CMPUT 551 — Project Report

Adam St. Arnaud - ajstarna@ualberta.ca
Alexandr Petcovici - apetcovi@ualberta.ca
Janek Goergens - gorgens@ualberta.ca
Robert Post - rpost@ualberta.ca
Sankalp Prabhakar - sankalp@ualberta.ca

Instructor: Prof. R. Greiner
Supervisor: Dr. Greg Kondrak
Date: Friday, 12/Dec/2014

Contents

1 Introduction

1.1 The Problem

Sentiment analysis is the area of natural language processing (NLP) concerned with computationally identifying the emotion expressed by an author of a piece of text (positive, negative, neutral). Determining the sentiment of a piece of text has important implications for opinion mining, parsing users reviews and recommendation systems. Humans communicate in complicated ways and often don't strictly use literal language. Figurative language, such as sarcasm, irony, and metaphors, are quite prevalent in standard human communication. Hence, in order to create better representations of human language, systems must take figurative language into account.

1.2 Related Work

1.3 Motivation

2 Problem Formulation

2.1 Twitter Data

Twitter is a web platform where users can post short messages with up to 140 characters to broadcast things they want the world to know. These messages are called *tweets* and the whole Twitter system became very famous in the last years. That's the reason why there's a high interest in performing sentiment analysis on twitter data. We are provided with a set of 8000 tweets where each tweet was annotaded by 7 persons. Each one scored the given tweet on a range from -5 to 5 (where -5 indicates a very negativ, 0 a neutral and 5 a positive sentiment). The highest and lowest rate were ignored and the average of the remaining 5 scores is given for every 8000 tweets. This data is provided by *SemEval-2015 Task 11*.

2.2 Histogram

In figure 1 you can see, that the most tweets have a score next to negative 2

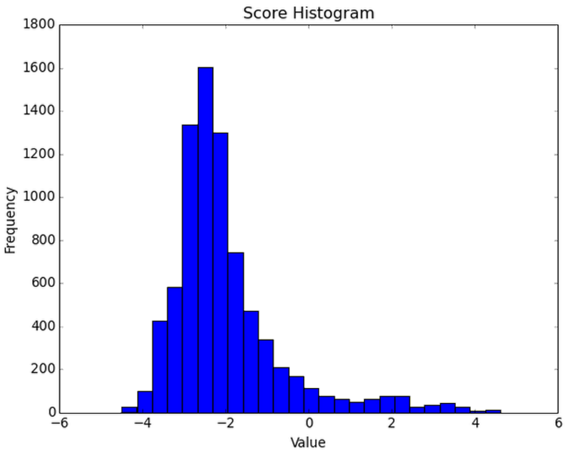

Figure 1: Histogram showing the score distribution for the given data

2.3 Learning Task

yes

2.4 Pipeline

2.5 Evaluation

3 Preprocessing

3.1 pp1

3.2 pp2

4 Features

4.1 n-Grams

4.2 Part of Speec (POS) Tagging

One of the most fundamental parts of the linguistic pipeline is the part-of-speech (POS) tagging, a basic form of syntactic analysis which has countless applications in Natural Language Processing (NLP). We studied some of the best POS tagging tools & settled on using Tweet NLP, a twitter specific POS tagger. WeâĂŹve used part-of-speech tagging as a feature for our task wherein every word/token in a given tweet is tagged based on its part-of-speech, some of which are twitter specific.

For our task, we selected 15 of the most common tags in linguistics (including the twitter specific tags). we made a count of the number of tagged tokens (with a confidence score $>$ 0.9) in a tweet & then used that as a feature.[rough work]

4.3 Sentiment

We used *SentiWordNet* to add features specifically related to the sentiment of the tweets. This is dictionary, designed for opinion mining, where each of the over 100,000 words is assigned a positive and negative sentiment score. For a given tweet, we calculate a positive sum feature: $p_{sum} = \sum_i^n p_i$, where piis the positive sentiment score from SentiWordNet for the ith word in a tweet with n words. We similarly calculate a negative sum feature. If a word is not found in the dictionary, then it does not contribute to either of the two sentiment features. These two features are added to the bag of words feature vector as a real number. It is possible that different preprocessing steps could affect the sentiment score of a tweet by modifying said tweet (such as by stemming); future experimentation is required to determine the effects of different combinations of preprocessing on sentiment score.

4.4 Irony

To account for possible irony in the tweets we implement two features based on the work of Reyes et. al [year]. The features are created to detect the so-called *counter-factuality* and *temporal compression* of a tweet. Reyes et.

al. determined that ironic tweets were more likely to have a high level of these two measures.

Counter-factuality: The first measure, counter-factuality, is focused on "discursive terms that hint at opposition or contradiction in a text, such as about, nevertheless, nonetheless, and yet." (Citation). The full list of counter-factual words includes 41 entrees; there are a total of 4187 occurrences of these words in the data set, and 3123 tweets contain at least one counter-factual word.

Temporal Compression: The second measure of tweet irony that we considered is temporal compression, which focuses on words related to an opposition in time, thus indicating an abrupt change in narrative. (citation) The list of temporal compression words contains 13 words such as suddenly, abruptly, and now. There are only 170 instances of a temporal compression word in the dataset, with only 103 tweets even containing a single instance.

Feature Creation: For each measure, we create a real-numbered feature based on the ratio of how many words in a given tweet possess the characteristic we are analyzing. That is, we have two ratio features $r_t = \frac{n_t}{n}$, where $t \in \{\texttt{counterFactuality}, \texttt{temporalCompression}\}$, n_t is the number of words in a given tweet that fit into category t, and n is the total number of words in the tweet.

5 Experiments / Results

5.1 Experiment 1

5.2 Experiment 2

6 Conclusion