Opinion-Based Entity Ranking using learning to rank

Shariq Bashir [a,*], Wasif Afzal [b], Abdul Rauf Baig [c]

[a] *Department of Computer Science, Mohammad Ali Jinnah University, Islamabad, Pakistan*
[b] *School of Innovation, Design and Engineering, Mlardalen University, Sweden*
[c] *Department of Information Systems, College of Computer and Information Sciences, Al Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia*

ARTICLE INFO

ABSTRACT

As social media and e-commerce on the Internet continue to grow, opinions have become one of the most important sources of information for users to base their future decisions on. Unfortunately, the large quantities of opinions make it difficult for an individual to comprehend and evaluate them all in a reasonable amount of time. The users have to read a large number of opinions of different entities before making any decision. Recently a new retrieval task in information retrieval known as Opinion-Based Entity Ranking (OpER) has emerged. OpER directly ranks relevant entities based on how well opinions on them are matched with a user's preferences that are given in the form of queries. With such a capability, users do not need to read a large number of opinions available for the entities. Previous research on OpER does not take into account the importance and subjectivity of query keywords in individual opinions of an entity. Entity relevance scores are computed primarily on the basis of occurrences of query keywords match, by assuming all opinions of an entity as a single field of text. Intuitively, entities that have positive judgments and strong relevance with query keywords should be ranked higher than those entities that have poor relevance and negative judgments. This paper outlines several ranking features and develops an intuitive framework for OpER in which entities are ranked according to how well individual opinions of entities are matched with the user's query keywords. As a useful ranking model may be constructed from many ranking features, we apply *learning to rank* approach based on genetic programming (GP) to combine features in order to develop an effective retrieval model for OpER task. The proposed approach is evaluated on two collections and is found to be significantly more effective than the standard OpER approach.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

With the development of social web content on the Internet, people are more likely to express their views and opinions. These opinions are important for individual users for making decisions. This trend is affecting more and more critical business processes such as customer support and satisfaction, brand and reputation management, product design and marketing [34,31,48,47]. This global trend has led to an evolution in the behavior of web users who are now increasingly reading reviews or comments before purchasing products or services [25,2,18,5]. There is now a massive growth of opinions on the web, ranging from opinions on businesses and products to diseases and people. While these opinions are meant to be helpful, the vast number of such opinions is overwhelming to users as there is just too much to read. For example, for popular products or hotels such as iPhone, Marriott or Hilton, the number of opinions can be up to hundreds or even up to thousands [31,17]. The large numbers of these opinions make it difficult for a potential customer to read and understand them in a limited time and to make an informed decision on whether to not to purchase a product/service. Thus, there is a need to develop information retrieval techniques in order to help users to exploit available opinions.

Opinion-Based Entity Ranking (OpER) is an information retrieval task for automatically ranking entities on the basis of opinions [17,7]. OpER directly ranks interesting entities based on how well the opinions on these entities are matched with the user's preferences. The idea is to represent each entity with the text of the opinions of all its users. Then, given a user's search query (where keywords of query represent aspects for entities), OpER can then rank the relevant entities based on how well opinions of entities

* Corresponding author. Tel.: +92 51 25128001.
E-mail addresses: shariq.bashir@jinnah.edu.pk (S. Bashir), wasif.afzal@mdh.se (W. Afzal), raufbaig@ccis.imamu.edu.sa (A.R. Baig).

(expressed by other users) match with the user's search preferences. In the presence of such automatic ranking system, the user does not need to read a large number of opinions available on all entities of a topic, but rather the user can now focus on a much smaller set of relevant entities that came on the top and roughly matches his/her preferences with the judgments of other users. Further, this type of ranking is flexible in the sense that it can be applied to any collection of entities for which opinions are available.

Previous information retrieval attempts based on OpER task determine the relevance (weights) of query keywords for a particular entity by assuming all of its opinions as a single field of text as commonly done in regular information retrieval [17]. These weights are aggregated and normalized for a specific entity so that a final score can be assigned to the entity for a particular topic. This assumption ignores the relevance of query keywords in individual opinions and does not model weights according to the subjectivity (judgments) of individual opinions. If a system does not have such a capability, then it is possible that an irrelevant entity may be ranked high just because of a greater matching of query keywords in large number of negative opinions. As we will show later in the paper, modeling the importance of query keywords in individual opinions significantly helps in improving the ranking effectiveness of OpER. In order to do this we propose a set of heuristically motivated ranking features. One subset of these features is based on standard document weighting schemes (such as TFIDF, BM25, PL2), while another subset of these features approximates subjectivity of query keywords when calculating relevance of entities. We call these features *keyword-opinion features*. We perform an effectiveness analysis of these keyword-opinion features to identify their correlation to relevance with the top ranked retrieved entities. Although single features show significant effectiveness, further improvement is possible by combining these features using *learning to rank approach* [21,15,42]. Thus, we employ the use of a machine learning approach to search for an optimal solution in the space of (keyword-opinion) feature combinations. At the end of learning, we evaluate the effectiveness of an optimal solution over entity collections in order to analyze to what extent it achieves a significant increase in effectiveness over the use of single features.

The remainder of this paper is structured as follow. Section 2 reviews related work on the OpER and other related areas. Section 3 starts with the description of the architecture of our proposed approach. This section also lists keyword-opinion features that we employ for ranking entities. Section 4 describes the setting for experiments, the collections, query sets and relevance judgments that we use to validate the effectiveness of our approach. Section 5 shows the effectiveness analysis of keyword-opinion features. In Section 6, we combine keyword-opinions features using learning to rank approach for automatically evolving effective retrieval model. Finally, Section 7 briefly summarizes the key lessons learned from this study.

2. Related work

Opinion-Based Entity Ranking (OpER) is a new retrieval task in information retrieval. We start the related work discussion with a brief introduction about the OpER task and then discuss several lines of related work that are similar to this domain.

Opinion-Based Entity Ranking (OpER): Ganesan and Zhai [17] proposed a novel concept of ranking entities on the basis of opinions. OpER directly ranks entities on the basis of the user's search preferences that are given in the form of query keywords and existing opinions on those entities. In their experiments, they analyzed the effectiveness of several state-of-the-art retrieval models for this task. One major extension that they proposed over existing models is how to model multiple aspects of entities so that entities could not get higher weight by just matching only a single or few aspects. In another set of experiments query expansion is used for further boosting the effectiveness of the search. Although their proposed extensions showed significantly higher effectiveness over base-line models. However, in all of their proposed extensions entity weights are calculated through query keywords match similar to as it is mostly done for regular information retrieval, and no importance is given on how to calculate the relevance of entities according to the subjectivity of query keywords. Due to this limitation their system may wrongly rank irrelevant entities at top ranked positions that have good match for query keywords in large number of negative opinions. Our proposed ranking system calculates relevance of entities by taking into account their relevance and subjectivity with query keywords.

Sentiment Analysis: Sentiment analysis or opinion mining is the area to identify and extract subjective information of opinions and then categorize opinions into different classes based on the subjectivity, such as *"positive"*, *"negative"* or *"neutral"* [31]. Generally the aim of sentiment analysis is to determine the judgment of a speaker or a writer on a topic or on the overall contextual polarity of a document. In recent years a large amount of research work has been done on sentiment analysis for predicting subjectivity ratings of a given text at the document, sentence, or feature/aspect level [32,16,30,27,45,29,36,19]. However, no research work has been done on developing a system that provides a direct support for ranking entities based on how well entities are matched with users' queries [17].

Opinion Retrieval: The aim of the opinion retrieval task is to consider the opinion mining problem from an information retrieval perspective and to retrieve and rank blog posts that are relevant and contain opinions about a given topic (query) [28]. It was introduced in TREC 2006 as opinion retrieval task. In recent years, much research has been done on blog opinion retrieval in which researchers follow the opinion retrieval definition as used in the TREC blog track. The proposed methods can be classified into two classes: (a) lexicon-based methods [41,46], and (b) classification-based methods [32,26,16].

Entity Ranking: Another research area that is conceptually close to OpER is Entity Ranking. Entity Ranking allows users to search and rank named entities from any kind of text sources [14,4]. An Entity Ranking system could list persons, dates and/or locations with respect to a given query. Similar to OpER, Entity Ranking also needs to calculate relevance scores of entities from a group of information sources rather than from a single source. Although both tasks are conceptually related, however, the focus of this research is to rank entities by modeling the subjectivity of different text segments (opinions) rather than trying to rank entities based on how well they match a topic.

Multifaceted Search: Multifaceted search has close resemblance with OpER due to the fact that in both tasks relevant information is searched with the help of aspects of the collection [23,40]. However, the following are some differences between the two tasks. First, in OpER task, desired information is retrieved from some loosely structured collection containing opinions about entities, whereas in multifaceted search desired information is searched from a highly structured or categorical collection. Furthermore, OpER is an ad-hoc faceted navigation system where users specify their search preferences in the form of entity aspects directly in the queries through keywords. For example, a user who is searching for a smart phone would provide a query with keywords *"very light"*, *"bright screen"*, *"long battery life"*, etc, whereas in regular faceted navigation these facets are explicitly defined and are usually fixed [23].