Unsupervised Lexicon-Based Sentiment Topic Model

Yan Yu Chen, Qisi Deng, Hengxin Li and Bochao Zhang

Abstract— **This report presents the Unsupervised Lexicon-Based Sentiment Topic Model (ULSTM) as a sentiment analysis model for reviews on the popular crowd-sourced review forum Yelp. The model applies an unsupervised learning since the supervised method has many constraints. Furthermore, instead of employing an existing sentiment lexicon, we developed a sentiment dictionary using the linguistic corpus WordNet; the self-defined lexicon allows more targeted scoring towards the evaluated dataset. Finally, the ULSTM adopts the Latent Dirichlet Allocation model to find the most mentioned topics in reviews for individual businesses.**

I. INTRODUCTION

The field of sentiment identification of a given text is of vital importance nowadays, as opinions expressed by others can have significant influence on our daily decision-making process. From a business perspective, analyzing sentiments of customer reviews can help them better cater customers needs. With the emergence of Microblogging platforms, researchers in Natural Language Processing (NLP) have increased interest in the automatic detection of sentiment out of mass texts.

Supervised sentiment analysis, or opinion mining, has achieved state-of-the-art performance using variations of Long Short-Term Memory Model. However, it came to our attention that such approach has certain constraints. Firstly, supervised learning requires labeled training data, which is not readily available in the field of sentiment analysis. Secondly, peoples quantifications of the same sentiment are highly subjective. For instance, an ok sushi dish might receive a four-star from one but a three-star from the next. Furthermore, it is not feasible to find a well-structured dataset with standardized sentiments. Hence, an unsupervised analysis is superior in terms of sentiment detection since it requires neither a true label attached to the text nor a universal scoring standard.

Inspired by Taboada et al. (2011) and Hu and Liu (2004), we present the Unsupervised Lexicon-Based Sentiment Topic Model (ULSTM) using a self-established sentiment lexicon. The current application of the model mainly concerns with reviews from the Yelp Open Dataset, which contains almost 6-million customer reviews. The model assigns each review a corresponding sentiment score based on its semantic meaning and syntactic structure. Then, it applies Latent Dirichlet Allocation (Blei et al., 2003) (LDA) to selected businesses, aiming to extract global topics as highlights and/or opportunities for improvement of the restaurants. Using these Yelp reviews as a starting point, we hope to extend the model to all public tweets/texts on different social platforms.

II. SENTIMENT LEXICON

Prior to constructing our own lexicon, we first adopted the existing SentiWordNet 3.0, which is a popular lexical source with a sentiment score assigned to each word. In order to utilize it, the syntactic function and specific meaning of a word must be known beforehand. Even though Stanford CoreNLP POS (Part-of-speech)-tagger has shown promising performance to identify syntactic function, we still lacked effective tools to detect the exact meaning of a word within a sentence. Thus, the sentiment score attached to each word is not always accurate. For example, the word researcher receives a sentiment score of 1 (the highest score possible in SentiWordNet) which does not make sense intuitively. Therefore, we developed a sentiment dictionary, YelpSentiWord lexicon, which is more pertinent to a restaurant setting.

To create YelpSentiWord lexicon, we explicitly defined a list of **seed words** for which sentiment polarities are easy to distinguish. Seed words were filtered from the most common adjectives, verbs and adverbs appeared in Yelp reviews, which are mainly descriptive vocabularies for food taste, restaurant vibes, and food prices. They were further split into three subclasses, namely Positive, Negative, and Neutral. We also introduced two more variables: layer as a position variable, which will be used as we grow the dictionary; parent denoting the parent word from which it was generated. Hence, each seed word has been annotated with its semantic orientation and layer number.

Once seed words have been defined, we built individual trees to grow synonyms upon each seed, assuming that synonyms stemmed from a seed word would share the same semantic orientation as the seed word. The layer variable now comes into play to indicate at which ”layer” the corresponding word is found. In the case of positive and negative classes, the semantic orientation of a word is always prioritized to the one for which its root occurs at a lower layer. We adopted such prioritization as words grew in higher layer are more susceptible in losing the original sentiment of the seed word. Also, if a word occurs in both positive (or negative) and neutral classes at the same ”layer”, then semantic orientation is always assigned to non-neutral one. Relatively speaking, there are fewer neutral Senti-words (word with a sentiment) comparing to either positive or negative. Since our dictionary consists of a comprehensive list of neutral words, it is likely that some synonyms of these neutral words crossed over to the positive branch. The following is an example of the mentioned ”tree” with seed word being sketchy:

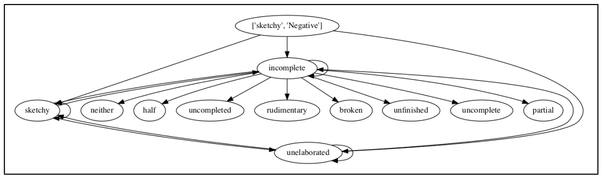

Fig. 1. word-tree from YelpSentiWord lexicon with seed word ”sketchy”

III. PREPROCESSING

Raw datasets often contain a deluge of irrelevant information and would prolong the processing time. As the initial step, we converted reviews into **lowercases** and divided each review into **sentences**. Considering Yelps multi-language nature, we only selected **English** reviews because of the ample amount of NLP programming packages based in English.

For the purpose of performing LDA in the later stage of the project, we first **grouped reviews by business names**. Under the assumption that core sentiments of a multi-sentence review are often expressed at the beginning or the end, we investigate the sentiments embedded in first and last few sentences of a review. Doing so not only mitigates impacts made by noise but also gives a pool of more focused topics as discussed in the LDA section.

The last step of preprocessing is **lemmatization**. Instead of lemmatizing all the words, we only perform lemmatization to words tagged as verbs and plural nouns in their POS-tagging. The reason is intuitive: our dictionary only contains nouns in singular form and verbs in based form, but adverbs and adjectives are invariable.

IV. SENTIMENT CALCULATOR

The calculation of sentiment for a review begins with two general assumptions: 1) sentence-level sentiment is accumulated by the amount of Senti-words; 2) semantic orientation can be expressed as a numerical value, which in our case, 1 for Positive, -1 for Negative, and 0 for Neutral. In addition to sentiment-lexicon, we also defined supplementary dictionaries, intensifier, negation, and conjunction words, to better capture the sentiment under certain syntactic relationships. We will now explain how the score for a Senti-word is defined.

A. NEIGHBOURHOOD

To detect sentiment more accurately, we proposed the idea of **neighbourhood** for each Senti-word such that only words in its neighbourhood will affect the sentiment score of it. We define words in the neighbourhood as words in the independent clause of the Senti-word and within a distance of 7 words from the Senti-word. We will elaborate on the use of neighbourhood for intensification and negation below.

TABLE I

EXAMPLE FOR NEIGHBOURHOOD

Sentence	Word	Neighbourhood
I went to school and bought a shawarma.	school	I went to

B. INTENSIFIER

Intensifier plays a significant role in sentiment evaluation, as it can increase, weaken or reverse a sentiments intensity. The use of intensifier is based on the assumption that it can only embellish Senti-words within its predefined neighbourhood. To account for intensification, we have adopted an existing intensifier dictionary (Taboada et al, 2011) together with some of our self-defined ones, where each intensifier has a corresponding scalar and will have a multiplication effect on the Senti-words original score by (1 + scalar).

TABLE II

EXAMPLE FOR INTENSIFIER

Sentence	Word	Original score	Intensifier	Scalar	Final score
The food is very good.	good	1	very	0.5	1.5

C. NEGATOR

The purpose of using negation is to simply reverse the semantic orientation of the ”Senti-word” next to a **negator**. For example, good is a word with a positive sentiment, but not good gives a negative sentiment. Here, we made an assumption that a negator will change a neutral word to a negative word. We also assumed that negators only affects a Senti-word if they are in the neighbourhood of that Senti-word.

TABLE III

EXAMPLES FOR NEGATOR

Sentence	Senti-word	Original score	Negator	Final score
Nobody likes this restaurant.	likes	1	nobody	-1
I cant say the dish was delicious.	delicious	1	can’t	-1

D. EMOTICON

Emoticons have been widely used on social platforms as a pictorial representation of sentiments using charactersusually punctuation marks, numbers, and letters. We incorporated them into our model after observing significant usage in the Yelp dataset. Each emoticon is treated as a special type of Senti-word, which preserves its sentiment score when intensifiers and negators are present. For ULSTM, we used the emoticon-scoring dictionary from the AFINN library in Python.

TABLE IV

EXAMPLES FOR EMOTICONS

Emoticon	Score
:(	-1
<3	1

E. WEIGHTED SCORE FOR REVIEWS WITH MULTIPLE SENTENCES

Since most reviews are composed of multiple sentences, we used a weighted average approach to compute the final sentiment score of a review. The initial weights are uniformly defined to be the reciprocal of the number of sentences in a review. After further analysis, we discovered that many users often mentioned other businesses for comparison purpose in certain sentences. Since the sentiment in these sentences does not describe the reviewed business, we decided to penalize them by a lower weight. To do so, we utilized the ***POS-tagger*** and ***to_truecase*** functions in the **StanfordNLP** library to identify sentences with proper nouns. If the found proper nouns are different than the commented business name, we would down weight the sentence by half. Additionally, since different sentences within a review might express opposite sentiments, we defined a dominant sentiment of a review based on the amount of sentences carrying same sentiment. To be more precise, If a review carries more sentences of one sentiment than the others, an up-weight factor will be applied to reward those sentences to magnify the overall sentiment.

V. LATENT DIRICHLET ALLOCATION (LDA)

With an accurate sentiment score for each review, we are ready to unveil the attractions and areas of improvement for different restaurants. We categorized the sentiment scores into three sentiment levels (positive, neutral, negative) according to the magnitude of the scores. Considering the cost of labeling reviews with multiple topics, we decided to apply LDA, a classic unsupervised topic model, to classify reviews of each sentiment level by their latent topics. With the help from LDA, we extracted latent topics of each review and obtained the most frequently mentioned topics. These topics, each representing one aspect of a dining experience (e.g.: taste, service, price and etc.), were assumed to reflect areas that a restaurant needs to maintain the quality of or make immediate improvements on.

Similar to sentiment analysis, LDA begins with data preprocessing. The procedure is standard: tokenization, stopwords removal, lemmatization, stemming, and dictionary construction. Yet, considering the nature of our dataset, we decided to filter out tokens that appeared in less than 500 reviews or more than 70% of the documents. Such filtering was based on the hypothesis that rare words in Yelp dataset are likely to be menu items, whereas words occurring too frequently may not provide much insight to restaurant owners. Later, we constructed a bag of words (BoW) from the preprocessed reviews and created a TF-IDF (term frequency-inverse document frequency) model to train the LDA model. The TF-IDF model adjusted the weight of each word in BoW to reflect that some words appear more frequently in general.

As the last step of topic modelling, we sorted the latent topics of a review by topic scores in descending order and explored ten topics with the highest scores for each restaurant. We then grouped these topics into either attractions or areas of improvements based on the sentiment orientation of the review. Since the classification of latent topics is unsupervised, we could not obtain the accuracy level by comparing our prediction with a true label. However, we realized that most of the topics found per reviews made sense intuitively when compared with types of restaurant. For instance, [example later]

VI. RESULTS

After running the sentiment calculator on the Yelp dataset, we were able to predict the sentiment of each review. Here are som examples:

TABLE V
EXAMPLES OF ULSTM SCORES

Review	Sentiment score	Label
The service is terrible and while walking to the bathroom I got a glimpse off the kitchen it was so dirty and disgusting how the health department has not shut them down yet I have no idea I know for a fact I will never there again	-4	Negative
Excellent food+ excellent service, everyone is very nice and helpful especially our nice and friendly host Constantine!. We will definitely come back again next time we are in Vegas.	5.825	Positive
Makis are good but globally meals are overpriced & hyped. Go off-strip if you want to have Japanese for half the price!	0.5	Neutral

Although the sentiment calculator classifies the reviews in an unsupervised manner, we were still able to calculate the accuracy of the model using the Yelp dataset. When reviewing a restaurant, the Yelp users are asked to rate the businesses with a rating of one to five stars. Therefore, we were able to evaluate the accuracy of the model by assuming the rating to be the true labels of the reviews (i.e. one- or two-star means Negative, three-star is Neutral, four- or five-star represents Positive). By comparing the predicted labels and true labels, we got the following results: [result]

TABLE VI
EXAMPLES OF ULSTM SCORES

Review	Sentiment score	Label
The service is terrible and while walking to the bathroom I got a glimpse off the kitchen it was so dirty and disgusting how the health department has not shut them down yet I have no idea I know for a fact I will never there again	-4	Negative
Excellent food+ excellent service, everyone is very nice and helpful especially our nice and friendly host Constantine!. We will definitely come back again next time we are in Vegas.	5.825	Positive
Makis are good but globally meals are overpriced & hyped. Go off-strip if you want to have Japanese for half the price!	0.5	Neutral

Furthermore, we were able to generate the latent topics for selected restaurants in the filtered dataset. To visualize, we chose the restaurant ”Wicked Spoon” for which we produced the word cloud from the TF-IDF model.

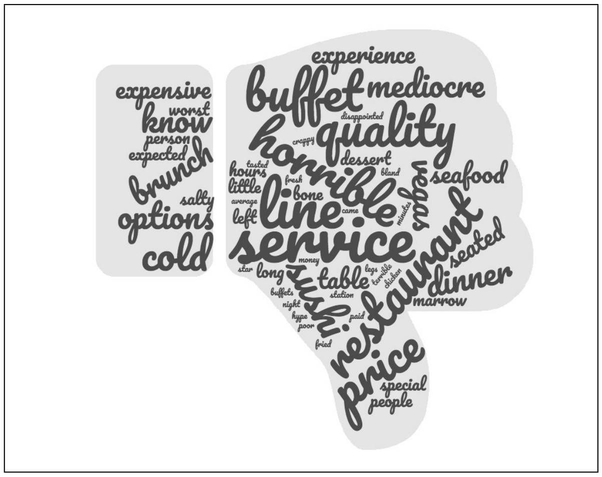

Fig. 2. Word cloud for topics generated from reviews of restaurant ”Wicked Spoon”

Then, as we mentioned ealier, most of the topics found per reviews made sense intuitively when compared with types of restaurant. For instance, running LDA with 100 latent topics on reviews with positive sentiment (sentiment scores ¿ 0.5) on a restaurant named SkinnyFATS allowed us to identify the following strong points:

TABLE VII

Topic distribution from reviews of restaurant ”SkinnyFATS”

Topic	food	burger	service	vegas	fries	amazing	chicken
Dist.	0.09	0.08	0.08	0.07	0.06	0.03	0.02

In words, from the latent topics provided by LDA, we inferred that SkinnyFATS, located in Las Vegas, serves delicious burger, fries, chiken, and steak along with amazing service. These are accurate in terms of the type of food served in the restaurant and can inform restaurant owners of how to retain existing customers.

Now if we run LDA on a restaurant that has a lot of negative reviews, for instance, Wicked Spoon in Vegas, we can obtain the following topic distribution:

TABLE VIII

Topic distribution from reviews of restaurant ”Wicked Spoon”

Topic	horrible	buffet	service	price	salty	vegas	wait
Dist.	0.03	0.03	0.03	0.03	0.02	0.02	0.02

Therefore, With the help from LDA, we can reasonably suggest the owner of Wicked Spoon to perhaps adjust buffet price, increase meat selection, and raise service quality.

VII. Conclusion and Future Improvements

To conclude, in order to overcome some constraints of supervised learning for sentiment analysis, we developed an unsupervised model ULSTM. This model captures the sentiment of text by rating it with a self-developed sentiment dictionary. By doing so, we were able to increase drastically the accuracy of the prediction when compared with the results predicted using existing sentiment dictionaries like SentiWordNet. Additionally, the model employs LDA topic model to extract global topics as highlights to maintain and/or opportunities for improvement.

Even though the results report a great (hopefully) accuracy with our model, there are still a few possible improvements. As mentioned in the preprocessing section, we converted each word into lowercase to avoid multiple representations for the same word. When the project progresses, we recognized that few people are accustomed to using capitalization (eg: I LOVE this restaurant so much!) to express a strong emotion. Conversion to lowercase in the preprocessing thus ignores it.

Moreover, as we established YelpSentiWord lexicon based upon their semantic classifications, the vocabularies were only distinguished by indicators for each three class, without accounting for the fact that the amount of sentiment each word express is very different. For example, love expresses a more intense sentiment than like.

Finally, for the purpose of our project, we only considered reviews related to food and restaurants and established a dictionary accordingly. We are confident at the extensibility of our model and look forward to discovering other applications of it in future works.

REFERENCES

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. Journal of Machine Learning Research 3, pages 9931022.

[2] M. Hu, B. Liu, Mining and summarizing customer reviews, (Periodical style), ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), pp. 168177.

[3] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, Lexicon-Based Methods for Sentiment Analysis (Periodical style), Computational Linguistics, Volume 37 , No. 2, May 2011, pp.267-307.