Customer Experience Text Analytics for Social Media

Christoph Emunds, Benedikt Heinrichs, Dominik Nerger, Richard Polzin

March 15, 2017

1 Introduction

The project presented in this project plan focusses on finding and analyzing customers' opinions on products and their aspects from social media posts.

This project plan describes related work and the principals that we are going to build upon. We present the current state of the project explain what has been done so far. Finally, we define the goals for our project and give a schedule for the upcoming tasks.

2 Related work

2.1 Word embeddings

More concise than the *bag of words* model with one-hot encoded vectors, which, depending on the amount of words in the vocabulary, have a huge dimensionality, while only one of the vector's components is set to 1 and the rest to 0. Such simple models also fail to incorporate meaning of and similarities between words. For example in the sentences

> the cat got squashed in the garden on friday
>
> the dog got flattened in the yard on monday

simple language models do not capture the similarities between the word pairs *cat/dog*, *squashed/flattened*, *garden/yard* and *friday/monday*.

Today, word embeddings are used in a lot of natural language processing (NLP) tasks.

A famous example for the expressive power of word embeddings is the following:

$$king - man + woman \approx queen$$

This equation shows that if one were to subtract the vector representation for the word *man* from the vector for *king* and add the vector for *woman*, this would result in a point near to the vector for the word *queen*.

The high dimensional vector representations of the words can be processed with the *t-SNE* dimensionality reduction algorithm to be able to plot them in a two dimensional vector space.

2.2 POS-Tagging

POS-Tagging (part-of-speech tagging), also called grammatical tagging or word-category disambiguation, refers to the process of identifying partical parts of speech like nouns or verbs.

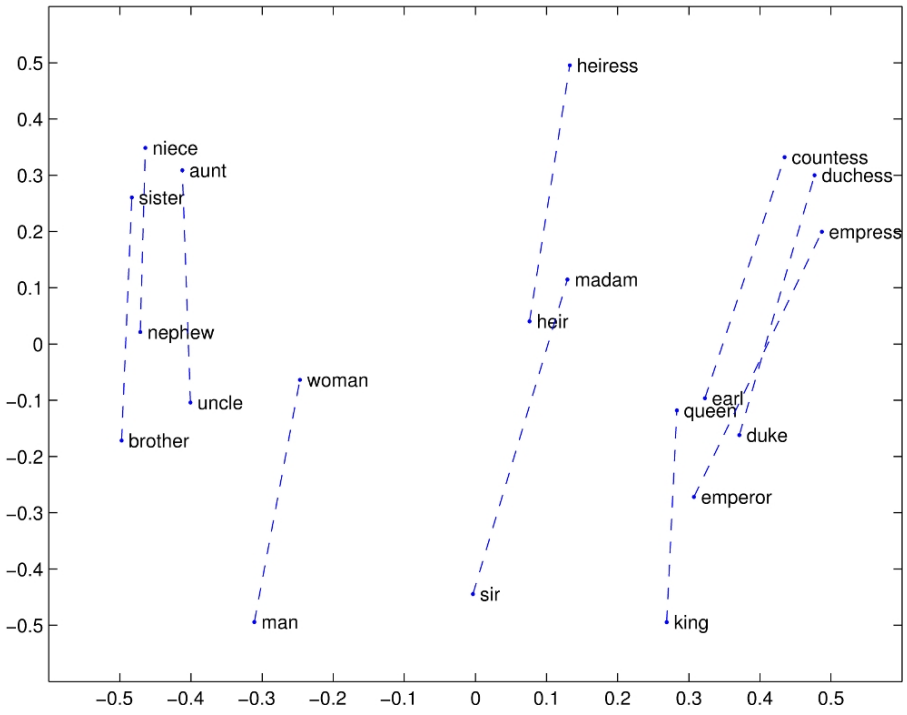

Figure 1: Representation of words relating to the differentiation between man and woman

Identifying the role of a certain word within a sentence is important for the task of sentiment analysis, as identifying entities and their corresponding aspects can be handled much easier relying on certain assumptions about the part of speech of words.

Frequent nouns or noun phrases often describe aspects of products and the vocabulary that is used to describe those usually converges. With such an approach many important aspects can easily be found, but on a more general scheme grammatic based relationships can be used to extract important aspects.

For example words that express opinion on something, like 'great' or 'bad' can be incorporated in rules to extract aspects. To sum it up POS-Tagging is an important part of sentiment analysis.

While being important POS-Tagging is also a complex problem. Word-forms in natural language are often ambiguous. For example the word 'dogs' in usually thought of as a plural noun, but can be used as a verb as well:

> The sailor dogs the hatch.

Due to the complexity machine learning techniques are often applied in POS-Taggers. Popular approaches such as the Viterbi algorithm, the Brill tagger or the Baum-Welch algorithm work with techniques such as dynamic programming, supervised learning or hidden markov models.

2.3 NER-Tagging

2.4 Dependency parsing

3 Extracting entities and aspects

Extract the posts' text into separate files.

we will use a method proposed by Hu and Liu (2004) to extract entities and aspects.

Annotate all files with POS-Tags.

Identify frequent nouns and noun phrases with a POS-Tagger. Frequency threshold is chosen experimentally. People commenting on aspects of a product usually use a limited vocabulary. Thus, frequent nouns and noun phrases are most likely important aspects.

However, since the first step can miss quite a lot of important aspects, the next step tries to find some of them by exploiting the relationships between aspects and opinion words.

4 Determining sentiment polarities

We will focus on two approaches

4.1 LSTM-Network

4.2 Lexicon-based approach

lexicon-based approach, because it is unsupervised and generally more powerful than supervised learning(?)

We will apply a method by Ding, Liu and Yu (2008), which has the following steps: First, all sentiment words and phrases in the text will be marked with the help of a sentiment lexicon.

Second, sentiment shifters will be applied.

Third, but-clauses must be handled.

Fourth, opinions need to be aggregated.

This method can be made more effective by applying dependency parsing to the sentences, such that the scope of each individual sentiment word can be determined more accurately. This allows us to discover the sentiment orientation of context dependent words (Bing Liu, 2012).

5 State of the project

We built and tested the syntax parser *Parsey McParseface* (aka SyntaxNet) by Google, which we will use as POS-Tagger for extracting entities and aspects, because it is easy to use and yields state-of-the art results. Moreover, it is based on Google's Deep Learning Framework TensorFlow and therefore integrates nicely into the rest of our implementation.

Use pre-trained word embeddings from the GloVe project.

We preprocessed the data by extracting the actual text of the posts into separate files, which can be processed by SyntaxNet.

We will utilize the sentiment lexicon by Bing Liu, which includes mis-spellings, morphological variants, slang and social-media mark-up of 2006 positive and 4783 negative words.

Table 1: Per-token accuracy of different POS-Taggers

Model	News	Web	Questions
Ling et al. (2015)	97.44	94.03	96.18
Andor et al. (2016)	97.77	94.80	96.86
Parsey McParseface	97.52	94.24	96.45

Table 2: Accuracy of different dependency parsers

Model	News	Web	Questions
Martins et al. (2013)	93.10	88.23	94.21
Zhang and McDonald (2014)	93.32	88.65	93.37
Weiss et al. (2015)	93.91	89.29	94.17
Andor et al. (2016)	94.44	90.17	95.40
Parsey McParseface	94.15	89.98	94.77

6 Goals and schedule

The overall goal is to extract as many quintuples $(e_i, a_{ij}, s_{ijkl}, h_k, t_l)$ as possible, where e_i is the i'th entity, a_{ij} is the j'th aspect of entity i the opinion is expressed on, s_{ijkl} is the sentiment polarity, h_k the opinion holder and t_l the time at which the opinion was expressed.

Build a database with opinions and statistics on products and their aspects. Visualization via dashboard.