Sentiment Analysis and Summarisation of Product Reviews

Aashi Manglik, Ashwani Gautam, Neeraj Kumar
Guide: Prof. H C Karnick

September 30, 2016

1 Problem Statement

In Amazon Fine Food Reviews dataset, a food product which is uniquely identified by its Product Id is reviewed by more than one user. It becomes difficult to go through all the reviews and find out the opinion or sentiment of majority. Our task is to extract the major opinion about the product and summarise it. For this purpose, we concatenated the different reviews of a product and process it as one document or text to generate summary. By summary, we mean a set of concise and non-redundant phrases of two to five words each which convey the key opinions in the text. In particular, the output will be a set of n-gram phrases $M=\{m_1,...,m_k\}$, where the number of phrases will be determined by the constraints of optimisation problem described in Ganesan et al [1].

2 Sentiment Classification

We started with analysing the dataset and perform sentiment classification on it. Initially, we assumed that doing the sentiment classification of reviews into positive and negative categories will also help in summarising the review, but the methodology for summarisation was quite different.

2.1 Preprocessing

The dataset contains a rating between 1 and 5 for each review. We considered the review as positive if the rating is above 3 and negative otherwise to implement a supervised approach for classifying sentiment after splitting the dataset into train and test data. To make data suitable for text classification, we removed the stopwords and applied lemmatizer to reduce inflectional forms followed by punctuation removal.

2.2 Tf-idf Representation

The most frequent 5000 words were chosen from the processed text and vectorised using Tf-idf transformer. The obtained matrices were to be used for training and evaluating the model.

2.3 Feature Reduction

The dataset is quite large to run a machine learning model and thus, it became important to reduce the size of feature set. In particular, the feature set was reduced to 200 components by applying Truncated SVD(a variant of Principal Component Analysis) on the sparse matrices.

2.4 Results

The trained logistic regression model and Decision Tree Classifier were tested on the computed test matrices. The accuracies were 80.47 for Logistic Regression and 84.3 for Decision Tree Classifier.

3 Approach for Summarisation

We explored an unsupervised approach to generate ultra-concise summaries of opinions as proposed in Ganesan et al [1]. Given a set of sentences S from a review text (text generated by concatenation of different reviews of the same product), we tried to generate a short phrase between 2 and 5 words representing the major opinion. For measuring this representativeness, we have calculated a representative score, S_{rep} described in section 3.2. We composed the new phrases using words from the original text. While we use the words that have occurred at least once in S, we do not require the phrase to be an exact subsequence of the text. Hence, this is more of an abstractive summarisation problem.

3.1 Generation of Seed Bigrams

As a first step, we generated a set of promising bigrams by combining the high frequency unigrams. In previous work [1], the unigrams with count larger than the median count (after discarding the words with frequency of 1) were shortlisted to form bigrams. Since we did not remove the english stopwords like *'and'* or *'a'* as they could be the part of phrase, the shortlisted unigrams contained pronouns like *'he'*, *'we'* and *'you'* which are irrelevant in concise summaries. Hence, we decided to do POS tagging of the review to bring in focus the most frequent nouns and adjectives. As a result, the most frequent nouns indicated the product class being talked about in the reviews to form bigrams like *'dog food'* or *'cacao powder'*.

3.2 Representative Score

To formulate the representativeness of a phrase, the two key properties are emphasized. The words in the phrase should be strongly associated within a contextual window C in the original review text and should be sufficiently frequent. The score is computed following the definitions in Ganesan et al [1]. Higher the score, better is the representativeness of phrase. We have chosen C to be 1. To compute the modified pointwise mutual information [1] between two words, we need to find the joint probability of the same two words occurring together. Initially, we only used the brown corpus to find these probabilities

and as a result, many potential word combinations like *'dog food'* have zero occurrence. To overcome this, we used the complete text in Amazon Fine Food Reviews combined with a genesis corpus available in nltk to compute the joint probabilities of words. Yet the problem could not be removed completely, the joint probability of few promising bigrams comes out to be 0.

3.3 Jaccard Similarity between two phrases

To see whether the two phrases in output are non-redundant or convey similar information, we used the Jaccard similarity [2]. As an example, the phrases *'good quality food'* and *'healthy food'* are considered similar while *'poor taste'* and *'not as advertised'* are non-redundant phrases. The problem with this similarity measure is that at present, it cannot distinguish between synonyms but since the phrase is atleast a bigram, it can be used to filter out redundant phrases.

3.4 Results

The unigrams shortlisted from the review text based on frequency satisfactorily constitute the words that can be combined to generate higher order n-grams to express the key opinions. At present, we have recorded the representative score of phrases consisting of two words(derived bigrams) for few products. In order to find the threshold for accepting the phrase as representative, more reviews and n-grams need to be assessed. It requires human effort to read the complete review text and see if the phrase expresses the major opinion or not and then, identify if the scores also reflect the same. As stated earlier, due to insufficient corpus, the probabilities for some relevant bigrams is 0, and thus the score returned is also 0.

4 Future Work

We have generated the potential bigrams which will form the summary, the bigrams in many cases are incomplete to express the opinion. We will work to combine this bigrams to trigrams and so on to form a phrase which optimises the sum of representative and readability scores [1].

References

[1] Ganesan, Zhai, Viegas. "Micropinion Generation: An Unsupervised Approach to Generating Ultra-Concise Summaries of Opinions." WWW 2012 – Session: Information Extraction. April 16–20, 2012, Lyon, France

[2] R. Real and J. M. Vargas. The Probabilistic Basis of Jaccard's Index of Similarity. Systematic Biology, 45(3):380–385, 1996