Improving Opinion-based Entity Ranking

Christos Makris and Panagiotis Panagopoulos
Department of Computer Engineering and Informatics, University of Patras, Patras, Greece

Keywords: Opinion Mining and Sentiment Analysis, Web Information Filtering and Retrieval, Searching and Browsing.

Abstract: We examine the problem of entity ranking using opinions expressed in users' reviews. There is a massive development of opinions and reviews on the web, which includes reviews of products and services, and opinions about events and persons. For products especially, there are thousands of users' reviews, that consumers usually consult before proceeding in a purchase. In this study we are following the idea of turning the entity ranking problem into a matching preferences problem. This allows us to approach its solution using any standard information retrieval model. Building on this framework, we examine techniques which use sentiment and clustering information, and we suggest the naive consumer model. We describe the results of two sets of experiments and we show that the proposed techniques deliver interesting results.

1 INTRODUCTION

The rapid development of web technologies and social networks, has created a huge volume of reviews on products and services, and opinions on events and individuals.

Opinions are an important part of human activity because they affect our behavior in decision-making. It has become a habit for consumers to be informed by the reviews of other users, before they make a purchase of a product or a service. Businesses also want to be able to know the opinions concerning all their products or services and modify appropriately their promotion and their further development.

The consumer, however, in order to create an overall evaluation assessment for a set of objects of a specific entity, must refer to many reviews. From those reviews he must extract as many opinions as possible, in order to create an observable conclusion for each of the objects, and then to finally classify the objects and discern those that are notable. It is clear that this multitude of opinions creates a challenge for the consumer and also for the entity ranking systems.

Thus we recognize that the development of computational techniques, that help users to digest and utilize all opinions, is a very important and interesting research challenge.

In (Ganesan and ChengXiang, 2012) it is depicted the setup for an opinion-based entity ranking system. The idea is that each entity is represented by the text of all its reviews, and that the users of such a system, determine their preferences on several attributes during the evaluation process. Thus we can expect that a user's query, will consist of preferences on multiple attributes. By turning the problem of assessing entities into a matching preferences problem, we can use, in order to solve it, any standard information retrieval model. Given a query from the user, which consists of keywords and expresses the desired characteristics an entity must have, we can evaluate all candidate entities based on how well the opinions of those entities match the user's preferences.

Building on this idea. in the present paper, we develop schemes which take into account clustering and sentiment information about the opinions expressed in reviews. We also propose a naive consumer model as a setup that uses information from the web to gather knowledge from the community in order to evaluate the entities that are more important.

2 RELATED WORK

In this study we deal with the problem of creating a ranked list of entities using users' reviews. In order

Makris C. and Panagopoulos P..
Improving Opinion-based Entity Ranking.
DOI: 10.5220/0004788302230230
In *Proceedings of the 10th International Conference on Web Information Systems and Technologies* (WEBIST-2014), pages 223-230
ISBN: 978-989-758-024-6

to approach effectively its handling, we are moving to the direction of aspect-oriented opinion mining or feature-based opinion mining as defined in (Ganesan and ChengXiang, 2012). In this consideration, each entity is represented as the total text of all the available reviews for it, and users express their queries as preferences in multiple aspects. Entities are evaluated depending on how well the opinions, expressed in the reviews, are matched user's preferences.

Regarding reviews, a great deal of research has been done on the classification of reviews to positive and negative based on the overall sentiment information contained (document level sentiment classification). There have been proposed several supervised in (Gamon, 2005), (Pang and Lee, 2004), unsupervised in (Turney and Littman, 2002), (Nasukawa and Yi, 2003), and also hybrid in (Pang and Lee, 2005), (Prabowo and Thelwall, 2009) techniques.

A related research area is opinion retrieval in (Liu, 2012). The goal of opinion retrieval is to identify documents that contain opinions. An opinion retrieval system is usually created on top of the classical recovery models, where relevant documents are initially retrieved and then some opinion analysis techniques are being used to export only documents containing opinions. In our approach we are assuming that we already have available the texts, which contain the opinions for the entities.

Another related research area is the field of Expert Finding. In this area, the goal is to recover one ranked list of persons, which are experts on a certain topic (Fang and Zhai, 2007), (Baeza-Yates and Ribeiro-Neto, 2011), (Wang et al., 2010). In particular, we are trying to export a ranked list of entities, but instead of evaluating the entities based on how well they match a topic, we use the opinions for the entities and we are observing how well they match the user's preferences.

Also, there has been much research in the direction of using reviews for provisioning aspect based ratings in (Wang et al.,2010), (Snyder and Barzilay, 2007). This direction is relevant to ours, because by performing aspect based analysis, we can extract the ratings of the different aspects from the reviews. Thus we can assess entities based on the ratings of the aspects, which are in the user's interests.

In section 6, we examine the naive consumer model as an unsupervised schema that utilizes information from the web in order to yield a weight of importance to each of the features used for evaluating the entities. We choose to use a formula that has some resemblance to those used in item response theory (ITL), (Hambleton et al., 1991) and the Rasch model (Rasch), (Rasch, 1960/1980). Item response theory is a paradigm for the design, analysis, and scoring of tests, questionnaires, and similar instruments measuring abilities, attitudes, or other variables. The mathematical theory underlying Rasch models is a special case of item response theory. There are approaches in text mining that use the Rasch model and item response theory such as (Tikves et al., 2012), (He, 2013). However our approach differs in the chosen metrics and in the applied methodology and we do not use explicitly any of the modeling capabilities of these theories. For other different approaches that take aspect weight into account see (Liu, 2012) and more specifically (Yu et al., 2011) however our technique is simpler and fits into the framework presented in (Ganesan and ChengXiang, 2012).

2.1 Novelty in Contribution

In (Ganesan and ChengXiang, 2012), they presented a setup for entity ranking, where entities are evaluated depending on how well the opinions expressed in the reviews are matched against user's preferences. They studied the use of various state-of-the-art retrieval models for this task, such as the BM25 retrieval function (Baeza-Yates and Ribeiro-Neto, 2011), (Robertson and Zaragoza, 2009), and they also proposed some new extensions over these models, including query aspect modeling (QAM) and opinion expansion. With these extensions they were given the opportunity to classical information retrieval models to detect subjective information, i.e. opinions, that exist in review texts. More specifically, the opinion expansion introduced intensifiers and common praise words with positive meaning, placing at the top entities with many positive opinions. This expansion favoured texts, and correspondingly entities, with positive opinions on aspects, which is the goal. *However this approach does not impose penalties for negative opinions.*

We further improve this setup by developing schemes, which take into account sentiment (section 4) and clustering information (section 5) about the opinions expressed in reviews. We also propose the naive consumer model in section 6.

3 THE PROBLEM OF RANKING ENTITIES AS INFORMATION RETRIEVAL PROBLEM

Consider an entity ranking system, RS, and a collection of entities E= $\{e_1, e_2, ..., e_n\}$ of the same kind. Assume that each of the entities in the set E, is accompanied by a big collective text with all the reviews for it, written by some reviewers. Let R = $\{r_1, r_2, ..., r_n\}$ be the set of all those texts. Then there exists an "1-1" relationship between the entities of E and the texts of R. Given a query q that is composed by a subset of the aspects, the RS system produces a ranking list of entities in E.

The idea to assess the entities, is to represent each entity with the text of all the reviews referred to in that entity. Given a keyword query by a user, which expresses the desirable features that an entity should have, we can evaluate the entities based on how well the review texts (r_i) match the user's preferences. So the problem of entity ranking becomes an information retrieval problem. Thus we can employ some of the known information retrieval models, such as BM25. This setup is being presented in (Ganesan and ChengXiang, 2012). In particular they employed the BM25 retrieval function (Baeza-Yates and Ribeiro-Neto, 2011), (Robertson and Zaragoza, 2009), the Dirichlet prior retrieval function (Zhai and Lafferty, 2001), and the PL2 function (Amati, and van Rijsbergen, 2002) and proposed some new extensions over these models, including query aspect modeling (QAM) and opinion expansion and they performed a set of experiments depicting the superiority of their approach. The QAM extension uses each query aspect to rank entities and then aggregates the ranked results from the multiple aspects of the query using an aggregation function such as the average score. The opinion expansion extension, expands a query with related opinion words found in an online thesaurus. The results of the experiments showed that while all three state-of-the-art retrieval models show improvement with the proposed extensions, the BM25 retrieval model is most consistent and works especially well with these extensions.

4 OPINION-BASED ASPECT RATINGS

Addressing the entity ranking problem as a matching preferences problem on specific features using an information retrieval model as presented in (Ganesan and ChengXiang, 2012) favors texts, and correspondingly entities, with positive opinions on aspects, which is the goal. *However this approach does not impose penalties for negative opinions.* Then we seek to examine the performance of known techniques for sentiment analysis. These techniques take into account the positive and negative opinions on the entity rating process. Our goal is to compare their performance with the performance of the information retrieval approach.

Instead of using the significance of the features (aspects) of the query for a review text (r_i) to create a ranking of the review texts, we attempt to use the sentiment information that exists in the opinions, expressed in the review texts, on specific features. In order to create a model which takes into account the sentiment information of the opinions that are expressed in the reviews by the reviewers, we use two simple unsupervised sentiment analysis techniques.

Given that each review text r_i contains the users' opinions for a particular entity, we apply simple aspect-based sentiment analysis techniques to extract the sentiment information about the features from the sentences, and aspect-based summarization (or feature-based summarization) to calculate the score of the features throughout the text.

4.1 Lexicon-based Sentiment Analysis

Let A = $\{a_1, a_2, ..., a_m\}$ be the set of the query aspects. We perform aspect level sentiment analysis. by extracting from the reviews r_i the polarity, $s(a_j)$, which is expressed for each of the aspect query keywords.

The total score the review r_i receives is the sum of the aspects sentiment scores, in this text, normalized by the number of aspects in the query. To calculate the sentiment score $s(a_j)$, we locate in the review text r_i the sentences on which any of the query aspects (a_j) appear and we assign to them a sentiment rating. The score $s(a_j)$ is the sum of the sentences scores on which there is the aspect a_j. To calculate the sentiment score of a sentence, we apply a pos tagging process (Tsuruoka), in order to tag every term with a pos tag, and we process only the terms that have been assigned the following pos tags (see List of part-of-speech tags at references):

{ RB / RBR / RBS / VBG / JJ / JJR / JJS }

as elements that usually contain sentiment information. For each of those terms we find the word's sentiment score in a sentiment dictionary (Liu, Sentiment Lexicon), 1 if it is labeled as a positive concept term, -1 if it is labeled as a negative concept term. Finally we sum the scores of all the terms. If the sum is positive the sentence's sentiment score is 1, while if the sum is negative the sentence's sentiment score is -1.

Also we take care of the negation and we use sentiments shifters. If in a sentence there is one of the following words: {not, don't, none, nobody, nowhere, neither, cannot}, we reverse the polarity of the sentence's final sentiment score.

4.1.1 Query Expansion

This automatic process reads every review text (r_i), sentence by sentence, and processes only those that contain one or more of the query's aspect keywords. But users usually use different words or phrases to describe their opinions on a feature (aspect) of the entity. To manage this effect we perform query expansion on the original query, which we seek to enrich with synonyms of the aspects $\sigma(a_j)$, as they are from the semantic network WordNet (Fellbaum, 1998).

For example, suppose a query q which consists of the aspect keywords $\{a_1, a_2, a_3\}$. For each keyword in q, we try to find synonymous terms $\sigma(a_j)$ using the semantic network WordNet and we import them to the query. The final query that emerges is $q = (a_1, \sigma_{1a1}, \sigma_{2a1}, a_2, \sigma_{1a2}, \sigma_{2a2}, \sigma_{3\,a2}, a_3, \sigma_{1\,a3})$. In this case the sentiment score of the aspect a_i, $s_{exp}(a_i)$, is the sentiment score of the term a_i plus the sentiments scores of all imported terms σj_{ai}, $s(\sigma j_{ai})$.

$$s_{\exp}(a_i) = s(a_i) + \sum_{j=1,2,\ldots,h} s(\sigma j_{a_i})$$

4.2 Syntactic Patterns based Sentiment Analysis

In this scheme we employ as base of our construction the algorithm that is presented in (Turney, 2002) in order to calculate the sentiment score of each sentence. This process performs analysis in a similar manner to the first. Like the first sentiment analysis technique, which is presented in section 4.1 above, so this technique reads every review text (r_i), sentence by sentence, and processes only those that contain one or more of the query's aspect keywords. However here, instead of using the sentiment score of the words in the sentence, we use the sentiment orientation (SO) of syntactic patterns in the sentence, which are usually used to form an opinion.

Syntactic patterns are identified within a sentence based on pos tags of terms, which appear in a specific order. The following are syntactic patterns that are used to extract two-word phrases:

1st Word	2nd Word	3rd Word(not extracted)
JJ	NN/NNS	anything
RB/RBR/RBS	JJ	not(NN/NNS)
NN/NNS	JJ	not(NN/NNS)
RB/RBR/RBS	VB/VBD/VBN/VBG	anything

In order to calculate the sentiment orientation (SO) of the phrases, we use the point wise mutual information (PMI). The PMI metric measures the statistical dependence between two terms. The sentiment orientation of a phrase is calculated based on its relationship with a set of positive reference words and a set of negative reference words. We use the set '+' = {excellent, good} as positive reference words and the set '-' = {horrible, bad} as negative reference words, and we enrich these sets with synonyms from the semantic network WordNet. Thus the sentiment, orientation of a phrase is calculated as follows:

$$SO(phrase) = \log_2 \frac{hits(phrase\ NEAR\ '+')\, hits('-')}{hits(phrase\ NEAR\ '-')\, hits('+')}$$

where the hits( ) for all the elements of a set are added. For example:

$$hits('+') = hits('excellent') + hits('good') + \sum_{j=1,2,\ldots,h} hits(\sigma j)$$

with σ_j being represented by the synonymous terms which is added into the set from the WordNet during the process.

5 SMOOTHING RANKING WITH OPINION-BASED CLUSTERS

In this scheme we strive to use clustering information around the reviews to improve the ranking of entities. We use the algorithm ClustFuse of Kurland (Kurland, 2006), which makes use of two components to provide a score to a document d, the probability's relevance of the text to the query and the assumption that clusters can be used as proxies for the texts, that rewards texts belonging "strongly" in a cluster which is very relevant to the query.

The ClustFuse algorithm uses cluster information to improve the ranking. In summary, the algorithm in order to create a document ranking to a query q, creates a set of similar queries to q, let it be $Q = \{q_1, q_2, \ldots, q_k\}$, and for each one of them receives a text ranking L_i. Then it tries to exploit clustering to all texts in the rankings L_i. Finally it produces a final text ranking using the following equation:

$$p(q \mid d) = (1-\lambda)p(d \mid q) + \lambda \sum_{c \in Cl\ (CL)} p(c \mid q)p(d \mid c)$$

In our case as CL we set all review texts (r_i). To create the set of queries Q = {q_1, q_2, ..., q_k} for each query q, we use combinations of synonyms of the terms from the semantic network WordNet. We employ the Vector space model for representing review texts (r_i), the cosine similarity as texts distance metric, the k-means algorithm for clustering, the FcombSUM (d, q) as fusion method (Kurland, 2006), and the BM25 metric for assessing r_i to the questions and produce the ranked list L_i. Also as r_is' features we use the sentiment ratings of aspects as they are obtained by the process described above in Section 4. So each cluster will consist of review texts with similar ratings in aspects. A detailed presentation of Kurland's scheme and an interpretation of the equations is presented in (Kurland, 2006).

6 THE NAIVE CONSUMER MODEL

In the previous schemes we employed a set of aspects keywords (features) as queries and evaluated the review texts on the relativity with those. But we consider that all aspects are equally important to be used in the assessment of the entities. For example, in the domain of the car we may say that the aspect "fuel consumption" is more important than the aspect "leather seats". It may not. We believe that the answer can only be given by the community. So we retrieve the appropriate information from the web; let D_{inf} be the set of those texts.

With this model we attempt to simulate the behavior of a consumer who is trying to assess entities from a specific domain and he knows some aspects, but he does not know the importance that each aspect has as criteria in the assessment. Usually such a user consults the web, for relevant articles in Wikipedia, in blogs, in forums, as well in sites that contain reviews of other users, to understand the importance that each aspect has.

We are attempting to collect the knowledge of D_{inf}, on which of the features are more important. We create a set of queries Q = {q_1, q_2, ..., q_k} containing the aspect query keywords. Each of the elements of Q is given as a query in a web search engine and the first ten results are being collected. Considering that search engines use a linear combination of measures such as BM25 and PageRank (Page, Larry, 2002), (Baeza-Yates and Ribeiro-Neto, 2011), we can say that all the texts (pages) that we collect are relevant to the entities' domain which we examine, and that those texts are important nodes in the graph of the web, so important for the community.

Concerning the significance of a term t in a document d, as part of a text collection, we can say that is calculated from the BM25 score of the term t in d. Having the D_{inf} set of all texts, we are trying to extract how important is each aspect query keyword (a_i) for the entity domain that we examine, by calculating a score of significance. For this computation we can apply many formulas, but we choose to use the following which contains the participation rate of the feature a_i in the score of reviews:

$$scoreA(a_i) = \frac{\sum_{d \in D_{inf}} BM25(a_i)_d}{\sum_{a' \in A} \left(\sum_{d \in D_{inf}} BM25(a')_d \right)} \qquad (1)$$

This formula tends to be similar with the Rasch model. In the analysis of data with a Rasch model, the aim is to measure each examinee's level of a latent trait (e.g., math ability, attitude toward capital punishment) that underlies his or her scores on items of a test. In our case the test is the assessment of the reviews, the examinees are the aspects, and the items of the test are the review texts.

Based on this idea we develop two models NCM1 and NCM2.

6.1 NCM1

Having the rate scoreA(t) for each aspect, expressing how important this feature is, when used to evaluate entities of a particular class, we can combine it with the term that expresses how important each aspect for a specific review text (r_i) as part of a text collection (R), in order to assess reviews in queries consisting of aspect keywords, as follows:

$$p(q,ri) = \sum_{t \in q} score(t)_{ri} * scoreA(t) \qquad (2)$$

where p(q, r_i) is the probability of relevance between the query and the review r_i, $score(t)_{ri}$ is the BM25 score of the word / aspect t for the review text r_i and scoreA(t) is derived from (1).

6.2 NCM2

NCM2 works as at NCM1 applying additionally the Kurland's schema (Kurland, 2006), to exploit any cluster organization, which may exist across the review texts. We employ equation (2) to assess the review texts to all queries $Q = \{q_1, q_2, ..., q_k\}$ and create the L_i lists, where scoreA(t) is the importance score of aspects for their use in the evaluation process of entities, as it is calculated from the set of texts collected from the web. We use the algorithm ClustFuse as shown previously, in section 5.

7 EXPERIMENTS

We performed two sets of experiments to test the performance of our schemes, using two different datasets respectively. The datasets consist of sets of entities that are accompanied by users' reviews, which come from online sites. The queries consist of aspects keywords. For each one of the queries we produce the ideal entities' ranking based on the ratings given by the users in aspects together with the texts of the reviews. It is calculated as the average of the ratings given by each user for a certain characteristic as the Average Aspect Rating (AAR). For queries that are composed by several aspects, the average of the AAR aspects' scores of the question is calculated as the Multi-Aspect AAR (MAAR). More specifically, consider a query $q=\{a_1, a_2, \dots, a_m\}$, with m aspects as keywords, and an entity e, then $r_i(e)$ is the AAR of the entity e for the i-th aspect. Consequently MAAR is calculated as follows:

$$MAAR(e,q)=\frac{1}{m}=\sum_{i=1}^{m} r_i(e)$$

In the first set of experiments we use the OpinRank Dataset, which was presented in (Ganesan and ChengXiang, 2012) and consists of entities, which are accompanied by reviews of users from two different domains (cars and hotels). The reviews come from the sites Edmunds.com and Tripadvisor.com respectively. We use the reviews from the domain of the cars which includes car models and the corresponding reviews, for the years 2007-2009 (588) and we perform 300 queries. The texts of the reviews have averaged about 3000 words.

In the second set of experiments we use a collection of review texts for restaurants from the website www.we8there.com. Each review is accompanied by a set of 5 ratings, each in the range 1 to 5, one for each of the following five features {food. ambience. service, value, experience}. These scores were given by consumers who had written the reviews. In the second set of experiments we use 420 texts with reviews, averaging 115 words, as published on the link: http://people. csail.mit.edu/bsnyder/naacl07/data/ and we perform 31 queries. In this set of experiments, we also compare the performance of our schemas with a multiple aspect online ranking model which is presented in (Snyder and Barzilay, 2007), and is based on the algorithm Prank which is presented by Crammer and Singer in (Crammer and Singer, 2001). This supervised technique has shown that it delivers quite well in predicting the ratings on specific aspects of an entity using reviews of users for this. To create an m-aspect ranking model we use m independent Prank models, one for each aspect. Each of the m models, are trained to correctly predict one of the m aspects. Having represented the review texts r_i as a feature vector $x \in R^n$, this model predicts a score value $y \in \{1, .., k\}$ for each $x \in R^n$. The model is trained using the algorithm Prank (Perceptron Ranking algorithm), which reacts to incorrect predictions during training, updating the weight (w) and limits (b) vectors.

We evaluate the performance of the our schemas to produce the correct entity ranking, calculating the nDCG at the first 10 results.

7.1 Experimental Results

Initially we compare the performance of the BM25 model with and without the AvgScoreQAM and opinion expansion extensions, which are presented in (Ganesan and ChengXiang, 2012). We note that in both sets of experiments we conducted, using the BM25 model with the proposed extensions gives better results. This is one of the main observations in (Ganesan and ChengXiang, 2012), and here it is verified. In our measurements, however, we did not observe the expected increase in performance at the first set of experiments. It should be noted that in our experiments we did not use pseudo feedback mechanism, as in (Ganesan and ChengXiang, 2012).

The experimental results depict that the use of sentiment information present in reviews on the evaluation of the entities, can be used equally well as the conventional information retrieval techniques, such as the use of the BM25 metric. Both sentiment schemas perform sentiment analysis at sentence level, each in a different way. In the first set of experiments, our schemas show that they almost perform the same, while in the second set the technique using syntactic patterns evinces better than that using the sentiment lexicon. We believe that the better performance of the syntactic patterns-based sentiment analysis in the second dataset is probably due to the fact that in the second dataset the reviews are small (average 115 words) and users express immediately and clearly their opinions forming simple expressions. It should be noted that although we chose simple unsupervised sentiment analysis techniques which do not perform in-depth analysis, we hoped that they would exceed in performance the information retrieval approach. This is because they have the ability to recognize and negative opinions, knowledge that ignores a model like BM25std+AvgScoreQAM+opinExp. However we do not observe this. We still believe that with more sophisticated sentiment techniques that can be accomplished. We must not forget that sentiment analysis techniques have to deal with the diversity of human expression. People use many ways to express their opinions, and there are many types of opinions. On the other hand, the performance and the simplicity of the information retrieval approach makes it an attractive option.

More we observe the performance of our two clustering models, the BM25std+Kurland and the BM25std+Kurland+opinion-based clusters, with which we seek to exploit clustering information from the review texts to improve the ranking of entities. In the first set of experiments BM25std+Kurland performs well, while in the second has low performance. The low performance of BM25std+Kurland is probably due to the fact that the texts in the second dataset are small in length (average words per text 115 words). So its representation in the vector space characteristics are similar, which introduces noise in the clustering process using the k-means algorithm. The BM25std+Kurland+opinion-based clusters scheme, performs well in both experiment sets. Also it is always better than that the standard BM25 formula and the BM25std+Kurland schema. Thus we can say that opinion based clustering can identify similar assessment behaviors to similar aspect queries among the entities and use this information to make a better entity ranking. This also shows that the opinion-based clustering is more suitable for an opinion-based entity ranking process than the content clustering.

Both of the naive consumer models show to perform better in the first set of experiments, while in the second set of experiments the schema that uses the Kurland technique and makes use of the cluster information, seems to overcome even the supervised classifier technique of the m-aspect prank model. So we see that it indeed plays an important role the knowledge of the importance of each attribute used in the entities assessment, and also the knowledge of the aspects groups as they are defined by the users' community.

Table 1: We present the average of the nDCG@10 of the questions for all schemes on the two set of our experiments.

method	**1st exp. set**	**2nd exp. set**
BM25std	0.87	0.936
BM25std+AvgScoreQAM+opinExp	0.88	**0.955**
lexicon-based SA	0.865	0.91
syntactic patterns-based SA	0.869	0.94
BM25std+kurland	0.88	0.90
BM25std+Kurland+opinion-based clusters	**0.89**	**0.956**
NCM1	**0.891**	0.938
NCM2	**0.893**	**0.96**
m-aspect Prank	-	**0.95**

8 CONCLUSIONS

In this paper we examined the problem of ranking entities. We developed schemes, which take into account sentiment and clustering information, and we also propose the naive consumer model. In order to supply more analytical hints we need more experiments for various application areas and this is a topic of future work however in this paper we aimed at providing a proof of concept of the validity of our approach. The information retrieval approach with the two extensions, the aspect modeling and the opinion expansion, presented in (Ganesan and ChengXiang, 2012), is a working and attractive option. The NCM model can be used to reveal more reliable entity rankings, thanks to the knowledge it extracts from the web. The opinion-based clustering schema can be also used to generate more accurate entity rankings. Regarding the sentiment analysis techniques, which are those that would probably give the complete solution on the entity ranking problem, for now, they are dependent on the level of analysis and on the characteristics of the opinionated text. The syntactic patterns based sentiment analysis technique in the second set of our experiments has better performance than the lexicon-based sentiment analysis. In the second dataset the reviews are small and users express immediately and clearly their opinions forming simple expressions, while in the first dataset the reviews are longer and opinion extraction becomes complex. Although there are datasets that contain short texts, such as twitter datasets, in which opinion extraction can be quite difficult and require techniques that perform deeper sentiment analysis.

REFERENCES

Amati, G., and van Rijsbergen, C. J.,. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst., 20(4):357{389, 2002.*

Baeza-Yates, R., and Ribeiro-Neto, *B. Modern Information Retrieval: the concepts and technology behind search*. Addison Wesley, Essex, 2011.

Crammer K., Singer Y., Pranking with ranking. *NIPS* 2001, 641-647.

Fang H. and Zhai C., Probabilistic models for expert finding. *ECIR 2007*: 418-430.

Fellbaum, C., editor. WordNet, an electronic lexical database. *The MIT Press*.1998.

Gamon M., Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. *COLING* (2005), pp. 841-847 .

Ganesan, K., and ChengXiang Z., Opinion-Based Entity Ranking. *Inf. Retr.* 15(2): 116-150 (2012).

Hambleton, R. K., Swaminathan, H., and Rogers, H. J. Fundamentals of Item Response Theory. *Newbury Park, CA: Sage Press* (1991).

He, Q., *Text Mining and IRT for Psychiatric and Psychological Assessment.* Ph.D. thesis, University of Twente, Enschede, the Netherlands. (2013).

Kurland Oren, *Inter-Document similarities, language models, and ad-hoc information retrieval*. Ph.D. Thesis (2006).

Liu Bing, *Opinion/Sentiment Lexicon* http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html.

Liu Bing, *Sentiment Analysis and Opinion Mining.* Morgan & Claypool Publishers, 2012 .

Nasukawa T. and Yi J., Sentiment analysis: capturing favorability using natural language processing. *Proceedings K-CAP '03 Proceedings of the 2nd international conference on Knowledge capture, pp.* 70-77 .

Page, Larry, PageRank: Bringing Order to the Web. *Proceedings, Stanford Digital Library Project, talk. August 18, 1997* (archived 2002).

Pang B. and Lee L., A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *Proceedings, ACL'04 Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics.*

Pang B. and Lee L., Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *ACL 2005*.

Prabowo R. and Thelwall M., Sentiment analysis: A combined approach. *Journal of Informetrics, Volume 3, Issue 2, April 2009, Pages 143–157.*

Rasch, G., Probabilistic Models for Some Intelligence and Attainment Tests, (Copenhagen, Danish Institute for Educational Research), with foreward and after word by B. D. Wright. The University of Chicago Press, Chicago (1960/1980).

Robertson, S., Zaragoza, H., *The Probabilistic Relevance Framework: BM25 and Beyond.*, Foundations and Trends in Information Retrieval 3(4): 333-389 (2009).

Snyder B. and Barzilay R., Multiple aspect ranking using the good grief algorithm. *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pp. 300-307.*

Tikves, S., Banerjee, S., Temkit, H., Gokalp, S., Davulcu, H., Sen, A., Corman, S., Woodward, M., Nair, S., Rohmaniyah, I., Amin,A., A system for ranking organizations using social scale analysis, *Soc. Netw. Anal. Min., (2012).*

Titov, Ivan and Ryan McDonald, A joint model of text and aspect ratings for sentiment summarization., *In Proceedings of Annual Meeting of the Association for Computational Linguistics (AC L-2008)., (2008a).*

Titov, Ivan and Ryan McDonald, Modeling online reviews with multi-grain topic models., *In Proceedings. of International Conference on World Wide Web (WWW-2008). 2008b. doi:10.1145/1367497.1367513.*

Tsuruoka Yoshimasa, *Lookahead POS Tagger*, http://www.logos.t.u-tokyo.ac.jp/~tsuruoka/lapos/.

Turney, P.D, *Thumbs up or thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews.* ACL, pages 417-424. (2002).

Turney P. D. and Littman M. L., Measuring praise and criticism: Inference of semantic orientation from association. *In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL 2002).*

Wang H., Lu Y., and Zhai C., Latent aspect rating analysis on review text data: a rating regression approach. *In Proceedings KDD '10 Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 783-792 (2010).*

Yu, Jianxing, Zheng-Jun Zha, Meng Wang, and Tat-Seng Chua, Aspect ranking: identifying important product aspects from online consumer reviews. *In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. (2001).*

Zhai, C. and Lafferty, J., A study of smoothing methods for language models applied to ad hoc information retrieval. *In Proceedings of SIGIR'01, pp. 334–342 (2001).*

Rasch, *The Rasch model,* http://en.wikipedia.org/wiki/Rasch_model.

ITL, *Item Response Theory,* http://en.wikipedia.org/wiki/Item_response_theory.

List of part-of-speech tags, http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html.