Automated Aspect Extraction and Aspect Oriented Sentiment Analysis on Hotel Review Datasets

Varun Agarwal
Information Technology
Dwarkadas J. Sanghvi College Of Engineering
Mumbai, India
varun.agrwl96@gmail.com

Pratik Aher
Information Technology
Dwarkadas J. Sanghvi College Of Engineering
Mumbai, India
pratikaher88@gmail.com

Vinaya Sawant
Information Technology
Dwarkadas J. Sanghvi College Of Engineering
Mumbai, India
vinaya.sawant@djsce.ac.in

Abstract— **Humans always try to make the best of their money be it food, clothes or hotel. Reviews provided by people play a crucial role in deciding if the hotel is worth staying. A hotel has many different aspects to be judged on while reviewing like food, rooms, service, etc. Due to the huge volume of available data on reviews it becomes a cumbersome task to summarize the gist of reviews. Hence, a need for an automated rating system arises. We have implemented an automated system that extracts the aspect terms from reviews and performs aspect-oriented sentiment analysis to summarize the review in a rating system on the basis of prominence of aspect terms.**

Keywords— Word2vec; Stanford Dependency Parser; Sentiment Analysis; Reviews; Clustering.

I. INTRODUCTION

People according to their experience provide reviews to hotels online. Ample amount of reviews regarding hotels are provided by people all around the world. Reviews provided are usually raw text with no specific structure to it. Reading those big reviews is a cumbersome task. Problem in automated rating systems is predefined aspects which makes it very case specific. A rating system that automates the selection of aspects and does sentiment analysis based on these aspects is the goal of the paper. Consider a review "Great experience. Hotel rooms were exquisite. Staff was very friendly and welcoming. Food was a little disappointing." As a customer trying to explore how good a hotel is given such unstructured reviews is quite hectic. Thus, extracting the core aspects of a review on which it should be judged and assigning a rating to it is the need of the hour. For the given review, aspects would be – "Room"," Staff", "Food". Most review systems predefine such aspects. Also most reviews need not use same words as the aspect names. For example, "service" also denotes "staff", "beds" would indicate "room". So for a review defining "service is good", system might not consider it in the "staff" aspect, ideally what the case should be. Hence proper clustering of aspects based on its synonym vector and a sentiment analysis on it would provide better results than just predefining aspects.

II. RELATED WORK

Referring to our knowledge bubble and discovery, we found that creation of aspects depending on the given dataset is always a difficult task and requires a lot of work. Classifying opinionated texts at document level or at sentence level is useful in many applications; but it does not provide necessary details needed for many applications. A positive opinionated document about a particular entity does not mean that the author has complete positive opinion about all the features of the entity. Likewise, a negative opinionated document does not mean that the author does not at all like all the features of the entity. In a typical opinionated text the authors writes both positive and negative opinions with respect to entities and their attributes. Aspect Based Sentiment Analysis aims to identify the aspects of entities being used in expressing sentiments. It is also used to determine the sentiment that expressed by author towards each aspect of the entity[2]. Sentiment analysis, also called opinion mining, is the field of study that analyzes people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes. It represents a large problem space[6]. Sentiment Analysis is done on three levels: Document level, Sentence level & Entity and Aspect level[6].

Aspect, Polarity word (A-P) collocation extraction and aspect polarity recognition can be considered as the basic tasks of the aspect-based sentiment analysis[3]. The A-P collocation extraction attempts to extract the collocation, while the aspect polarity recognition aims to identify the "negative" polarity tag through the polarity word that modifies the aspect [3]. The syntactic relation "SBV" (Subject and Verb) between the aspect and the polarity word2 can be used as an important evidence to extract the A-P collocation[3]. In a previous approach, they have focused on part-of-speech information where the focus was on nouns, pronouns, verbs, and adjectives. Based on the observation, nouns and pronouns are good candidates for aspect terms[1]. In Sentiment Analysis [4], an opinion is defined by making reference to a quintuple (o; a; so; h; t) that consists of:

978-1-5386-5257-2/18/$31.00 ©2018 IEEE

1) Object ‘o’, which is the opinion target. It can be a product, service, topic, issue, person, organization, or an event.

2) Aspect ‘a’, which is the targeted attribute of the object ‘o’.

3) Sentiment orientation ‘so’ that indicates whether an opinion is positive, negative or neutral.

4) Opinion holder ‘h’, which is the person or organization that expresses an opinion.

Authors in [5] have mentioned use of two probabilistic models: Supervised and Unsupervised. Supervised techniques for aspect based sentiment analysis have obviously proven much better than unsupervised ones in most domains but are challenged by non availability of labelled data in certain domains. Automated aspect creation is a need in this automated world and few solutions for the same have been put forth. Assumption of the fact that dataset will have the aspect words with highest frequency is just not accurate enough. So we moved a step ahead of clustering all the similar words in the high frequency word count by using Word2Vec to get a vector of similar words and then cluster them using K-means. Also after retrieval of aspects, sentiment analysis is done only for the aspect words without consideration of other words that are similar to the aspect words. So we make use of Stanford Dependency Parser to extract Amods, Advmods and COP to extract relevant words and their sentiment. Then these amods, advmods and cop’s are classified into the respective aspects by using Wordnet similarity. This takes into consideration words similar to the aspects instead of neglecting them and hence gives a better accuracy in reviews. Then sentiment analysis is done on those amods for each aspect and average is taken for a final review score.

III. PROPOSED METHOD

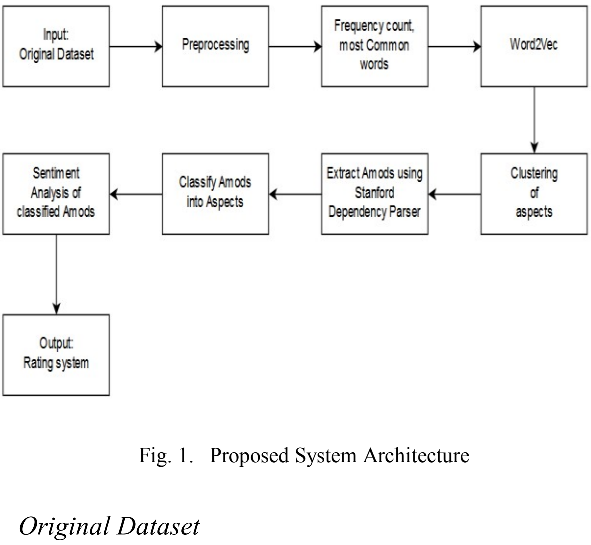

Input:
Original Dataset
Preprocessing
Frequency count, most Common words
Word2Vec
Clustering of aspects
Extract Amods using Stanford Dependency Parser
Classify Amods into Aspects
Sentiment Analysis of classified Amods
Output:
Rating system

Fig. 1. Proposed System Architecture

A. Original Dataset

First task was to collect a corpus of hotel reviews. For this, we took a sample corpus from Kaggle. Our dataset included many attributes of a hotel review like timestamp, address, country, review text, username, overall rating, etc. But we only needed the text review and no other attributes for our paper.

B. Preprocessing

After extracting the required text review attribute from the dataset, next was to preprocess the extracted text. Our aim is to find the aspects related to a hotel automatically and not by predefining it. Thus, our system needs to automatically find the aspects like room, staff, food, etc from the review corpus. Extracted reviews were of different languages, so for simplicity we only considered reviews in English. All other language reviews were discarded. In this fast paced digital world, hardly any attention is paid to spell out the words right. This is so common among people but discarding of such words such as “god” which was supposed to mean “good” and “bad” which was to be spelled as “bad” cannot be discarded as they change the context of a review and needs to be considered. This auto-correction was done using a library “autocorrect” and importing spell from it. All reviews are long written texts not providing any or significant value for finding aspects. There are certain stopwords like “a, an, the, is, or, me, I, etc” which are of no use. Also punctuation marks do not prove of any help in finding the aspects pertaining to the hotel. Also removal of HTML tags from the reviews was needed. We also know that any word having length less than 3 cannot be an aspect from manual evaluation. The entire corpus was first tokenized into words and all the stopwords and irrelevant content was discarded with the help of NLTK (Natural Language Toolkit).

C. Frequency count, most common words

Now, after having a clean pre-processed text from the corpus dataset, next task was to find aspects. We did a frequency count of all the words in the pre-processed text. This was done by taking a pragmatic approach that all the reviews will specify the aspects like food, staff, etc whether they are good or bad which in turn will increase their frequency of appearance. This frequency counting of words from preprocessed text was done by using counter from collections. Now we get a count of most common words.

D. Word2Vec

Now some reviews might specify “room” while some might use “bedroom”. Similarly some reviews may have “food” while others may have “breakfast” or “lunch” or “dinner”. These words must fall into a same aspect as they convey the same meaning. Thus, a need to cluster words that convey similar meaning is required. This leads us to the next step of clustering the high frequency common words into a single aspect like (food, breakfast, lunch, dinner) as food, (room, bedroom) as room, (staff, service, manager) as staff, etc. First Word2Vec was used to get a vector of words that are synonyms to the given word. For eg: Word2Vec of breakfast may be [food, breakfast, lunch, dinner,….]. After receiving the vector of synonyms, clustering of similar words was done using K-means algorithm. After clustering these high frequency most common words, we finally get our aspects list (food, room, staff, etc). Now these act as our main aspects on which rating is to be found. Now, the next task was to extract terms related to these aspect list and to find their polarity whether it conveys a positive, negative or a neutral stand. First let's extract the related terms from the corpus.

This was done using Stanford Dependency Parser. This parser would return Amod (Adjective mod), Advmod (adverb mod) and COP (copula). These are used to extract words which are adjectives, adverbs and find related words that indicate whether the extracted adjective, adverb are positive, negative or neutral. This gives a list of tuples like amod(breakfast, amazing), etc. Now from the extracted tuple we know that food served was amazing at the hotel.

The only task now left is to associate the extracted words to their corresponding aspect and depending on the polarity of the extracted word, we need to assign a rating to the corresponding aspect. So firstly, we need to associate the word to the corresponding aspect, which here in our case is to associate the word "breakfast" from amod (breakfast, amazing) to the aspect "food" from the aspect list. This association is done using Wordnet Similarity. Since "breakfast" is highly similar to aspect "food" from the aspect list it is associated to food aspect. After association, Sentiment Analysis is done on the extracted word and the obtained score is assigned to the aspect. This is done for all the associated words i.e. a sentiment score is obtained and assigned to the corresponding aspect. After all words are associated and scores are assigned to the corresponding aspect, average of all the scores is done to obtain a final value which is the rating for a given aspect. This provides a much better rating and also is aspect specific as most hotels are excellent in some aspect but may be poor in other aspect but current overall rating does no justice in such cases.

E. *Rating*

Our system would provide aspect specific ratings which provides better and just understanding of a hotel and also eliminates the need of predefining the aspects and automating the process.

IV. RESULTS

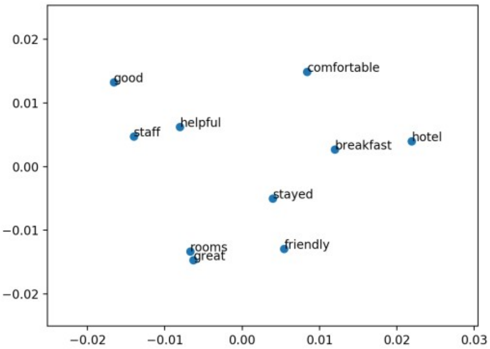
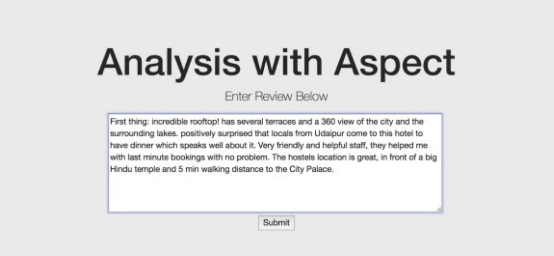


Fig 2. Sample of clustering of different terms into aspects

The most common words from frequency count are clustered. This clustering is done after preprocessing. The output after clustering is bifurcated into distinct lists. On performing the analysis on the dataset, We get three distinct classes: Food, Room Staff. Each of these classes consists a list of clustered words whose vectors are closest to each other.



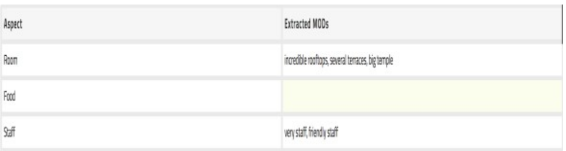
Analysis with Aspect

Enter Review Below

First thing: incredible rooftop! has several terraces and a 360 view of the city and the surrounding lakes. positively surprised that locals from Udaipur come to this hotel to have dinner which speaks well about it. Very friendly and helpful staff, they helped me with last minute bookings with no problem. The hostels location is great, in front of a big Hindu temple and 5 min walking distance to the City Palace.

Fig 3: A web Interface to accept input of a sample review

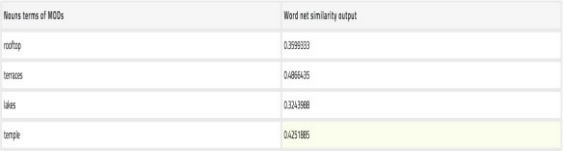
The web interface is used to take input of a particular review for analysis. The interface consists of a 'Textarea' and a 'Submit' button.



Aspect	Extracted MODs
Room	incredible rooftops, several terraces, big temple
Food	
Staff	very staff, friendly staff

Fig 4.Extracted MODs

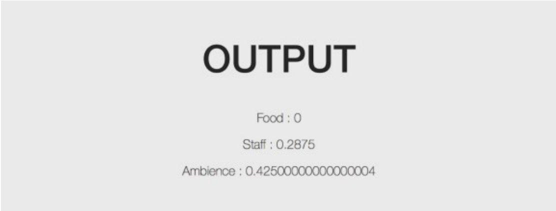
The MODs, AMODs and COPs from the text review are extracted one by one and are classified into one of our classes based on wordnet distance between noun part of MOD and all the elements in the lists.



Nouns terms of MODs	Word net similarity output
rooftop	0.3599333
terraces	0.4866435
lakes	0.3243988
temple	0.4251885

Fig 5 .Degree of Resemblance with 'Room' aspect.

The process of similarity calculation by comparison with list words of particular list is shown here. The aggregate score is calculated and decision is made based on maximum score.



OUTPUT

Food : 0

Staff : 0.2875

Ambience : 0.4250000000000004

Fig 6: A web Interface depicting output of Aspect Terms along with sentiment.

The output is shown by calculating sentiment values of the aspect terms. The values are then normalized to fit in the range(0,1). Note that the review does not talk about food, hence the value is 0.

V. Conclusion

An attempt was made to build a system make a rating system based on the rating of aspects. We managed to get the aspects directly rather than manually. So this system could essentially be applied to any data. The system improves exponentially with more amount of data. Advmod, Amod and cop work well with the given data.

For every new review that comes in use Stanford dependency parser to retrieve the grammatical components like a mod advmod and cop. It has been found to research that these components that are able to convey the meaning in the English language. It grammatical component that was extracted is in a two-part format where in the first part would explain the ceiling and the second part would be the noun associated with the feeling. e.g incredible house.

The retrieved components are then split. The part of the component which would be the noun is classified into one of the predefined aspect clusters. This is achieved through wordnet similarity where each word is compared with the rest of the aspect words. The word is classified into the aspect which gets the highest score. In this manner, all the words are classified into the respective aspects.

The grammatical component which conveys the feeling is then appended to the words. At this stage, we get a number of lists for a particular aspect which has to be used for sentiment analysis.

Sentiment analysis is performed using textblob library in Python. Text block have functions like sentiment polarity to obtain the sentiment value of the input. Texts are passed through this textblob library to get a score which is averaged to obtain the final score. Thus for every aspect defined we get a score based on the analysis performed by textblob.

References

[1] D. H. Sasmita, A. F. Wicaksono, S. Louvan and M. Adriani, "Unsupervised aspect-based sentiment analysis on Indonesian restaurant reviews," 2017 International Conference on Asian Language Processing (IALP), Singapore, 2017, pp. 383-386. doi: 10.1109/IALP.2017.8300623

[2] Naveen Kumar Laskari , Suresh Kumar Sanampudi,Aspect Based Sentiment Analysis Survey,IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 22780661,p-ISSN: 2278-8727, Volume 18, Issue 2, Ver. I (Mar-Apr. 2016), PP 24-28,(2016)

[3] Wanxiang Che ; Yanyan Zhao ; Honglei Guo ; Zhong Su ; Ting Liu,Sentence Compression for Aspect-Based Sentiment Analysis,IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 23, NO. 12, DECEMBER 2015, pp 1-4 ,(2015)

[4] Naaima Boudad , Rdouan Faizi, Rachid Oulad Haj Thami, Raddouane Chiheb, Sentiment analysis in Arabic: A review of the literature,Ain Shams Engineering Journal, pp 1-2,(2017)

[5] Deepa Anand , Deepan Naorem Semisupervised Aspect Based Sentiment Analysis for Movies using Review Filtering 7th International conference on Intelligent Human Computer Interaction, IHCI 2015

[6] Liu B. Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers; 2012.