Instructions for ACL-2015 Proceedings

Shuwei Zhang
shuweiz@usc.edu

Maiqi Tang
maiqitan@usc.edu

Second Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

1 Project Domain & Goals

The of this model is to give old users of Yelp restaurant recommendations based on their previous reviews posted on Yelp.

Our group picked task1-c as our project topic. The non-NLP domain algorithm we use is item-based collaborative filtering in data mining, widely used in the item recommendation system. Meanwhile, we want to enhance the user matching , which is a significant step in item-based collaborative filtering algorithms with several NLP algorithms.

The data set we will use is the data set of user reviews on Yelp. We want to use several methods to do data cleaning and preprocessing. After that, we use these combined algorithms mentioned before to calculate the user's review's similarity instead of simply using the user's rating to do user matching. For example, if we found several users' reviews are similar, then these users are matched. We would use these users' ratings in the group and use a test data set to calculate accuracy. Finally, we will evaluate the performance of each method in each step which would lead to the best result. And using the best combination to train our model, giving a better outcome for users.

The problem we solved is the recommendation by user's rating cannot fit users' interests well. For example, if we use ratings to find a restaurant that chicken is juicy and its overall rating is exceptionally high, the system recommends it to the user. However, this user prefers chicken, which is overcooked. And we could not find a matched restaurant by rating or simple categories filter. Then we use the NLP algorithm to analyze reviews' similarity. If the recommendation system could find a restaurant's reviews that are similar to the user's old review, this restaurant can better fit the user's interest.

2 Related Work

3 Data

In order to build a text-based recommendation system, we need to find a data set that records reviews and rating toward restaurants. Fortunately, Yelp has established a well-organized streamed database [1] that provides reviews and ratings from customers, and the information about the restaurants those reviews are about. In this database, all the data is stored in JSON format with consistent schemas. The data set that we will mainly work on is the review data set, which provides the reviews from users to business; and the business data set, which provides the information of business listed on Yelp. In the reviews data set, we will mainly work with following attributes: business_id, which is unique for each business listed on Yelp; text, which the review left to this restaurant; and stars, which is the rating left by the reviewer. For the business data set, we will use the business_id from the reviews data set to match to the reviewed business. Then we can use the information from the business data set to tag and group up business.

The review data set had over 8 million instances, and it is hard to load all the data at once due the limitation of our computation resource. Therefore, we will utilize the business data set and apply some data mining algorithms to split the data set to reduce the size and avoid the biased data set. We will use some data mining tool, like Pyspark, to load and process the data parallelly Afterward, we will shuffle and split the data set into a 4: 1 train and test partition. For the text part, we

[1] Yelp Data set documentation url: https://www.yelp.com/dataset/documentation/main

will first do the data clean to remove the unnecessary characters and do the contractions for the text. Meanwhile, the vocabularies used in the review are not alway correct. Therefore, we will use the spell corrector library in python to correct the spelling and reduce the vocabulary size. Also, another challenge is raised by the length of each review, which varies from 1 word to a long paragraph. Therefore, we will find a threshold to truncate the long review to increase the efficiency of feature extraction.

4 Technical Challenge

1. Contextual word and phrases homonyms. When we compare two sentences with semantic similarity, it is possible to pair two different meanings of sentences which contain the same words into one cluster. While NLP language modeling can learn different meanings and definitions, differentiating between them in context can still present problems.

2. Irony and sarcasm. Language models usually interpret words or phrases, it can be positive or negative, however, in fact, the words or phrases actually connote the opposite.

3. Error in text. Misspelled or misused words can also cause problems for sentence text analysis. Although we can use grammar correction to alleviate problems, it cannot interpret the writer's intention.

4. Colloquialisms and slang. Informal phrases, expression, idioms, and culture-specific lingo also present problems for language modeling. Cultural slang is constantly morphing and expanding, so new phrases and words pop up every day. It is hard for a formal language model to distinguish new words and phrases.

5. Large data sets cause scale problems. For some reviews, it is too short to perform language analysis. One option to solve the problems is padding the shorter reviews or treat it as the outlier and remove it.

6. New users can also present problems. Our model will find similar reviews and then recommend restaurants to users who share similar tastes. But this model cannot be applied to the new users in the platform, because they may not have comments yet. We can tackle this problem by implementing the content based analysis, which is a data mining technical.

Acknowledgments

The acknowledgments should go immediately before the references. Do not number the acknowledgments section. Do not include this section when submitting your paper for review.

References

Alfred V. Aho and Jeffrey D. Ullman. 1972. The Theory of Parsing, Translation and Compiling, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

American Psychological Association. 1983. Publications Manual. American Psychological Association, Washington, DC.

Association for Computing Machinery. 1983. Computing Reviews, 24(11):503–512.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. Journal of the Association for Computing Machinery, 28(1):114–133.

Dan Gusfield. 1997. Algorithms on Strings, Trees and Sequences. Cambridge University Press, Cambridge, UK.