**Project Report**

**Restaurant Review Summarization using part of speech tagging**

**Course: CS 6320 Natural Language Processing**

**Instructor: Dr. Moldovan**

Kumaran Rajendiran (kxr131430)

Table of Contents

Introduction: 3

Objective: 3

Related Work: 5

Dataset used: 5

Training phase: 5

Testing phase: 6

Feature Extraction and Classification: 6

Typed Dependencies for opinion extraction: 7

Analyze sentiment using WordNet expansion: 8

Sentiment Classification: 8

Evaluation and Results: 8

Conclusion and Future work: 9

References: 9

**Restaurant Review Summarization using part of speech tagging**

# Introduction:

Summarizing reviews based on sentiments is quite a challenging task. In the current scenario, there exists a lot of websites such as Yelp, we8there etc., which provide descriptive reviews for almost every restaurant. In order to understand the qualities of the restaurant, we need to read all the reviews associated with that restaurant which is nearly impossible for a person. Most of the times it is quite difficult for the users to understand the good and bad features of a restaurant and the reviews could confuse the users about overall positive or negative sentiments about specific features.

For classifying text, a basic text processor using a bag of words model would just look at the words that appear on the document and classify based on the probabilities computed in the training set and applying them on the test set. A different task altogether, would be to find the structure of the sentences in a document, by finding the parts of speech of words in each sentence.

At the end of this report, I would be presenting to you, the accuracy of my implementation on a test set.

# Objective:

The objective of this project is to extract the features from the restaurant reviews and classify them as ‘Good’, ‘Moderate’ or ‘Bad’ based on the positive and negative sentiments of the reviews. The features could be either general such as food, ambience, service etc. or specific to the restaurant. There will be three phases in identifying the solution for this problem.

1) Feature Extraction

2) Sentiment Analysis

3) Sentiment Classification

In the first step, the useful sentences about the restaurant will be extracted. Each sentence can belong to one of the following classes: ‘FOOD’, ‘PRICE’, ‘SERVICE’, and ‘AMBIENCE’. After the features from reviews have been extracted, we need to find the important sentiments fore each sentence. The second step is to associate the sentiments with one of the following classes: ‘GOOD’, ‘BAD’, and ‘MODERATE’. After extracting the features and their sentiments we need to summarize on these features and derive an overall sentiment for a particular feature.

Classify sentiments and summarize

Extract sentiments for each feature using Typed Dependencies

Semantic Analysis

Database of features

NN, NNS,JJ

Classify features

Extract important features

Restaurant review

Extract features

Part of speech tagging

Training set: Restaurant reviews

**TESTING PHASE**

**TRAINING PHASE**

WordNet Expansion

# Related Work:

The project is closely related to the idea proposed by Hu and Liu [1]. The idea is to extract frequent features and apply bootstrapping techniques to analyze sentiments. There have been several ideas proposed which are based on mining to automatically summarize the customer reviews. Such a system needs to identify the product features and also considers the opinions suggested by users in the reviews to generate a summary of the review. After the important features corresponding to a product has been determined, their system uses the WordNet expansion to determine the opinions and semantically orient those opinions which closely align with the features and make use of this information to generate a review summary. OPINE systems could also be used for extracting the features and associate the opinions to these features using the sentiment analysis method in opining mining. In this project, we classify each of our features into four classes: food, service, price and ambience. Combination of nouns and adjectives and their pair count helps us to prune unwanted features. WordNet expansions of positive and negative opinions along with part of semantic structure of the sentence have been used for the sentiment classification.

# Dataset used:

For the training phase, I have used the data set provided by Yelp. The Yelp Challenge dataset is much larger and richer than the Academic Dataset. The dataset can be downloaded from the following link:

www.yelp.com/dataset\_challenge

In order to classify the restaurant review features, I have used the original dataset of Ganu et al (2009). The dataset consists of over 3000 English sentences from the restaurant reviews each of which are labeled into one of the following classes: ‘FOOD’, ‘PRICE’, ‘SERVICE’, ‘AMBIENCE’, and ‘MISCELLANEOUS’.

# Training phase:

In the training phase, the important features from the dataset obtained from Yelp.com, which are specific to food, ambience, service and price were extracted. These features were then stored and were used for the sentiment analysis of the reviews.

I wrote a script in Java that would parse the raw dataset and would segregate the restaurant reviews. After this step, the reviews were classified into one of the three categories: GOOD, BAD and MODERATE based on the ratings provided by the users. There were around 77,000 bad reviews, 120,000 good reviews and 53,000 moderate reviews. Then the Stanford POS tagger was run on each of the reviews to tag each word in the reviews with its associated part-of-speech tag.

An example pos-tagged sentence:

If\_IN you\_PRP want\_VBP average\_JJ food\_NN and\_CC terrible\_JJ service\_NN this\_DT is\_VBZ your\_PRP$ place\_NN

The assumption is that the most of the features related to a restaurant would be either noun (NN, NNS) or adjectives (JJ, JJS, JJR). The script would then collect the nouns and adjectives and their frequency. After this step, the top 1000 most occurring nouns and adjectives were extracted in each of the categories. These would be features, which the users use the most while reviewing food, price, service, and ambience, which are specific to a restaurant.

# Testing phase:

## Feature Extraction and Classification:

In the process of summarizing the reviews for a restaurant, we need to identify the sentences related to food, price, service and ambience which specific to that restaurant. All the other sentences in the review need to be pruned, as they do not contribute to the sentiments associated with the review. In order to extract the important features of the review, I have written a script, which would tokenize the text into sentences. Then using the OpenNLP Document Categorizer API, the sentences are classified into one of the following classes: FOOD, SERVICE, PRICE, AMBIENCE, and MISCELLANEOUS. The Document Categorizer is trained on annotated restaurant review dataset of Ganu et el. The result of training is the maxent model, which is later used for classifying the text.

Example:

**Actual review:** I went to the restaurant named XYZ with my friend. The food was great. The pizza I had there was the best. The staffs were super attentive. The atmosphere was filled with aroma.

**Important features that were extracted:**

FOOD => The food was great

FOOD => The pizza I had there was the best

SERVICE=> The staffs were super attentive

AMBIENCE => The atmosphere was filled with aroma

## Typed Dependencies for opinion extraction:

After extracting the important features using the previous step, the Stanford typed dependency parser is run on these sentences to determine the dependency format for each sentence. These dependencies provide a representation of grammatical relations between words in a sentence.

Example:

**Sentence**: The food was excellent

**Typed dependencies generated by the parser:**

det(food-2, The-1)

nsubj(excellent-4, food-2)

cop(excellent-4, was-3)

root(ROOT-0, excellent-4)

After determining the typed dependencies, important opinions are extracted for each feature using its typed dependencies. The following dependencies are used for determining the opinion words.

**nsubj:** nominal subject. A nominal subject is a noun phrase, which is the syntactic subject of a clause.

Example: The food was excellent **nsubj(excellent-4, food-2)**

Here the opinion word related to the feature is ‘excellent’.

**amod:** adjectival modifier. An adjectival modifier of a noun phrase is any adjectival phrase that serves to modify the meaning of the noun phrase.  
Example: The management was super attentive **amod(attentive-5, super-4)** Here the opinion words related to the feature are ‘super’ and ‘attentive’.

**neg:** negation modifier. The negation modifier is the relation between a negation word and the word it modifies.

Example: The pizza was not hot **neg(hot-5, not-4)**.

Here the opinion word for this sentence is ‘not hot’. In this case, if the word ‘hot’ belongs to GOOD category then the category of the opinion ‘not hot’ is BAD.

## Analyze sentiment using WordNet expansion:

The assumption is that most of the opinion words are adjectives. Also there are three lists (each list belonging to one of the three categories: ‘GOOD’, ‘BAD’ and ‘MODERATE’) of frequently occurring adjectives, which I had obtained during the training phase. Since these lists don’t cover all opinion words, we need to expand the list using the word suggestions from WordNet. I have used Java API fro WordNet Searching (JAWS) for the expansion problem by considering the synsets of the opinion words, which are of the form AdjectiveSatellite. For each of these synsets I have extracted all the wordForms and obtained the expansion set of good and bad words.

## Sentiment Classification:

The count of good, moderate and bad opinions for each category: FOOD, SERVICE, AMBIENCE and PRICE are determined. Each category is then classified as ‘GOOD’, ‘BAD’ or ‘MODERATE’ based on the good, moderate and bad opinion count.

If the number of good opinions equals the number of bad opinions, then the class is ‘MODERATE’

Else if the number of bad opinions is greater than the number of good opinions and the number of moderate opinions, then the class is ‘BAD’

Else if the number of moderate opinions is greater than the number of good opinions, then the class is ‘MODERATE’

Else the class is ‘GOOD’

## Evaluation and Results:

**Feature**: FOOD

Number of sentences belonging to feature ‘FOOD’: 87

Number of sentences properly classified into the right category: 68

Percentage Correct = 68/87 = 78.2%

**Feature**: SERVICE

Number of sentences belonging to feature ‘SERVICE’: 49

Number of sentences properly classified into the right category: 37

Percentage Correct = 37/49 = 75.5%

**Feature**: AMBIENCE

Number of sentences belonging to feature ‘AMBIENCE’: 42

Number of sentences properly classified into the right category: 33

Percentage Correct = 37/49 = 78.5%

# Conclusion and Future work:

There are several methods to improve the results of our method

1. Currently, I have used simple heuristics to compute the boundary between various sentences in the review. The problems due to issues like missing punctuation, slang language etc., were not handled properly. The standard sentence boundary detection problem needs to be solved to get more realistic results.
2. The usage of bigram features in the training phase and testing phase (while extracting the opinions) could improve the accuracy of the sentiment classification. Also the context information around an opinion word should be analyzed.
3. The accuracy of the classifier could be improved if we use the labeled and biased training set and maintain a sub-list of features which could be used for opinion classification
4. A machine-learning multiclass classifier could be used for semantic classifier.

# References:

[1] M. Hu and B. Liu, Mining and summarizing customer reviews, in KDD ’04 : Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 168–177, New York, NY, USA, 2004, ACM

[2] Bo Pang and Lillian Lee, A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts

[3] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani, Istituto di Scienza e Tecnologie dell’Informazione Consiglio Nazionale delle Ricerche Via Giuseppe Moruzzi 1, 56124 Pisa, Italy : SENTIWORDNET 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining

[4] Manoj Pawar, Deepak Mallya : Automated Summarization of Restaurant Reviews

[5] Opinion mining and sentiment analysis Bo Pang and Lillian Lee

[6] **Java Api for WordNet Searching (JAWS)**

<http://lyle.smu.edu/~tspell/jaws/>

[7] **Stanford Log-linear Part-Of-Speech Tagger** <http://nlp.stanford.edu/software/tagger.shtml>

[8] **The Stanford Parser: A statistical parser**

<http://nlp.stanford.edu/software/lex-parser.shtml>

[9] **Apache OpenNLP Developer Documentation**

<https://opennlp.apache.org/documentation/1.5.2-incubating/manual/opennlp.html>

[10] www.yelp.com