semantically similar words that reviewers may use when talking about item aspects. Furthermore, they are used alongside the explicit aspect ratings to provide the basis for estimating the importance score of different aspects for a user and for an item.

To capture the dependency between sentiments towards aspects, our method clusters reviews that express similar opinions about a set of aspects in *latent opinion groups*; each latent opinion group contains the reviews of the item that mention the same set of aspects with the same sentiment (positive or negative). Then, for any given item and given user, PRLOG predicts the sentiments that the user is likely to express toward the item's aspects, as well as the overall rating that this user is likely to assign to this item. To that end, our method uses aspects' importance scores that are calculated for that user and that item and information of latent opinion groups of the item.

We have evaluated the recommendation performance and the quality of explanation of PRLOG on two popular review datasets, Yelp [6] and Tripadvisor [50]. Our method performs well in predicting the overall rating of an item by the user, the set of aspects that the user would mention in his reviews of an item, and individual sentiments toward the item aspects, expressed in the user reviews.

The remainder of this paper is organized as follows. In Section 3.2, we review the relevant previous works on the problem of explainable recommendation systems. Then, our proposed method is introduced in detail in Section 3.3. Section 3.4 is devoted to reporting the experimental designs and the results. Conclusions are drawn in Section 3.5.

3.2 Related work

The term *explainable recommendations* refers to the output of a special type of recommendation algorithms that, in addition to providing the user with items that are highly likely to meet their needs, offer the reasons why these items were selected among other candidates. Current explainable recommendation systems can be classified into three broad categories, based on how they generate their recommendations and the 'reasons' to explain them [57], discussed in the following three subsections. The most recent category, i.e., "aspect-based recommendation explanation", relies on the identification of subjective aspects in the review text, of special importance to the review authors; the general area of aspect recognition is discussed in the last subsection.

3.2.1 Explanations based on item similarity or user similarity

Early explainable recommendation methods were based on user- and item-based collaborative filtering [57]. User-based collaborative filtering methods

[11, 41, 54] assume that, if a group of users, similar to the recommendation recipient, have assigned high ratings to an item, the recipient is highly likely to also like it. For example, Herlocker et al. [20] provide an aggregated histogram of the ratings of the users that are similar to the target user in order to explain why the item is recommended to the target user.

Item-based collaborative-filtering methods [46] recommend items similar to at least one other item that the target user has rated highly previously. For example, in Abdollahi and Nasraoui [1] a movie is recommended to the user because several other, similar films have been rated highly by the user.

Explanations based on user similarity are less convincing: since the users often know nothing about these "similar" users, the trustworthiness of the explanations is limited [20]. Ren et al [40] and Tsai and Brusilovsky [47] address this challenge by using social friends information to provide improved explanations. Furthermore, similar-user-based explanations [57] tend to weigh past reviews based on the similarity between the target user and the review authors; therefore all reviews from authors who are in "a cluster of similar users" are likely to receive the same (or very similar) weight, even if each review may talk about different aspects of the item.

Although explanations based on item similarity are usually intuitively more understandable, they do not necessarily explain how the recommended item is similar to what the user liked previously [47].

3.2.2 Feature-based explanations

Feature-based explanations are associated with content-based recommendation systems that match the user's profile with features of the candidate items [15, 12]. Features are the objective attributes of the items and are assumed to have the same meaning and value to all users. Similar to the category above, this type of recommendation explanations are not personalized. For example, the

features of a film includes its genre, actors and directors. Vig et al. [49] provide recommendations and explanations for the user by using these films features. The movies that match the features that a user likes are recommended to the user.

Demographic-based recommendation [60, 59] is another example of using feature-based explanations. Age, gender, and residence location can be used as users' demographic features. For example, "90% of customers with same gender bought this item". Zhao et al. [60] used the representation of users and items in the demographic feature space to recommend items to users. They also improved the performance of their methods by integration of demographic features and social media information of the users to recommend items in their next paper [59]. Unlike similar-item-based explanations, feature-based methods provide more detailed explanations about how the recommended item is related to the user. However, they still do no use any user-generated information to provide recommendations to users.

3.2.3 Aspect-based explanations

Using user-generated content that expresses the users' opinions about certain items and their various aspects, such as the reviews authored by each user, improves the generation of high-quality user profiles and relevant recommendations to each individual user [61]. Furthermore, this source of information can be used to generate more useful and persuasive explanations to help users make more informed decisions.

Aspects are subjective attributes of the items, whose importance differs from one user to the other; this means that the rating of aspects would be different for the users based on their personal preferences. The interests, experiences, and lifestyles of users affect which aspects they choose to discuss in their reviews, and in what words. Most of the previous methods [58, 52] do not

consider the user's preferences in order to recommend an item. However, the interests, objectives and lifestyles of users affect which aspects they choose to discuss in their reviews using a number of different semantically related words.

Aspects can be represented by sequences of words used in the reviews to describe aspects of an item. For example, the aspect "quality of location" can be described by the set of words {"place","view","locate",...} used in the review of a hotel. This allows aspects of an item and the user's sentiments toward them to be extracted from user-generated texts such as reviews.

Zhang et al [58] introduced an explicit factor model (EFM) to present aspect-sentiment world cloud as explanations to highlight the important aspects of an item. Ren et al. [40] adopted social relations in collaborative filtering model to predict item ratings based on the reviews of the user. McAuley and Leskovec [33] integrated Matrix Factorization and Latent Dirichlet Allocation (Hidden Factor and Topic model (HFT)) to use, simultaneously, ratings and the text of reviews to improve the rating prediction accuracy.

Wu and Ester [52] provide a word-cloud explanation for the recommended item on the three aspects of a hotel ("quality of location", "Cleanliness" and "Service") using topic modeling on textual reviews. These word-cloud based explanations such as "view - nice" highlight the performance of the aspect of the item for the user. However, the user's preferences are not considered in this type of aspect-based explanation.

Baumen et al. [6] proposed the Sentiment Utility Logistic Model (SULM) to integrate the user's sentiments toward the item aspects into a matrix factorization model. They do not only predict the overall rating of the item for the user. They also predict the aspects of an item that are important to the user. These important aspects of an item for the users are presented as explanations of a recommendation. Based on SLUM's assumption, the most important aspects to the user are the ones with highest ratings and most impact on the

overall rating. They assumed that user consider higher weight for their important item aspects in order to estimate the overall rating of that item. The method learns k-dimensional latent vectors corresponding to aspects, for users and items, to predict the sentiment utility values for all the aspects in a review and the impact scores of the aspects on the overall rating of the item. SULM uses the predicted sentiment utility value of each aspect in a review, and the impact score of the aspect on the overall rating of the item to estimate the overall rating of the item for the user.

Luo et al. [32] and Hou et al. [22] also propose an aspect-based matrix factorization approach to recommend items to a user by integrating ratings and review information. However, the transformation of aspect information into an overall item recommendation is usually opaque in that they do not explain why they recommend the item to the user. Although both methods use aspect information of products to make their recommendations more convincing, they miss the opportunity to communicate the rationale of their recommendations and gain the trust of users. They only provide two case studies to justify the performance of their recommendations. However, there is not a detailed explanation for the recommendation of the product as the output of these methods [32, 22].

To the best of our knowledge, SULM [6], is the only aspect-based recommendation method which provides an explanation for their recommendation using the aspects of items. However, the challenges mentioned in the introduction section 3.1 like using a fixed set of words to describe the item aspects and ignoring the dependency between item aspect ratings still exist in SULM [6]. Since SULM [6], on the other hand, also outperformed strong baselines in the overall rating prediction problem, SULM [6] is the most appropriate comparison partner in the evaluation of our method. Other models such as [17, 37, 39, 53] are not directly comparable with our method because

they focus only on the overall rating prediction problem.

3.2.4 Aspect extraction

Different users may discuss the same aspect using different terms in their review: for example, when describing their opinion about the aspect "quality of location" of a hotel or a restaurant, different users may refer to "the place", "the view", "the location", *etc.*, which necessitates a semantics-aware aspect extraction methodology. In general, there are three different categories of methods for extracting aspects from a collection of texts.

Frequency-based methods In a corpus of reviews for an e-commerce system, a limited set of words are more frequent than the rest of the vocabulary. In frequency-based aspect-extraction methods, these frequent words (usually only single nouns or compound nouns) are considered as aspects. A clear shortcoming of these methods is that not all frequent words are actually referring to aspects and infrequently discussed aspects may be ignored. In [23], all nouns and noun phrases occurring at least in one sentence are considered as aspects. Then, the frequency of each aspect is estimated. Also, different pruning strategies, like "every aspect needs to be followed by an adjective", are applied to improve the accuracy of the frequency-based aspect extraction methods.

Syntax-based methods Instead of considering the term frequencies, syntax-based methods identify aspects by means of syntactic relations that exist in sentences. A number of seed words are used as initial aspects. Then, if a certain noun has already been identified as an aspect, syntactically-related nouns are also considered as candidate words for describing aspects; for example, if 'photography' is known to be a film aspect, the sentence "the photography and script are the best in this film!" results in the term 'script' being included

as an aspect term. A popular algorithm that follows this approach is Double Propagation [38]

Unsupervised machine learning The previous two approaches share a common problem: people may use different words to refer to a particular concept, and thus, an aspect can be usually referred to with several words. In order to overcome this problem, extracted aspect words are usually grouped together, often by using lexicographical similarities, synonymy relationships, and taxonomy-based distances. Topic modeling methods are used to simultaneously extract and group aspects. The majority of topic models for aspect extractions exploit word co-occurrences within reviews along with word distribution differences to infer semantic clusters (topics) for the collection. For example, in [19], the model improves the quality of aspect extraction by exploiting the distribution of word co-occurrences through the use of word embedding methods.

3.3 Methodology

Our method works on a set of reviews for a specific class of items (products or services), which we call a domain. Example of domains include "hotels", "restaurants", "beauty salons", and others. In any considered domain, items can be characterised by their aspects. Users express their sentiments toward these aspects in their reviews about items. Each aspect is associated with a set of words that reviewers typically use to describe their opinions about an aspect in question.

Given this setting, we cast the problem of personalized explainable recommendation of an item to a user as the problem of predicting the sentiments that the user would express about the item's aspects if they were to write a review about this item. For each item aspect, a naïve way to make this predic-

tion can be a weighted average of the sentiments expressed towards this aspect in different reviews of the item. A review can be weighted by a measure of how well it addresses the aspects that are important to the user for that item. However, as we discussed in section 3.1, users' sentiments towards different aspects may depend on each other. The nature of these dependencies is not the same for all users, and these variations are evident in the users' reviews. Weighting each review individually would ignore these dependencies; instead, our method clusters together reviews that express the same sentiments towards a set of aspects. These clusters, which we call *latent opinion groups*, effectively capture the dependencies between sentiments expressed towards aspects. Latent opinion groups are described in more detail in section 3.3.2.

To predict the sentiment that a user is likely to express towards an aspect of an item, our method computes a weighted average of the sentiments expressed towards this aspect in different latent opinion groups of the item in question.

We will call these weights the *contribution scores* of latent opinion groups. The contribution score of each latent opinion group in the prediction process is computed based on two factors: (a) which aspects are mentioned in a latent opinion group for this item and how often, and (b) how important those aspects are to the user.

Given the predicted sentiments and importance of the item aspects for a user, we then also predict an overall rating that this user is likely to assign to this item.

In section 3.3.1, we will discuss how to model aspects and sentiments expressed in different kinds of reviews. Then, in section 3.3.2, we will introduce latent opinion groups and define a coverage score that captures how often an aspect is mentioned in a latent opinion group. After that, in section 3.3.3, we show how one can estimate the importance of item aspects for a user, given an item. Taking both, latent opinion groups and importance scores of aspects

(given an item and a user), we then define how to predict the user sentiments towards the item aspects in section 3.3.4. In a final step, we also predict an overall rating of an item by a user, as described in section 3.3.5.

3.3.1 Preliminaries and Conventions

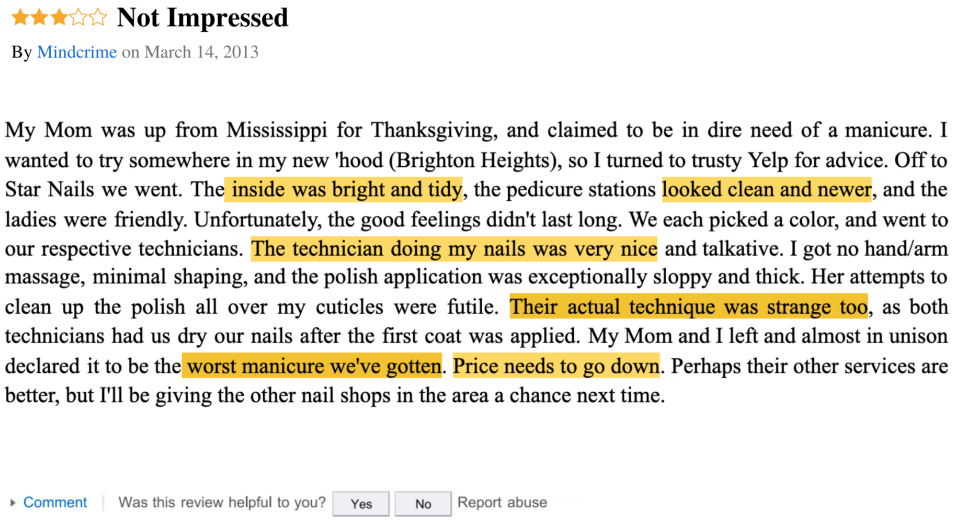
In principle, e-commerce sites offer one of two different kinds of reviews.

For the first kind of reviews, the e-commerce site explicitly identifies the relevant aspects, and users on these web sites are required to rate each of these explicit aspects for every item they review. On these "explicit aspect ratings" websites, users express their sentiment regarding each explicit aspect with their rating. As an example, consider the sample review of Tripadvisor shown in Figure 3.1a. This review includes a title, a short text, the overall rating for this item (which is 5 stars), and a rating from 1 to 5 for six different explicit aspects shown in the green box.

For the second kind of reviews, users can offer their overall rating for each item as well as their textual review, commenting on any aspect they may deem worthy of mention. As an example, consider the sample Yelp review shown in Figure 3.1b. This review only includes an overall rating (which is 3 stars) and some text that includes mentions of different aspects of the reviewed item and the reviewer's sentiments toward them (highlighted). In this case, the aspects mentioned and the user's sentiment towards them, must be extracted from the review text, using any of a number of aspect-extraction methods, discussed in section 3.2. A key observation here is that aspects may be represented by a variety of different words, which tend to appear together. For example, the words "food", "desert" and "soup" may be used to describe the aspect "food" of a restaurant.

As illustrated in the example shown in Figure 3.1, in domains with explicit aspect ratings, the accompanying review text is usually very short. For this

(a) A sample Tripadvisor review



(b) A sample Yelp review

Figure 3.1: Example of two different kinds of reviews

reason, when explicit aspect ratings are available, our method relies only on the explicitly rated aspects and does not attempt to extract additional aspects from the review text.

Given a review of an item by a user, we are interested in representing both the aspects of the item (either explicitly rated or extracted from a review text), as well as the user's sentiment towards these aspects: positive, negative, or neutral. We will represent the possible sentiments toward an item's aspects by elements of the set $\{+1, -1, 0\}$. For a given aspect, an associated value of '$+1$','-1' or '0 means that a review expresses a positive, negative or neutral

sentiment towards the aspect, respectively.

When the reviews include explicit aspect ratings, the user's sentiment is assumed to be positive for all aspects with a rating above the midpoint of the rating scale, negative for all aspects with a rating below the midpoint, and neutral otherwise.

When the reviews do not have explicit aspect ratings, the aspect-extraction process analyzes the user's review text to identify the aspects mentioned by the user as well as the polarity of the user's sentiment, i.e., positive, negative or neutral, towards each aspect.

We thus enhance the "raw" reviews by adding aspects and sentiments mentioned in these "raw" reviews either using explicitly provided aspect ratings or using a state-of-the-art aspect-extraction method. These enhanced reviews are called *aspect-sentiment-coded reviews*.

Given an aspect-sentiment-coded review r, we also use the notation T_r and A_r to refer to the two parts of r: the "raw" text and the set of aspects that are mentioned in r, respectively. Furthermore, we denote the sentiment expressed towards aspect a in A_r by $S(a, r)$, and we denote the number of times an aspect a is mentioned in the review text T_r by $f(a, r)$.

3.3.2 Latent Opinion Groups of each item

As discussed earlier, a user's sentiment for a particular item aspect often depends on their opinion about other aspects. To capture this intuition, our method organizes the reviews of an item into clusters that express the same "combination" of sentiments towards a set of aspects. These clusters, called *latent opinion groups (LOGs)* capture "similarly minded" reviewers sharing similar views about that particular item. Intuitively, a LOG may capture the fact that a group of users appreciates the service of a restaurant even though they may find that the price of the meal was high, while another LOG may

capture another group that finds the service insufficient given the high meal price.

A LOG for an item i, L is defined as a tuple containing all reviews R (for item i) that express the same "combination" of sentiments towards all aspects in a subset of aspects A, represented as a set of aspect-sentiment pairs AS, *i.e.*, $L =< R, A, AS >$.

The LOGs for an item i are discovered using the frequent itemset-mining method *Apriori* [3]. Apriori identifies all "itemsets" that occur in a database of "transactions" with a frequency above a minimum support threshold τ. In our case, the transactions are the aspect-sentiment-coded reviews for i, containing as "items", in the sense of Apriori, aspect-sentiment pairs. For each "frequent itemset" (*i.e.*, set of aspect sentiment pairs) $AS = \{< a_1, s_1 >, \ldots, < a_k, s_k >\}$ of length k, a LOG $L =< R, A, AS >$ is constructed, where R is the set of all aspect-sentiment-coded reviews that contain AS, and A is the set of aspects in the frequent set of aspect-sentiment pairs, *i.e.*, $A = \{a_1, \ldots, a_k\}$. Given a LOG L, we also use the notation R_L and A_L to refer to the two component of L: set of reviews and the frequent aspect set that belongs to L, respectively. Furthermore, we denote the sentiment expressed towards aspect a in A_L by $S(a, L)$. Figure 3.2 illustrates the different components of an LOG.

To capture "how often" an aspect is mentioned in a LOG, we define the *Coverage* of an aspect a by LOG L, $C(a, L)$, by aggregating the relative frequency with which a is mentioned in each review $r \in R_L$:

$$C(a, L) = \begin{cases} \sum_{r \in R_L} \frac{f(a,r)}{\sum_{a \in A_r} f(a,r)} & \text{if } a \in A_L\,; \\ 0 & \text{otherwise,} \end{cases} \tag{3.1}$$

We use the relative frequency with which an aspect is mentioned in a review (as opposed to just the count) so that reviews that mention aspect a more "prominently" contribute more towards the coverage of aspect a in L.

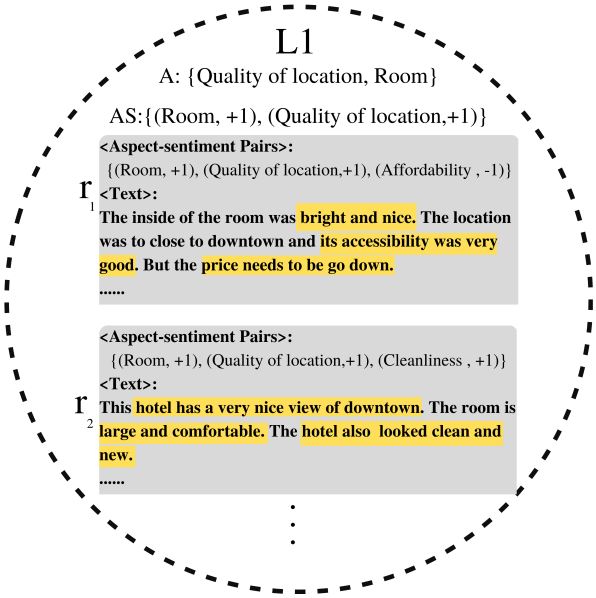


Figure 3.2: An Example to illustrate different components of an LOG

3.3.3 Importance of Item Aspects for a User

A key element of our method is the estimation of a score $\beta(a, u, i)$ that captures how important an aspect a of a given item i may be to a user u for whom a recommendation is being developed.

Our method assumes that reviewers consider the aspects they mention in their reviews as important. Based on this assumption, we will show how to estimate a *text-based importance score*, $\beta_{\text{Text}}(a, u, i)$ from review texts. When the reviews have explicit aspect ratings, the predefined aspects consist typically of single words or short phrases, which may or may not be explicitly mentioned in the review text. To estimate a text-based importance score, we will be using a "semantic-aware" method that is still able to estimate text-based importance scores for predefined aspects, even if the predefined aspect words or phrases are not directly used in a review text, but instead a phrase with similar meaning is used. However, since the accompanying review text is usually very short and users may not feel the need to comment on an aspect that they have

explicitly rated, we will also, in addition to the text-based aspect importance score, compute a *ratings-based importance score*, $\beta_{\text{Rating}}(a, u, i)$, based on the distribution of the rating scores of an aspect in the set of a user's reviews.

With both a text-based and a ratings-based importance score, our method uses a convex combination of $\beta_{\text{Text}}(a, u, i)$ and $\beta_{\text{Rating}}(a, u, i)$ to estimate the aggregate *aspect importance score* for a user and an item, $\beta(a, u, i)$ as follows:

$$\beta(a, u, i) = \lambda\beta_{\text{Text}}(a, u, i) + (1 - \lambda)\beta_{\text{Rating}}(a, u, i)\,. \tag{3.2}$$

The coefficient $\lambda \in [0, 1]$ determines the relative strength of the text-based and ratings-based importance scores in order to estimate the aggregate aspect-importance scores for the user given an item. In the absence of explicit aspect ratings, when only text is available, $\lambda = 1$, *i.e.*, $\beta(a, u, i) = \beta_{\text{Text}}(a, u, i)$.

How to effectively compute $\beta_{\text{Text}}(a, u, i)$ and $\beta_{\text{Rating}}(a, u, i)$ is described in detail in the following subsections 3.3.3 and 3.3.3. The aggregate aspect-importance scores and the LOGs are used together to predict the sentiments that a user is likely to express towards an item's aspects.

Text-based Aspect Importance Score

Users mention the aspects they deem important for an item in their reviews about that item. Let $P(a|u, i)$ be an estimate of how likely a user u is to talk about aspect a when reviewing item i. We can use $P(a|u, i)$ as a proxy for aspect a's importance score for user u and item i, based on textual reviews, *i.e.*, $P(a|u, i)$ is a proxy for $\beta_{\text{Text}}(a, u, i)$. While it is not obvious how to estimate $P(a|u, i)$ directly, we can first apply Bayes' rule:

$$\beta_{\text{Text}}(a, u, i) = P(a|u, i) = \frac{P(u, i|a)P(a)}{P(u, i)}\,. \tag{3.3}$$

Furthermore, users and items can be assumed to be independent from each other, since the domain is fixed to one specific class of items, e.g., hotels,

restaurants, etc. Then, the above equation can be rewritten as follows:

$$P(a|u,i) = \frac{P(u|a)P(i|a)P(a)}{P(u)P(i)}\,. \tag{3.4}$$

The terms $P(u)$, $P(i)$, and $P(a)$ in this expression are straightforward to estimate. $P(u)$ is the probability of u authoring a review, and it can be estimated as the ratio of u's reviews over the total number of reviews in the domain, R. $P(i)$ is the probability of a review being about item i, and it can be similarly estimated as the ratio of the number of reviews about item i, over the total number of reviews. $P(a)$ is the probability of a review mentioning aspect a, estimated as the ratio of the number of reviews that mention aspect a, over the total number of reviews.

Let us now consider $P(u|a)$ and $P(i|a)$. $P(u|a)$ is the probability of a user u authoring a review that mentions aspect a. Similarly, $P(i|a)$ is the probability that a review is about item i given that the review has mentioned aspect a. As discussed in section 3.3.1, aspects—which are either extracted from the reviews texts or provided explicitly—are represented by sets of words. Thus, $P(u|a)$ can be re-written in the following way:

$$P(u|a) = P(u|w_1, \ldots, w_k)\,, \tag{3.5}$$

where w_1, w_2, w_3, …, w_k are the k words that represent aspect a. Making the simplifying assumption that the k words are independent, $P(u|w_1, \ldots, w_k)$ can in turn be re-written as:

$$P(u|w_1, \ldots, w_k) = \prod_{j \in \{1,2,3,\ldots,k\}} P(u|w_j)\,.$$

In this form, we need to estimate the probabilities of a user u being the author of reviews that contain single words w_j.

Similarly, the probability that a review is about item i given that the review has mentioned aspect a, $P(i|a)$ can be re-written as follows:

$$P(i|a) = P(i|w_1, \ldots, w_k) = \prod_{j \in \{1,2,3,\ldots,k\}} P(i|w_j)\,. \tag{3.6}$$

In this form, we need to estimate the probabilities of reviews being about item i given that they contain single words w_j.

As mentioned before, different users may choose different words to describe the same item aspect. Therefore, our method applies a semantic-aware method to estimate the probabilities $P(u|w_j)$ and $P(i|w_j)$. To semantically match the different words that reviewers use when talking about the same aspects of an item, we adopt the *Semantic Entity Retrieval Toolkit* (SERT), a collection of neural entity-retrieval algorithms [48], which can be used to estimate the probabilities $P(u|w_j)$ and $P(i|w_j)$. This toolkit is designed to estimate the relevance score of a candidate expert c (entities to be retrieved) according to a textual query q, $P(c|q)$. Each query is presented by a sequence of words (e.g. representing research areas). The input of SERT is a document collection, domain-specific associations between documents and entities (*e.g.*, who wrote which document) and queries. The output of the toolkit is the relevance score of a an expert c for the query q, computed as the probability that documents containing the topics represented by the words in q appear in the documents that are authored by c. To do so, word-embedded vector representations for entities and words are learned using an unsupervised discriminative model based on the collection of documents and associations between documents and entities. We can adopt this approach to our problem by considering users and items as entities, reviews as associated documents and aspects words as queries. With this mapping, word-embedded vector representations for users, items and words are learned, and these word-embedded vector representations are then used to calculate the values for $P(u|w_j)$ and $P(i|w_j)$ using a log-linear model (See the SERT model [48] for details).

Ratings-based Aspect Importance Score

When ratings for explicit item aspects are available, our method also estimates a ratings-based aspect-importance score for a user and an item. Our approach for estimating these scores relies on the assumption that users are more discriminating in their set of reviews when rating item aspects that they care about. Intuitively, if a user does not consider a particular aspect very important when writing a review, they are more likely to rate this aspect with the same "default" value for all the items they review. On the other hand, if an aspect is important to the user, they are more likely to use a larger rating scale for this aspect in their item reviews, as the quality of this aspect is likely to vary among items. Thus, to measure the importance of an aspect to a user, we collect all the ratings of aspect a from all the reviews by user u in a set $ratings(a, u)$, and compute its standard deviation, *i.e.,* $\eta_{\text{rating}}(a, u) = stdev(ratings(a, u))$.

When it comes to the explicitly defined aspects of an item, we assume that they are all equally relevant for each item, giving them an equal weight $\eta_{\text{rating}}(a, i) = \frac{1}{|\mathcal{A}|}$, where $|\mathcal{A}|$ is the number of all aspects in the entire set of reviews. Lacking any further information, this is a reasonable assumption.

Using $\eta_{\text{rating}}(a, u)$ and $\eta_{\text{rating}}(a, i)$ we can define a ratings-based aspect importance score for a user u and item i, $\beta_{\text{Rating}}(a, u, i)$ as the product of these two factors:

$$\beta_{\text{Rating}}(a, u, i) = \eta_{rating}(a, u) \cdot \eta_{rating}(a, i)\,. \tag{3.7}$$

Using (3.7) and (3.3) the aggregated aspect-importance score (in equation (3.2)) of an aspect a can be estimated for a user u and an item i.

3.3.4 Predicting Sentiments towards Aspects

The final step in our method for recommending an item to a user is to predict the user's sentiments towards the various item aspects.

To capture the mentioned dependency between sentiments towards aspects, the user's sentiment toward each aspect is computed as a weighted average of the sentiments expressed towards that aspect in all LOGs of that item.

The weights assigned to the LOGs of an item—which we call contribution scores—are based on two factors:

1. Which aspects are mentioned in a LOG for this item and how often. The underlying intuition for considering how often an aspect is mentioned in reviews (compared to other aspects) is that when, for instance, there are two LOGs with identical frequent aspect sets, the LOG in which the reviews discuss the aspects more extensively (mention the aspects relatively more often) should have a higher weight.

2. How important those aspects are to the user. The underlying intuition for this factor is that typically, aspects are not all equally important to a user, and a LOG that mentions aspects that are more important to a user should have a higher weight for predicting the user's sentiments towards aspects.

Using these two factors, we estimate the contribution score $W(L, u, i)$ of a LOG L of an item i given a user u as follows:

$$W(L, u, i) = \sum_{a \in \mathcal{A}} C(a, L)\beta(a, u, i)\,, \tag{3.8}$$

where, $\mathcal{A}$ is the set of all aspects in the entire set of reviews and $\beta(a, u, i)$ is the aggregated importance score of aspect a for user u for item i, which is estimated based on previous reviews, as described above, using equation (3.2);

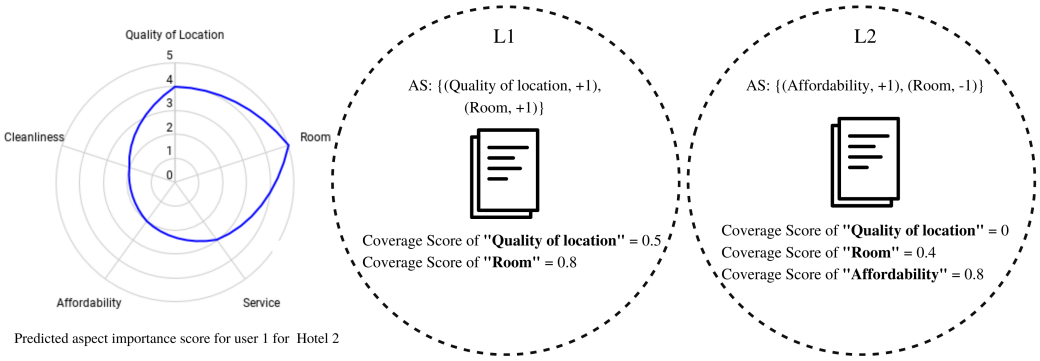


Figure 3.3: Example to illustrate different contribution scores of LOGs

$C(a, L)$ is the coverage score that captures "how often" aspect a is mentioned in the reviews of LOG L, as described in (3.1).

Figure 3.3 illustrates contribution scores of different LOGs. The importance scores of different item aspects for user u are depicted on the left side of of the figure. Given that user 1 essentially cares mostly about the 'Quality of Location' of the hotel and the 'Room' aspect, both LOGs should contribute to the prediction of user u's sentiments toward item aspects, since both contain these aspects, and LOG $L1$ should contribute more LOG $L2$ because $L1$ mentions the aspects that are important to user u more extensively.

Now,the sentiment towards aspect a given item i and user u, $S(a, u, i)$, can be estimated/predicted as the weighted average of the sentiments expressed towards aspect a over the set of all LOGs for item i, $\mathcal{L}_i$:

$$S(a, u, i) = \frac{\sum_{L \in \mathcal{L}_i} W(L, u, i) \cdot S(a, L)}{\sum_{L \in \mathcal{L}_i} W(L, u, i)} . \tag{3.9}$$

where $S(a, L)$ is equal to the sentiment expressed towards aspect a in A_L; and $W(L, u, i)$ is the contribution score of a LOG L of item i computed for user u.

3.3.5 Predicting an overall rating for an item for a user

Finally, the overall rating for an item i of a user u can be predicted as a linear combination of predicted sentiments of user u towards aspects of item i, weighted by the importance of aspects:

$$O(a, u, i) = \sum_{a \in A} \beta(a, u, i) \cdot S(a, u, i)\,. \tag{3.10}$$

These predicted overall ratings can then be used as a measure to rank and/or choose items to be recommended to a user. Accompanying the recommendation, we can provide the user with the item aspects that the user is likely to like, dislike or be neutral towards to, and thus explain why an item is recommended to a user or not recommended. Figure 3.4 illustrates examples for a personalized explanation of our method for a recommended and a disrecommended item.

User 1

Hotel 1

Predicted aspect importance score for user 1 for hotel1

We recommend Hotel 1 to you because you can expect the "Room" to be good and the "Quality of location" to be excellent in this hotel based on the experience of other users who visited this hotel and they talked the aspects that are important to you in their reviews .

(a) Recommendation

User 1

Hotel 2

Predicted aspect importance score for user 1 for hotel2

We disrecommend Hotel 2 to you because you can expect the "Cleanliness" to be unsatisfactory, the "Affordability" and the "Service" to be bad in this hotel based on the experience of other users who visited this hotel and they talked about the aspects that are important to you in their reviews .

(b) Disrecommendation

Figure 3.4: The sample personalized explanation of our method

3.4 Experiments

We conducted a set of experiments to investigate the following research questions:

Q1 *How does our method (when using a semantic-enhanced similarity measure) perform in predicting the aspects that users mention in their reviews? (aspect ranking performance)*

Q2 *How does our method (considering the aspect rating dependencies using latent opinion groups) perform in predicting the users' sentiments towards item aspects? (performance of prediction of sentiments towards aspects)*

Q3 *How does our method perform in predicting the overall rating of a user for an item?*

The experiments were conducted on an Apple MacBook Air with a 1.7 GHz Intel Core i7 processor and 8GBs RAM, and our algorithm is implemented in Python. For each dataset, we partition the collection of reviews into training and test sets in ratios of 80% and 20%, respectively. The split is done on a per-user basis, i.e., for each user, 80% of the reviews authored by that user is allocated for training and the remaining 20% for testing. Using the training dataset, we estimate the different components of $\beta_{\text{Text}}(a, u, i)$, $P(u|a)$, $P(i|a)$, $P(u)$, $P(i)$, and $P(a)$ using the review texts. We also use the ratings given to pre-defined aspects, when available, to estimate $\eta_r(a, u)$ and $\eta_r(a, i)$ in equation (3.7).

Table 3.1: Basic statistics of the four datasets: ANRI

Dataset	Users	items	Reviews	ANRI	ARNU	MNRI	MNRU
BeautySpa	319	1980	4653	2.35	29.17	35	56
Hotel	340	476	5493	11.53	32.31	161	121
Restaurants	3013	4746	62004	13.06	41.15	325	197
Tripadvisor	7453	10206	203020	19.89	54.48	799	75503

3.4.1 Datasets

We use four datasets in our experiments, including reviews from the 'Hotel', 'BeautySpa' and 'Restaurant' domains in Yelp [2], collected in several US cities over a period of 6 years, and the Tripadvisor dataset [3].

The number of reviews, items and users are reported in Table 3.1. The average number of reviews per item (ANRI), average number of reviews per user (ANRU), maximum number of reviews per item (MNRI), and maximum number of reviews per user (MNRU) are also reported in Table 3.1. We consider only users who have written more than 10 reviews.

To extract aspects and the users' sentiments toward these aspects in the Yelp datasets, we use Opinion Parser [6]. The number of extracted aspects for Hotel, BeautySpa and Restaurant applications are 19, 10 and 16, respectively. For the Tripadvisor dataset, the number of pre-defined aspects is 7.

3.4.2 Baselines

To demonstrate the performance of our method, we compare our model against SULM [6], a state-of-the-art, aspect-based explainable recommendation algorithm which is based on matrix factorization. SULM recommends an item alongside the most important aspects of that item to the user based on the predicted sentiment utility value of the aspect of the item for the user.

As shown in [6], SULM outperformed several previously proposed methods

[2]https://www.yelp.com/dataset
[3]http://times.cs.uiuc.edu/~wang296/Data

such as Probabilistic Matrix Factorization (PMF) [45], Explicit Factor Model (EFM) [58] and Rating-based Tensor Factorization (RTF) [9]. Since SULM [6] constitutes a strong state-of-the-art method that also outperformed the strong baselines in the overall rating prediction problem [45, 58, 9, 39] comparing PRLOG against the SULM [6] is sufficient for our purposes. Other methods, such as [17, 37, 39, 53], are not directly comparable with our method because they focus only on the overall rating prediction problem.

The parameters of SULM are set to the values that were recommended in its publication [6]. For SULM [6], we use the provided computer code published on GitHub [4] to reproduce the results over the Yelp datasets. We also apply SULM over the Tripdavisor dataset to compare the performance of our method with SULM over a dataset with pre-defined aspects.

3.4.3 Evaluation methodology

To evaluate the accuracy of the overall rating prediction, we use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) which are defined as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} \left(o_i - o_i{}^{`}\right)^2}{n}}\,; \tag{3.11}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n} \left|o_i - o_i{}^{`}\right|}{n}\,, \tag{3.12}$$

where o_i and $o_i{}^{`}$ are the predicted rating and the ground truth rating for the ith test element, respectively. n is the size of test set.

Classifying the ratings into "like" and "dislike" is advocated as an alternative approach to evaluate the overall rating prediction problem in previous works like [2]. Thus, we also frame the overall rating prediction as a binary classification problem by transforming the overall ratings into two classes:

[4] https://github.com/kobauman/SULM

'high' $\{4, 5\}$ and 'low' $\{1, 2, 3\}$. In this case, we evaluate how well PRLOG predict that a user would like an item (by giving it 'high' rating) or dislike it (by giving it a 'low' rating).

We evaluate the performance of PRLOG and SULM in this classification task using the accuracy measure, which is defined as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{n}, \tag{3.13}$$

where TP and TN are the number of instances for which the estimated and the ground truth overall ratings are both high or both low, respectively.

We also evaluate how well PRLOG predicts whether or not an item aspect appears in a user's review about that item. We use the aggregate aspect importance scores $\beta(a, u, i)$ to predict the list of aspects that a user would mention in their review. In particular, we first rank the item aspects for the user according to their aggregate aspect importance scores. Then, we select the top k of these ranked aspects and examine how many of them appear in their review. This measure is called *Precision@k*.

We consider the prediction of sentiments towards aspects as a classification problem. The users sentiments towards aspects are transformed into three classes: 'Positive','Neutral' and 'Negative'. Then, the accuracy measure is used to compare the performance of PRLOG in comparison to SULM: if a sentiment towards an aspect is greater than zero, zero or smaller than zero, the class of the sentiment towards the aspect is considered positive, neutral or negative, respectively.

3.4.4 Parameter Settings

The common parameter of our method for both datasets is the minimum support used by the Apriori algorithm, τ. Increasing the value of τ leads, in general, to fewer "frequent" LOGs for an item. Thus, we expect that larger

values of τ would lead to a decreased performance of recommendations for items with a small total number of reviews. Therefore, we recommend using small values for τ.

The dataset with the smallest total number of reviews and average number of reviews per item is the 'beautySpa' dataset. To ensure that for each dataset, including 'beautySpa', an itemset (LOG) has at least two supporting reviews to be considered "frequent", we set τ equal to 0.03 for all datasets.

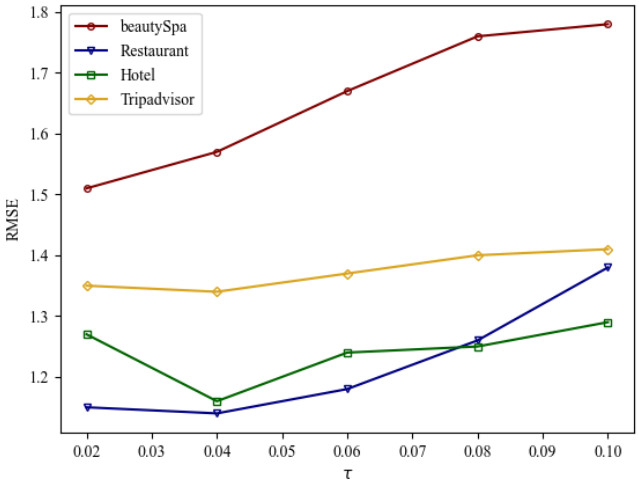
Figure 3.5 shows the RMSE and MAE of our method for different values of τ, supporting our expectation that, in general, increasing the value of τ lead to a decrease in performance (increase in error).

In the Tripadvisor dataset—which has pre-defined aspects, we have another parameter, λ, which is the parameter of the convex combination in equation (3.2) for the aggregated aspect importance score for a user for an item. We set this parameter to 0.5, assigning equal weight to the importance of text-based estimation and the estimation based on users' explicit ratings. This choice strikes a good balance between the two sources in lack of further information.

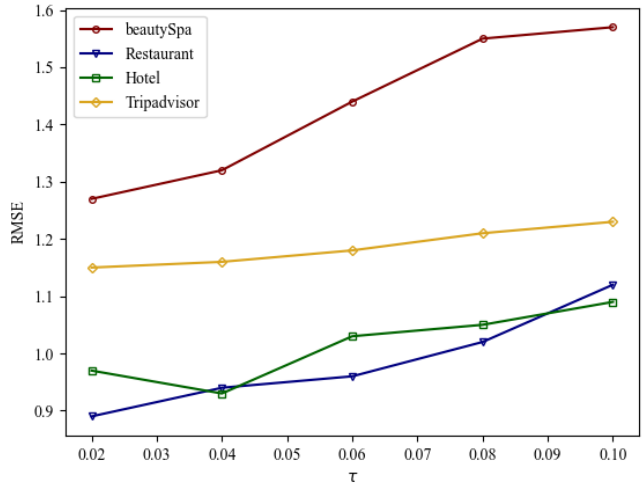
3.4.5 Results

Aspect ranking performance

Table 3.2 reports the average precision of predicting the top 3 and top 5 important aspects of an item for test users using PRLOG and SULM. Our method significantly outperforms (based on t-test with p-value < 0.05) SULM in predicting the set of aspects that a user would mention in their reviews of an item (identification of important aspects of an item for a user). This significant improvement in performance of PRLOG over SULM confirms the benefits of semantic matching when identifying important aspects for a user for an item, and directly modeling the aspect importance scores using details



(a) RMSE



(b) MAE

Figure 3.5: Error of our method using various settings for τ for the Tripadvisor dataset, and Restaurant, BeautySpa and Hotel applications of the Yelp dataset

of the reviews, rather than only considering the overall effect of an aspect rating on the overall rating.

Performance of prediction of sentiments towards aspects

To explain why an item is recommended or dis-recommended to a user, PRLOG provides the user with the item's aspects that the user is likely to like, dislike and be neutral towards. To gauge the explainability performance of our

Table 3.2: Comparison of aspect ranking performance of our method and SULM

Dataset	Method	Precision@3	Precision@5
BeautySpa	PRLOG	**0.36**	**0.30**
	SULM	0.22	0.19
Hotel	PRLOG	**0.42**	**0.38**
	SULM	0.40	0.32
Restaurant	PRLOG	**0.26**	**0.42**
	SULM	0.19	0.16
Tripadvisor	PRLOG	**0.30**	**0.42**
	SULM	0.23	0.35

method, we measure the accuracy of the predicted sentiments towards aspects. Using this measure, we can compare PRLOG and SULM on different datasets.

Table 3.3 shows that PRLOG again significantly outperforms ((based on t-test with p-value < 0.05) SULM in this task. PRLOG improves the precision of SULM in the aspect rating prediction problem by 50%, 85% and 65% over "BeautySpa", "Hotel", "Restaurant" datasets, respectively. We also improve the accuracy of the aspect rating prediction problem from 0.15 into 0.52 in the Tripadvisor dataset. Our method weighs the reviews based on whether they mention the aspects of the item that are important to the user, in order to predict the sentiments of the user toward the aspects. In contrast, SULM weighs the reviews based on the similarity of their authors to the user. This can lead to errors, as two users may be overall comparatively similar to each other, but their opinions may diverge substantially on what is important regarding some items. We have also factored in the interdependency of sentiments towards aspects by using latent opinion groups, which is missing in SULM. These experiments support our claim that our approach leads to improved performance in predicting the sentiments of the user towards aspects of an item.

Table 3.3: Comparison of the aspect rating prediction performance

Dataset	Method	Precision
BeautySpa	PRLOG	**0.93**
	SULM	0.62
Hotel	PRLOG	**0.96**
	SULM	0.52
Restaurant	PRLOG	**0.96**
	SULM	0.58
Tripadvisor	PRLOG	**0.52**
	SULM	0.15

Table 3.4: Comparison of the overall rating prediction performance using RMSE and MAE

Dataset	Method	RMSE	MAE
BeautySpa	PRLOG	**1.53**	**1.26**
	SULM	1.66	1.44
Hotel	PRLOG	**1.17**	**0.91**
	SULM	1.31	1.07
Restaurant	PRLOG	**1.16**	**0.94**
	SULM	1.27	1.04
Tripadvisor	PRLOG	**1.35**	**1.16**
	SULM	1.47	1.31

Overall rating prediction performance

Table 3.4 shows that PRLOG also significantly outperforms SULM (based on t-test with p-value < 0.05) in predicting the overall rating of an item by a user, both when looking at RMSE or at MAE, in all test datasets.

These results demonstrate that the combination of (1) our more accurate determination of important item aspects for a user and (2) our improvement

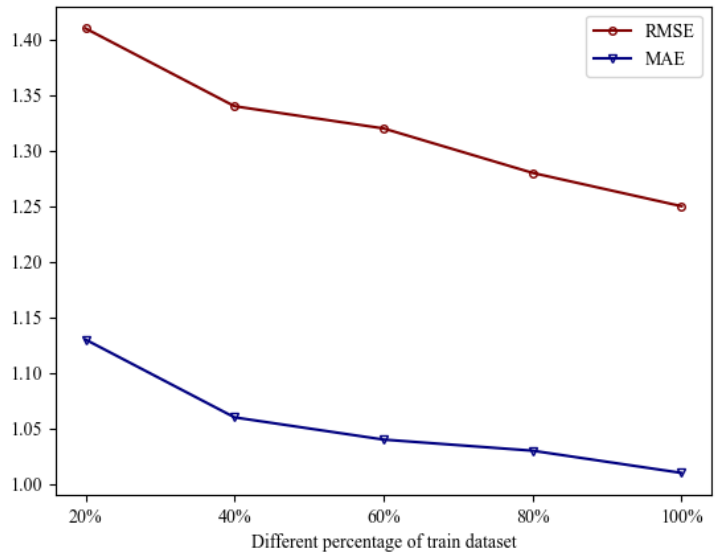


Figure 3.6: RMSE and MAE of PRLOG for different percentages of previous reviews for target users

in the performance of predicting the sentiments users express towards different item aspects, clearly improve also the accuracy of the overall rating prediction of our method.

To study how the number of reviews that a user previously wrote affects the performance of our method, we use the 'Restaurant' dataset. This dataset has a group of reviewers large enough for this kind of analysis: 50 reviewers, each which has written at least 30 reviews. For the experiment, when making a recommendation for one of these "target users" u, we only include a certain percentage of u's previous reviews into the training set. Figure 3.6 shows that the more previous reviews are available for a "target user" for which we want to make a recommendation, the better the results (the lower the prediction error).

Moreover, as Figure 3.6 shows, when we include only 20% of user's previous reviews in to the training set (considering few amount of user's past interaction as an example of the cold start problem), our methods' recommendation per-

Table 3.5: Comparison of the overall rating prediction performance using accuracy

Dataset	Method	Accuracy
BeautySpa	PRLOG	**0.66**
	SULM	0.54
Hotel	PRLOG	**0.59**
	SULM	0.52
Restaurant	PRLOG	**0.61**
	SULM	0.54
Tripadvisor	PRLOG	**0.75**
	SULM	0.53

formance is still reliable. Using latent opinion groups enables our method to consider different kinds of user opinions that may exist about the item coupled with their popularities in order to estimate user sentiments towards aspects even when we know nothing about the users' preferences.

In addition to RMSE and MAE, we use the accuracy measure to compare PRLOG to SULM. As we discussed in section 3.4.3, we can also consider the problem of overall rating prediction as a classification problem. Table 3.5 shows that our method outperforms SULM in the Accuracy of the predictions significantly (based on t-test with p-value < 0.05) as well, in all datasets. PRLOG improves the accuracy of predicting whether a target user likes or dislikes an item by 41%, 22%, 13%, and 13%, in the Tripadvisor, "BeautySpa", "Restaurant" and "Hotel" datasets, respectively.

In contrast to SLUM, which only uses explicit aspect ratings when predicting an overall rating in datasets with pre-defined aspects, PRLOG uses both sources of information, explicit aspect ratings and review texts to predict an overall rating. To study the contribution of both sources of information, we also compare the performance of our method in the overall rating predic-

tion problem using aspect weights calculated using (i) only the reviews texts, (ii) only the standard deviation of explicit aspect ratings by the users, and (iii) the combination of both as in equation (3.2).

Table 3.6: Comparison of the overall rating prediction performance using different sources

Dataset	Method	RMSE	MAE
Tripadvisor	$PRLOG_{both}$	**1.35**	**1.16**
	$PRLOG_{pre-defined}$	1.37	1.19
	$PRLOG_{text}$	1.80	1.24

Table 3.6 shows that using both sources of information improves the performance of our method in the overall rating prediction problem in both RMSE and MAE. In the Tripadvisor dataset, many reviews do not have meaningful texts. Also, the length of the texts of the reviews are too short in some cases Therefore, PRLOG cannot achieve an acceptable performance by using only review texts; however, using the combination of these sources enables PRLOG to obtain better results.

3.5 Conclusion

In this paper, we proposed PRLOG, a new method for explainable aspect-based recommendation. Our method exhibits better characteristics and accomplishes superior performance than existing state-of-the-art methods.

(i) It can handle the natural variation of wording that occurs when users write reviews in order to estimate aspect importance scores for a particular user and an item, which is novel among aspect-based recommendation methods.

(ii) Our method also provides better explanations than existing state-of-the-art by predicting not only the overall rating an item is likely to receive from a user, but also how the user will like or dislike individual aspects of an item.

(iii) PRLOG does not rely on restrictive and artificial assumptions around uniformity of importance of aspects across users or items and not even the assumption of uniformity of important aspects for a user across all items. These features make our method more general than existing state-of-the-art methods and enable it to provide better explanations that are more reliable and convincing to the users.

(iv) Our method also considers the dependency between sentiments towards aspects for different users. Rather than relying on a raw majority opinion to recommend an item to a user, PRLOG assigns different weights to different reviews based on what items aspects are mentioned in these reviews, how important these items aspects are for the user and how prevalent the opinion expressed in the review is.

The combination of the new features in our method results in better performance than what existing methods offer. We tested our method on two kinds

of datasets: one where the only information available through the reviews is the text of the reviews as well as one where the reviews have asked users to rate specific aspects of an item. For the former we tested on three different applications from the Yelp dataset and for the latter we used the TripAdvisor dataset. We showed that in all these datasets our method outperformed the best performing competitor method SULM significantly and by large margin.

In the future, we plan to explore (1) more sophisticated ways to model latent opinion groups, (2) ways to optimize the value for the parameter λ based on the amount of text available in review of applications with predefined aspects, and (3) more complex machine-learning models to predict the sentiments expressed towards aspects from latent opinion groups.

Chapter 4

Conclusions and future work

We studied two problems in multi-aspect paper-reviewer team assignment and personalized and explainable aspect-based recommendation of an item to a user as two applications of aspect-based recommendation.

4.1 Conclusion

First, we introduced a new framework for multi-aspect paper-reviewer assignment in both constrained and unconstrained settings. In this framework, we model the paper' thematic areas and the expertise of reviewers in term space; since the accuracy of topic modeling methods based on the small collection of short documents is not reliable. We also adjust the importance of the terms more relevant to the reviewers' research areas when computing the relevance of a particular reviewer for a paper. We provide a greedy forward-selection approximation algorithm to identify the best group of reviewers for each paper, optimizing the individual and group coverage of the paper's aspects, with preference to review teams that bring together diverse individual perspectives. In summary, our work makes two key contributions.

- It defines a new objective function for multi-aspect group-based paper-reviewer assignment in term space. This objective function considers the expertise of each reviewer, the overall expertise of the review team, as

well as the diversity of reviewers' expertise.

- It introduces a single efficient framework to solve both unconstrained and constrained problem variants.

We have empirically evaluated our method to demonstrate its superior performance relative to other state-of-the-art approaches.

Second, we proposed a new method for explainable aspect-based recommendation based on items reviewed by users.

We introduced a single framework which works for both explicitly defined aspects and implicitly extracted aspects from textual contents. When explicit aspects are not available, we apply the state-of the art aspect extraction methods to extract aspects from the reviewed text. In both problems that we solve in this thesis, we used the textual contents to model the aspects.

Then, our method estimates aspects importance scores for a particular user and item using a semantic-aware method. We also provide the user with the items aspects that the user is likely to like, dislike and be neutral towards, to explain why the item is recommended or disrecommended to the user. Rather than relying on a raw majority opinion to recommend an item to the user, our method assigns different weights to different reviews based on what items aspects are mentioned in these reviews, how important these items aspects are for the user and how prevalent the opinion expressed in the review is. The three main contribution of our work can be summarized as follows:

(i) It can handle the natural variation of wording that occurs when users write reviews in order to estimate aspect importance scores for a particular user and an item, which is novel among aspect-based recommendation methods.

(ii) Our method also provides better explanations than existing state-of-the-art by providing the predicted sentiments towards item aspects.

(iii) It does not rely on restrictive and artificial assumptions around uniformity of importance of aspects across users or items and not even the assumption of uniformity of important aspects for a user across all items.

(iv) Our method also considers the dependency between sentiments towards aspects for different users as well.

The study conducted in this thesis thus confirms that the combination of these new features in our method improves the performance of recommendation and the quality of explanation over the state-of-the-art related works.

In both problems studied here, we choose an item for recommendation recipients based on how good their preferences (such as their expertise or interest) match with item aspects. However, there are still two differences between these two problems. First, in contrast to the problem of recommending an item to the user in the context of e-commerce websites, "multi-aspect paper reviewer assignment" is an instance of the group formation problem. Thus, considering the interactions between the recommendation recipients in the process of matching the items and the recommendation recipients is critical.

Second, in the problem of "personalized and explainable aspect-based recommendation", we are trying to recommend or dis-recommend an item to the user. Thus, we use the user's sentiments towards aspects which shows how much a user likes or dislikes an item's aspects. However, in "multi-aspect paper reviewer assignment", we only consider the similarity between reviewers and papers in order to recommend the best matches for the papers.

Based on the study conducted in this thesis, we propose two different solutions to solve these problems which outperform their state-of-the-art related works.

4.2 Future work

An immediate direction for future research in multi-aspect paper-review assignment is to investigate additional functions for estimating the similarity between submissions and reviewers' expertise profiles. We also plan to examine the various parameters of our method and the relations between them in order to develop a systematic method for automatically configuring these parameters for a given data set.

Another possible future work is to empirically validate the usefulness of our method for conference organizers in the context of a real conference. Using co-citation network structure besides the textual information in our method also can be another possible way to extend our framework.

For the second problem, at the aspect extraction stage, we first suggest to use different state of the art methods to study how they affect the performance of our method.

Another possible extension to this work is to explore more sophisticated ways to model latent opinion groups. We can also build a more complex machine-learning model, rather than a linear regression, to predict the sentiments expressed towards aspects from latent opinion groups.

We suggest studying the correlation between the explicit aspect ratings and the predicted aspects importance scores for a user for an item.

We observed there may be room for improvement in predicting aspect importance scores by considering different scenarios based on explicit aspect ratings.

Another possible improvement to this work can be finding a general framework to mix two proposed methodologies in order to solve different kinds of aspect-based recommendation problems.

References

[1] Abdollahi, B., and Nasraoui, O. Using explainability for constrained matrix factorization. In *Proceedings of the Eleventh ACM Conference on Recommender Systems* (2017), pp. 79–83.

[2] Adomavicius, G., and Kwon, Y. New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems 22*, 3 (2007), 48–55.

[3] Agarwal, R., Srikant, R., et al. Fast algorithms for mining association rules. In *Proc. of the 20th VLDB Conference* (1994), vol. 487, p. 499.

[4] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. Network flows: theory, algorithms, and applications.

[5] Basu, C., Hirsh, H., Cohen, W. W., and Nevill-Manning, C. Recommending papers by mining the web. In *Proc. of the IJCAI99 Workshop on Learning About Users* (1999), pp. 1–11.

[6] Bauman, K., Liu, B., and Tuzhilin, A. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2017), pp. 717–725.

[7] Biswas, H. K., and Hasan, M. M. Using publications and domain knowledge to build research profiles: An application in automatic reviewer assignment. In *Int. Conf. on Information and Communication Technology 2007* (2007), pp. 82–86.

[8] Carbonell, J., and Goldstein, J. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proc. of the 21st Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval* (1998), pp. 335–336.

[9] Chen, X., Qin, Z., Zhang, Y., and Xu, T. Learning to rank features for recommendation over multiple categories. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval* (2016), pp. 305–314.

[10] Chowdhury, G. *Introduction to modern information retrieval*. Facet Publishing, 2010.

[11] Cleger-Tamayo, S., Fernandez-Luna, J. M., and Huete, J. F. Explaining neighborhood-based recommendations. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (2012), pp. 1063–1064.

[12] Cramer, H., Evers, V., Ramlal, S., Van Someren, M., Rutledge, L., Stash, N., Aroyo, L., and Wielinga, B. The effects of transparency on trust in and acceptance of a content-based art recommender. *User Modeling and User-adapted interaction 18*, 5 (2008), 455.

[13] Deng, H., King, I., and Lyu, M. R. Enhancing expertise retrieval using community-aware strategies. In *Proc. of the 18th ACM Conf. on Information and Knowledge Management* (2009), pp. 1733–1736.

[14] Dumais, S. T., and Nielsen, J. Automating the assignment of submitted manuscripts to reviewers. In *Proc. of the 15th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval* (1992), pp. 233–244.

[15] Ferwerda, B., Swelsen, K., and Yang, E. Explaining content-based recommendations. *New York* (2018), 1–24.

[16] Garcia, I., and Sebastia, L. A negotiation framework for heterogeneous group recommendation. *Expert Systems with Applications 41*, 4 (2014), 1245–1261.

[17] Hariri, N., Mobasher, B., Burke, R., and Zheng, Y. Context-aware recommendation based on review mining. In *ITWP@ IJCAI* (2011).

[18] Hashemi, S. H., Neshati, M., and Beigy, H. Expertise retrieval in bibliographic network: a topic dominance learning approach. In *Proc of the 22nd ACM International Conference on Information & Knowledge Management* (2013), pp. 1117–1126.

[19] He, R., Lee, W. S., Ng, H. T., and Dahlmeier, D. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (2017), pp. 388–397.

[20] Herlocker, J. L. *Understanding and improving automated collaborative filtering systems*. Citeseer, 2000.

[21] Hettich, S., and Pazzani, M. J. Mining for proposal reviewers: lessons learned at the national science foundation. In *Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* (2006), pp. 862–871.

[22] Hou, Y., Yang, N., Wu, Y., and Philip, S. Y. Explainable recommendation with fusion of aspect information. *World Wide Web 22*, 1 (2019), 221–240.

[23] Hu, M., and Liu, B. Mining opinion features in customer reviews. In *AAAI* (2004), vol. 4, pp. 755–760.

[24] Karimzadehgan, M., and Zhai, C. Integer linear programming for constrained multi-aspect committee review assignment. *Information Processing & Management 48*, 4 (2012), 725–740.

[25] Karimzadehgan, M., Zhai, C., and Belford, G. Multi-aspect expertise matching for review assignment. In *Proc. of the 17th ACM Conf. on Information and Knowledge Management* (2008), pp. 1113–1122.

[26] Kou, N. M., Hou, U. L., Mamoulis, N., and Gong, Z. Weighted coverage based reviewer assignment. In *Proc. of the 2015 ACM SIGMOD International Conf. on Management of Data* (2015), pp. 2031–2046.

[27] Lafferty, J., and Zhai, C. Document language models, query models, and risk minimization for information retrieval. In *Proc. of the 24th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval* (2001), pp. 111–119.

[28] Liang, S., and De Rijke, M. Finding knowledgeable groups in enterprise corpora. In *Proceedings of the 36th International ACM SIGIR Conference on Research and development in information retrieval* (2013), ACM, pp. 1005–1008.

[29] Liang, S., and de Rijke, M. Formal language models for finding groups of experts. *Information Processing & Management 52*, 4 (2016), 529–549.

[30] Liu, X., and Croft, W. B. Cluster-based retrieval using language models. In *Proc. of the 27th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval* (2004), pp. 186–193.

[31] Long, C., Wong, R. C.-W., Peng, Y., and Ye, L. On good and fair paper-reviewer assignment. In *2013 IEEE 13th International Conf. on Data Mining* (2013), pp. 1145–1150.

[32] Luo, H., Yang, N., and Philip, S. Y. Hybrid deep embedding for recommendations with dynamic aspect-level explanations. In *2019 IEEE International Conference on Big Data (Big Data)* (2019), IEEE, pp. 870–879.

[33] McAuley, J., and Leskovec, J. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems* (2013), pp. 165–172.

[34] Mimno, D., and McCallum, A. Expertise modeling for matching papers with reviewers. In *Proc. of the 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining* (2007), pp. 500–509.

[35] Moreira, C., and Wichert, A. Finding academic experts on a multisensor approach using shannon'92s entropy. *Expert Systems with Applications 40*, 14 (2013), 5740–5754.

[36] Neshati, M., Beigy, H., and Hiemstra, D. Expert group formation using facility location analysis. *Information Processing & Management 50*, 2 (2014), 361–383.

[37] Pero, Š., and Horváth, T. Opinion-driven matrix factorization for rating prediction. In *International Conference on User Modeling, Adaptation, and Personalization* (2013), Springer, pp. 1–13.

[38] Qiu, G., Liu, B., Bu, J., and Chen, C. Opinion word expansion and target extraction through double propagation. *Computational linguistics 37*, 1 (2011), 9–27.

[39] Qiu, L., Gao, S., Cheng, W., and Guo, J. Aspect-based latent factor model by integrating ratings and reviews for recommender system. *Knowledge-Based Systems 110* (2016), 233–243.

[40] Ren, Z., Liang, S., Li, P., Wang, S., and de Rijke, M. Social collaborative viewpoint regression with explainable recommendations. In *Proceedings of the tenth ACM international conference on web search and data mining* (2017), pp. 485–494.

[41] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work* (1994), pp. 175–186.

[42] Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. The author-topic model for authors and documents. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence* (2004), AUAI Press, pp. 487–494.

[43] Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics 20* (1987), 53–65.

[44] Salakhutdinov, R., and Hinton, G. Semantic hashing. *International Journal of Approximate Reasoning 50*, 7 (2009), 969–978.

[45] Salakhutdinov, R., and Mnih, A. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning* (2008), pp. 880–887.

[46] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (2001), pp. 285–295.

[47] Tsai, C.-H., and Brusilovsky, P. Explaining social recommendations to casual users: Design principles and opportunities. In *Proceedings of the 23rd International Conference on Intelligent User Interfaces Companion* (2018), pp. 1–2.

[48] Van Gysel, C., de Rijke, M., and Worring, M. Unsupervised, efficient and semantic expertise retrieval. In *Proceedings of the 25th international conference on world wide web* (2016), pp. 1069–1079.

[49] Vig, J., Sen, S., and Riedl, J. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent user interfaces* (2009), pp. 47–56.

[50] Wang, H., Lu, Y., and Zhai, C. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (2010), pp. 783–792.

[51] Wang, N., Wang, H., Jia, Y., and Yin, Y. Explainable recommendation via multi-task learning in opinionated text data. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (2018), pp. 165–174.

[52] Wu, Y., and Ester, M. Flame: A probabilistic model combining aspect based opinion mining and collaborative filtering. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining* (2015), pp. 199–208.

[53] Xu, Y., Lam, W., and Lin, T. Collaborative filtering incorporating review text and co-clusters of hidden user communities and item groups. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management* (2014), pp. 251–260.

[54] Zanker, M., and Ninaus, D. Knowledgeable explanations for recommender systems. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology* (2010), vol. 1, IEEE, pp. 657–660.

[55] Zhai, C., and Lafferty, J. Model-based feedback in the language modeling approach to information retrieval. In *Proc. of the 10th International Conf. on Information and Knowledge Management* (2001), pp. 403–410.

[56] Zhai, C., and Lafferty, J. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS) 22*, 2 (2004), 179–214.

[57] Zhang, Y., and Chen, X. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192* (2018).

[58] Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., and Ma, S. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (2014), pp. 83–92.

[59] Zhao, W. X., Li, S., He, Y., Wang, L., Wen, J.-R., and Li, X. Exploring demographic information in social media for product recommendation. *Knowledge and Information Systems 49*, 1 (2016), 61–89.

[60] Zhao, X. W., Guo, Y., He, Y., Jiang, H., Wu, Y., and Li, X. We know what you want to buy: a demographic-based system for product recommendation on microblogs. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (2014), pp. 1935–1944.

[61] Zheng, L., Noroozi, V., and Yu, P. S. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the tenth ACM international conference on web search and data mining* (2017), pp. 425–434.