

Politeness analysis: Planning

Carla Groenland (10208429), Harrie Oosterhuis (10196129),
Joost van Doorn (10805176), Ties van Rozendaal (10077391)

November 10, 2014

The assignment of our group consists of two parts:

- Find discriminative features for politeness and build a classifier based on the given data sets
- Make a visualization system that highlights polite and impolite parts of a sentence. If time allows, let the system give suggestions to make the sentence more polite.

We have split the given data in a test and train set. We will evaluate the system based on the performance on the (unseen) test set. In order to build a good classifier for politeness, we propose to compare the following:

- Baseline: represent sentences by a histogram of word frequencies. Use the training data to give a score to each individual word (by taking the mean score of all sentences it occurs in). Use these scores to score new sentences.
- Simple improvements on baseline: use a different type of classification system on the histogram of frequencies. Add POS tag information to the histogram. Add information about POS tag sequences to the histogram (attach to a word w_n the POS tags of w_{n-1}, w_n, w_{n+1}).
- Data preprocessing: many words, such as technical words, may have no influence on the politeness of the sentence. We can try to filter these by building decision stumps for single words and measuring the entropy and only keep words in the histogram that gave a sufficient amount of information gain.
- Extract parse subtrees using DOP and use these as features. The subtrees that appear only once or twice are left out.

Using entropy measures, we can try to find discriminative subtrees and words for politeness and use these to highlight polite or impolite sentences. If time allows, we may also give suggestions for changes to the sentence by generating changes to the sentences, classifying the new sentences and give the most polite change as suggestion to the user.

We divided the work as follows:

- **Bag of Words** *Carla and Joost*
 - Dimensionality reduction

- Data preprocessing
- Feature extracting
- Classification

- **Data Oriented Processing** *Harrie and Ties*

- Using existing DOP library (Disco-DOP)
- Make parser for data (sentences -> subtree) feature space will be subtrees of sentence
- Process all the data once and save efficiently
- Build classifier for DOP processed data, (try out multiple with existing libraries to pick a learning algorithm, implement the best one ourselves)

- **Evaluation** *Carla and Harrie*

- Combine BOG and DOP classifiers
- Interpret results, to recognize what words make sentence (im)polite, or what sentence structures make the sentence (im)polite
- Think of method that can generate suggestions which increase the politeness of the sentence

- **Visualization** *Joost and Ties*

- Finding a interface which can communicate with the classifiers
- Displaying sentences, and DOP trees, which trees/words cause (im)politeness
- Display expected politeness comprehensively
- Find a way to display suggestions