

Using Social Data for Personalizing Review Rankings

Vaishak Suresh*
Intuit Inc.
Mountain View, CA, USA

Syeda Roohi*
HP
Palo Alto, CA, USA

Magdalini Eirinaki
Computer Engineering Department
San Jose State University, CA, USA

Iraklis Varlamis
Department of Informatics and Telematics
Harokopio University of Athens Greece

ABSTRACT

Almost all users look at online ratings and reviews before buying a product, visiting a business, or using a service. These reviews are independent, authored by other users, and thus may convey useful information to the end user. Reviews usually have an overall rating, but most of the times there are sub-texts in the review body that describe certain features/aspects of the product. The majority of web sites rank these reviews either by date, or by overall “helpfulness”. However, different users look for different qualities in a product/business/service. In this work, we try to address this problem by proposing a system that creates personalized rankings of these reviews, tailored to each individual user. We discuss how social data, ratings, and reviews can be combined to create this personalized experience. We present our work-in-progress using the Yelp Challenge dataset and discuss some first findings regarding implementation and scalability.

Keywords

Opinion mining; feature ranking; sentiment analysis; Recommendation Engine

1. INTRODUCTION

With more and more businesses selling their products or advertising their services online, customers rely on word of mouth in the form of customer reviews to make their decisions. Most of the current websites that feature product/business/service reviews list these reviews in reverse chronological order, or by employing heuristic metrics (e.g. ranking higher reviews of “super users”, i.e. users with many reviews, or those with the most “helpful” votes). However, such a generic ranking requires from users to read or at least scan the tens or hundreds of reviews for one product/business/service.

Moreover, different people value different aspects of the same product/business/service. For example, when searching for a digital camera, one might be interested in the price and size, whereas another user may value the ease of use. Similarly, when searching for a good Italian restaurant, one user might value the ambience and wine list of a place, while another might prefer restaurants that are family-friendly. Ideally, users would like to be presented with only these reviews that highlight the qualities of a product/service that they value.

In this work, we present a system framework that addresses the above issue. In a nutshell, we create user profiles that reflect each user’s preferences for specific restaurants and restaurant qualities (e.g. food, ambience, etc.). The profiles are created using the

rating data as well as implicit preference as identified by applying aspect-based opinion mining to the reviews. Using these profiles, we identify similar users and rank their reviews for new restaurants higher. We also integrate the social network of the user, identifying those friends who have similar preference patterns with the active user, and highlight their reviews. Therefore, for the same restaurant, two different users will see a different list of reviews. The system is accompanied by a user-friendly interface that also highlights the main aspects of each review such that the user does not have to read the full text. To achieve this, we employ aspect-based opinion mining and neighborhood-based collaborative filtering techniques and integrate them in our system.

We also present a system prototype, built using the Yelp dataset¹ to demonstrate a first approach to this interesting problem [RSE14]. Without loss of generality, we focus on restaurant review recommendations, however our approach is easily extended to any other product/business/service as long as reviews, ratings, and an underlying social network are available.

Since scalability is also of concern, we also discuss our experience with deploying the first prototype of our system online. We should stress that this is a work in progress and our focus in this paper is to introduce this mash-up idea along with an initial approach to the problem, as well as our thoughts on how such a system could be further enhanced.

The rest of the paper is organized as follows: we present our system’s design in detail in Section 2. We provide a first-cut approach on extending the proposed model using social network connections and feedback in Section 3. Some discussion on the prototype implementation and performance benchmarks is included in Section 4. An overview of the related work is provided in Section 5 and we conclude with our plans for future work in Section 6.

2. SYSTEM DESIGN

The system architecture is shown in Figure 1. It comprises two main modules, an offline processing module, where the user profiles are being generated and the feature extraction and rating happens, as well as an online module, that generates real-time recommendations.

¹ http://www.yelp.com/dataset_challenge/

* This is work performed while the authors were affiliated with San Jose State University

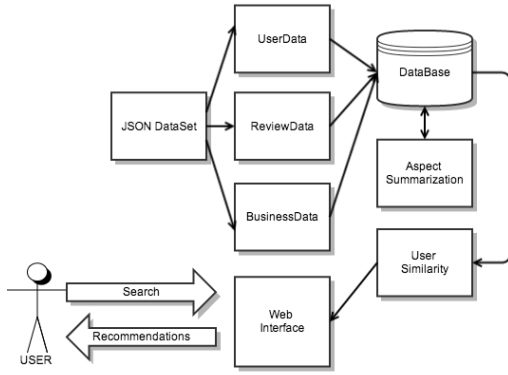


Figure 1: System Architecture

2.1 Offline Processing

There are two phases of offline processing: namely aspect summarization and user preference generation.

2.1.1 Aspect Summarization

This module aims at extracting the important features from each review, along with their polarity weight. To perform this we employ the subjectivity lexicon [WWH05] in order to map weak and strong positive and negative to numeric values (ranging from -4 to +4). Using a master list of positive and negative opinion words from an opinion lexicon [L12] we created a list of negation words (not, no, nothing etc.) which inverse the sentiment, and intensifiers (too, very, so, etc.), which increase the intensity of the sentiment (these are referred as “TOO words” in our algorithm).

More specifically, words of each review are tagged using the default POS (parts of speech) tagger from NLTK², a natural language processing Python package. This is done using the Treebank corpus. The text augmented with tags is then split into sentences and then into words. Each word is then examined to determine its type.

If the word is POS-tagged as an adverb or an adjective, it is considered as an *opinion* word. If the opinion word is POS-tagged as superlative or comparative the score is set to the maximum (+4) or minimum (-4) based on the polarity. During this process, the words that modify the polarity (e.g. “not”) and degree (e.g. “too”, “very”) are also considered for scoring the opinion word. The presence of these words can inverse or increase the sentiment score of the aspect respectively. The words POS-tagged as nouns are potential candidates to be the *feature* words. Apart from using the pre-defined feature look up file, these words are also tested to find any synonyms using the WordNet interface in NLTK.

Once the features and opinion words in a sentence are determined, a mapping is made to a feature and opinion word based on the distance between them. The aggregated opinion score for each feature is calculated for all the sentences in the review as mentioned above and the review document is updated with these values in the system’s database. The algorithm performing this process is outlined in Figure 2.

```

For each sentence in the review
For each word in the sentence
if the POS of the word is Adverb (RB) and is a TOO word
    Save the TOO word position
if the word is 'and'
    Continue the too rule ← There is no too rule mentioned in the
    algorithm!!
# opinion word
if the word is in the subjectivity lexicon or in the master list
    if the POS tag is a superlative adverb (RBS) or adjective (JJS)
        Set the superlative flag
    if the POS tag is a comparative adverb (RBR) or adjective (JJR)
        Set the comparative flag
if word in subjectivity lexicon
    Set the word score
else if word in positive master list
    Set the word score to +1
else if word in negative master list
    Set the word score to -1
if too exists and is adjacent
    if word is positive
        increase the score by 1
    if word is negative
        decrease the score by 1
if superlative flag Set
    if word is positive
        Set the score to +4
    if word is negative
        Set the score to -4
if comparative flag Set
    if word is positive
        Set the score to +3
    if word is negative
        Set the score to -3
if the opinion word is in negative context
    negate the sentiment of the score
Save the opinion_word_position and score
# Feature word
if the POS of the word is a Noun (NN)
    if the word is in feature list or a synonym of a feature
        Save the feature and position
Apply the opinion score to the potential feature in the sentence

```

Figure 2: Opinion score assignment algorithm

2.1.2 User profile generation

In order to generate personalized review rankings, we follow a neighborhood-based collaborative filtering approach. Given a user u and the set of businesses they have rated and/or reviewed B_u , each user is represented by a profile vector $U = \{p_1, \dots, p_k\}$ where p_i denotes the preference of a user for a business i and k is the total number of businesses in the system. We define p_i as follows:

$$p_i = \begin{cases} s_{ui} & \text{if } i \in B_u \\ 0 & \text{if } i \notin B_u \end{cases} \quad (1)$$

where s_{ui} represents the cumulative preference score of a user u for a business i , calculated using their overall rating or opinion on specific aspects of the business that are identified by their review for it. We introduce three alternative ways of calculating the preference score, namely using only the rating of the user, using the the specific review opinion scores, or weighing them by the overall preference/dislike of the user for each aspect, as shown in Equations 2, 3 and 4 respectively:

Rating-based preference score

$$s_{ui} = r_{ui} \quad (2)$$

where r_{ui} denotes the star rating of user u for business i .

² <http://www.nltk.org>

Business-based preference score

$$s_{ui} = \sum_{a \in R_{ui}} o_{ua} \quad (3)$$

where R_{ui} denotes the set of aspects included in the review of user u for business i and o_{ua} is the opinion score calculated for aspect a in this particular review.

Review-based preference score

$$s_{ui} = \sum_{a \in R_{ui}} w_{ua} \cdot o_{ua} \quad (4)$$

where w_{ua} denotes the overall preference/dislike of user u for aspect a , as expressed by their opinions on all reviews they've written. This can be calculated as the normalized sum of all the scores o_{ua} in all the reviews R_u .

Once the user profiles are created, we employ a user-based collaborative filtering technique to find similar users. In our implementation, we have used the Pearson correlation coefficient and the open source libraries provided by Apache Mahout³.

2.2 Online Recommendations

This step is used to rank and recommend reviews in real-time, as the user navigates the system and searches for new restaurants. When a given user searches for a specific restaurant, the recommendation engine computes the similarity of the current user with all the reviewers of the particular business and ranks and presents the related reviews in descending order of similarity. As a result, each user will be presented with a different set of reviews for the same business.

Moreover, the interface allows the end user to get the gist of the reviews without the need to read the entire review text. For each review, the overall star rating as well as the most important aspects of each review, are prominently shown. The aspects are intuitively marked as strong/weak positive/negative, by using colors and thumbs up/down images. We should stress that the same aspect might appear in more than one reviews and one review might contain more than one aspects.

3. SOCIAL NETWORK FEEDBACK

When available, information related to the user's social network can be incorporated in our model. There are two alternative ways this can be done, either at the last step of the process, or during the profile generation.

In the first case, the similarity between the user and their friends is calculated when the user searches for the restaurant. The friends' reviews for this restaurant are separately ranked and presented in a different list so that they are easily identifiable.

In the second case, the user preferences are weighed by the user's friends' opinion scores. To incorporate the social network feedback in the model, we extend Equation 1 as follows:

$$p_i = \begin{cases} w_{F_u i} \cdot s_{ui} & \text{if } i \in B_u \\ w_{F_u i} & \text{if } i \notin B_u \end{cases} \quad (5)$$

where F_u is the set of friends of user u and $w_{F_u i}$ can be defined as follows:

$$w_{F_u i} = \frac{\sum_{f \in F_u} s_{fi}}{|f|} \quad (6)$$

³ <https://mahout.apache.org/>

Equation 6 can be easily extended to incorporate the similarities between users.

Note that in this extension, we also address the cold-start problem since the user profile can be filled by social network feedback, even when the user has few, or none reviews/ratings in the system.

4. PROTOTYPE IMPLEMENTATION

We have already implemented a prototype based on our system design described in the previous sections. Our prototype implements the rating-based preference profile, assuming that the product aspects are predetermined. A screenshot of our prototype is shown in Figure 3. For demonstration purposes, each review is accompanied by some metrics showing the calculated polarity and subjectivity of the review [L12] as well as the similarity of each reviewer to the user. The end user may further filter the personalized list of reviews by filtering only those that come from his/her friends or by feature (e.g. location, food, etc.). More technical details on the implementation are included in [RSE14]⁴.

4.1 Experiments

With all user ratings unchanged from the dataset, following are the top 3 users whose reviews are recommended for the selected user (user id: _jjIaczpcVDPpHIH61334rQ, Name: Brian).

User ID	Name	Similarity
SC_bFYaDgDub7tDjKCvV_g	Michelle	2.7081284360304222E-14
qLCpuCWCyPb4G2vN-WZz-Q	Erica	1.615598444955481E-14
Ps1Db9zOatoF_76FZNO5CQ	Scott	1.6002397978549457E-14

Changing the rating for one of the restaurants that the user has rated from 4 to 1 as below, we see that there is a different set of users that are more similar.

```
{ "review_id" : "X5AJ05hPqLzXC9UaUIBOUQ", "stars": 4, "stars": 1, "type": "review" }
```

User ID	Name	Similarity
Ypz7hxOCnrg8Y8vxHJU-sQ	Michelle	5.466953550392E-15
U14IHoSXcztl-4PjyY0pug	Ed	5.081404003415E-15
HY9qpAamqLXFoa7xMwNntg	Braedy	4.1984444776220E-15

⁴ A screencast of the prototype is available at: <http://youtu.be/vMz5CobpIw4>

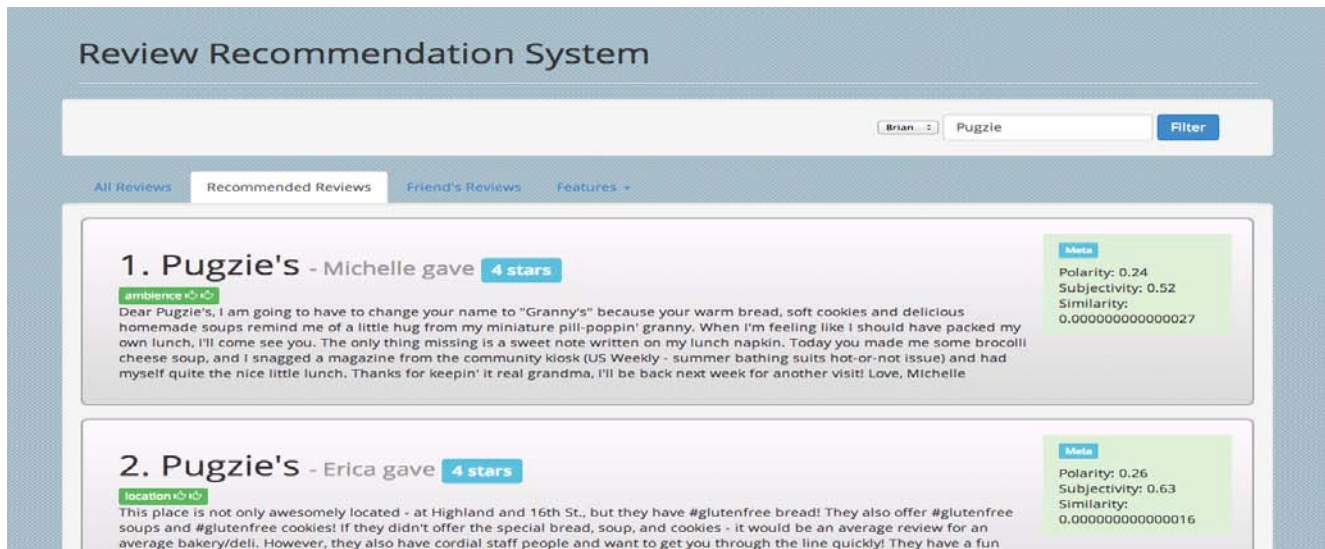


Figure 3: Client application - personalized recommended reviews

5. RELATED WORK

Many interesting works exist that focus on extracting the opinions from the customer reviews [L12]. The most recent ones employ features as an additional tool in representing the semantic orientation of a review [GZ+08, DLY08, EPS12]. This is an important line of work that provides very useful input in the creation of the rich user profiles of our system. The algorithm we introduce in this paper is along the same lines, however we should note that any similar approach could be easily integrated in our system.

None of the major web sites that include reviews as an indispensable part of their business provide aspect-oriented personalized review rankings. For instance, Amazon ranks reviews by helpfulness (number of “helpful” votes received) without providing any summary of the reviews, other than the overall star rating. Netflix’s rating system is also mainly based on the star ratings, whereas Google shopping allows users to create a list of pros and cons in addition to the review, but ranks them based on the review date. Finally, Yelp, whose dataset we are employing in this study, ranks reviews by helpfulness. It also provides an overall summary for each business in terms of several aspects (e.g. friendly for kids, romantic, etc.), as well as a short summary of the most common comments in the reviews. The last two businesses have some underlying social network that is not, however, utilized in re-ranking or personalizing the reviews.

Similarly, not much work has been done in the research community. In [GI07], the authors discuss the problem of using helpfulness as a way to rank results and perform an econometrics analysis on identifying correlations between the subjectivity of reviews and the increase in product sales. They conclude that for experience goods, users prefer a brief description of the “objective” elements of the item and then expect to see a subjective positioning, described by aspects not captured by the product description. Our work not only addresses these findings, but also proposes ways of personalizing the rankings for each user, taking into consideration their social network as well. To

the best of our knowledge, this is the first effort in addressing this problem.

6. CONCLUSIONS

The amount of online reviews for products and services has grown to such extent that often makes it impossible to read all of them. In this work we propose a system that personalizes the order in which the reviews are shown and provides an intuitive interface that allows the users to see the important aspects of each review in a glimpse. As part of our future work we plan to integrate further these two types of recommendations and enhance them by introducing trust-based and reputation metrics.

7. REFERENCES

- [DLY08] X. Ding, B. Liu, P. S. Yu, *A holistic lexicon-based approach to opinion mining*, in Proc. of WSDM '08
- [EPS12] M. Eirinaki, S. Pisal, J. Singh, *Feature-based Opinion Mining and Ranking*, J. of Computer and System Sciences (JCSS), 78(4), pp.1175-1184, July 2012
- [GI07] A. Ghose, P. Ipeirotis, *Designing Novel Review Ranking Systems: Predicting the Usefulness and Impact of Reviews*, in Proc. of ICEC '07
- [GZ+08] H. Guo, H. Zhu, Z. Guo, X. Zhang, Z. Su, *Address standardization with latent semantic association*, in Proc. of ACM KDD'09
- [L12] Bing Liu, *Sentiment Analysis and Opinion Mining*, Morgan & Claypool Publishers, May 2012
- [RSE14] Syeda Roohi, Vaishak Suresh, Magdalini Eirinaki, *Aspect based Opinion Mining and Recommendation System for Restaurant Reviews*, demo paper, in Proc. of ACM RecSys 2014
- [WWH05] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann, *Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis*. Proc. of HLT-EMNLP-2005.