Aspect-Based Opinion Mining and Recommendation System for Restaurant Reviews

Vaishak Suresh  
Computer Engineering Department  
San Jose State University  
San Jose, CA, 95192

Syeda Roohi  
Computer Engineering Department  
San Jose State University  
San Jose, CA, 95192

Magdalini Eirinaki  
Computer Engineering Department  
San Jose State University  
San Jose, CA, 95192

ABSTRACT
The success of a product/service in e-commerce largely depends on the user reviews. A product/service that has a higher average review or rating usually gets picked against a similar product/service with less favorable reviews. 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. This demonstration presents a system that extracts aspect-specific ratings from reviews and also recommends reviews to users based on their and other users’ rating patterns.

Categories and Subject Descriptors
H.3.5 [Information Storage and Retrieval]: Online information Services

Keywords
Recommendation Engine; feature ranking; sentiment analysis;

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/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). However, such a generic ranking requires from users to read or at least scan the tens or hundreds of reviews for one product/service.

Moreover, different people value different qualities of the same product/service. For example, when searching for a good Italian restaurant, one user might value the 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 propose a system that automatically generates personalized review recommendations using two different approaches. Firstly, drawing inspiration from traditional collaborative filtering systems, the system generates user rating profiles and tailors the list of reviews to the preferences of each user. Secondly, we employ aspect-based opinion mining to identify the important features highlighted in each review. As opposed to traditional sentiment analysis, which provides an overall picture of whether a review is positive or negative, aspect-based opinion mining provides a fine-grain analysis of both sentiment and strength of the review [1]. Given a predetermined set of features for a particular domain, results from aspect-based opinion mining can be used to rank/sort the reviews based on the aspect the users are most interested in. The proposed system displays a personalized list of reviews for the same product/service tailored to the preferences of each user. We demonstrate our system using real-life reviews from Yelp.

2. SYSTEM OVERVIEW
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. The prototype uses user review data from Yelp1. The dataset contains user information, business information and user reviews in the form of JSON objects. These objects are stored on MongoDB, a NoSQL database. A brief overview of the system is provided in what follows. For more details please refer to [2].

2.1 Offline Processing
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 [3] 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

1 http://www.yelp.com/dataset_challenge/
from opinion lexicon [1] we created a list of negation words (not, no, nothing etc.) which inverse the sentiment, and positive words (too, very, so, etc.), which increase the intensity of the sentiment. Each review is POS tagged using the default tagger from NLTK\(^2\) python package, which uses the Treebank corpus for training the model. These three inputs are subsequently fed to our algorithm that generates as output an opinion score for each feature identified in a review [2]. In this work we assume that the product aspects are predetermined.

2.1.2 User preference generation
The front-end of the application acts as a REST client and sends request to the RESTful server written using Jersey\(^3\) to recommend reviews. We employ a user-based collaborative filtering technique where user profiles consist of all their star ratings in past reviews. The RESTful service uses MongoDBDataModel API from Mahout to build a model based on the ratings of all the users. User similarity is calculated using the Pearson correlation coefficient.

2.2 Online Recommendations
This step is used to rank and recommend reviews. The client is a Django-based Python Web application that consumes the recommendations in the form of a RESTful response from the engine. 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. 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\(^4\). A sample result set is shown in Figure 2. For demonstration purposes, each review is accompanied by some metrics showing the calculated polarity and subjectivity of the review [1] as well as the similarity of each reviewer to the user (similarity is very low as the utility matrix is very sparse). The end user may further filter the personalized list of reviews by filtering only those that come from his/her friends (in a virtual social network setting), or by feature (e.g. location, food, etc.).

3. DEMONSTRATION SCENARIO
We will demonstrate the prototype by employing different users with different ratings and show how their review recommendations differ for the same businesses. The audience will be also shown some back-end details on how the user similarity is calculated. A screencast of the prototype is available at: \url{http://youtu.be/vMzScobplw4}.

4. 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.

5. REFERENCES