HANNES
HESSE

mail 2056 Emerson Street
Berkeley, CA 94703
phone 1 (510) 388-4214
email hannes@backin.de

December 13, 2006

IS256: Applied Natural Language Processing
Final Project

Email classification for semi-automated reply generation

INTRODUCTION

In this project, I examined methods to classify a corpus of emails by their content in order to suggest text blocks for semi-automatic replies.

The front desk of Youth For Understanding, a non-profit youth exchange organization based in Germany (www.yfu.de), receives 10-15 emails per day. These are mostly from students inquiring about exchange programs in general, requesting application material, following up on applications, and other kinds of questions. A survey of the email corpus indicates that a fairly large portion of these emails could be grouped into perhaps 10-20 distinct topical categories.

The long-term motivation for this project is to develop a system which suggests replies to incoming emails built from common text blocks that address the inquiry. An additional, more short-term, motivation is to automatically route emails to recipients within the organization based on the messages' contents.

TRAINING CORPUS

Emails were generated by various contact forms on the website. Out of an initial collection of 2.284 messages, 1.820 were selected that were generated by the main contact form and that were not spam. With the exception of a list compiled with first names, all personal information like sender's names and addresses was deleted prior to further processing. The collection covers a date range of about 16 months.

The largest portion of the messages came from interested students requesting information brochures or application material. This was surprising because the website offers a separate, more structured form for this purpose. Typical other questions included

- inquiries about age ranges for eligible applicants,
- application deadlines,
- durations of the exchange programs
- expression of interest to host an exchange student
- inquiries about application status

and more. Messages were often written in a colloquial style, with spelling errors, incomplete punctuation and abbreviated words. The average word length of a message was 45.5.

PREPROCESSING

Prior to indexing and classification, a number of preprocessing steps were performed.

1. Emails were converted to plain-text from .mbox files.
2. Headers and HTML components were removed.

3. Author name and body of the message were extracted.
4. Special characters (ä, ö, ü, ß) were replaced by equivalent letter combinations (ae, oe, ue, ss).
5. The message body was tokenized into words, stopwords were removed, and words were converted into lower case and stemmed using a German version of the Porter algorithm.

CLUSTERING

Because of the lack of pre-categorized training data, the initial approach was to apply a clustering algorithm to all documents, identify meaningful clusters and use those as prototypes for the classifier.

The open-source data mining framework Orange (www.ailab.si/orange) was used to explore characteristics of the training data and perform clustering. In preparation, the 500 most common terms in the collection were selected and a file of all messages and their term occurences compiled.

A K-Means clustering algorithms was performed on the data, with varying numbers of clusters. The sizes of the obtained clusters were very uneven, however, and upon manual inspection did not prove to represent any kind of sensible grouping.

MANUAL CLASSIFICATION OF TRAINING DATA

In order to obtain a training corpus for supervised learning algorithms, emails were classified by hand, using a category set devised step by step. 350 email messages were classified into 26 categories. Later, the number of categories was reduced to 10. However, even with this reduced category set, the distribution of the categories was quite uneven. The collection was dominated by the large portion of emails about application material, which was roughly ten times larger than most categories.

Some emails clearly belonged in more than one category because they touched on several subjects. In those cases, a decision had to be made which category describes them best.

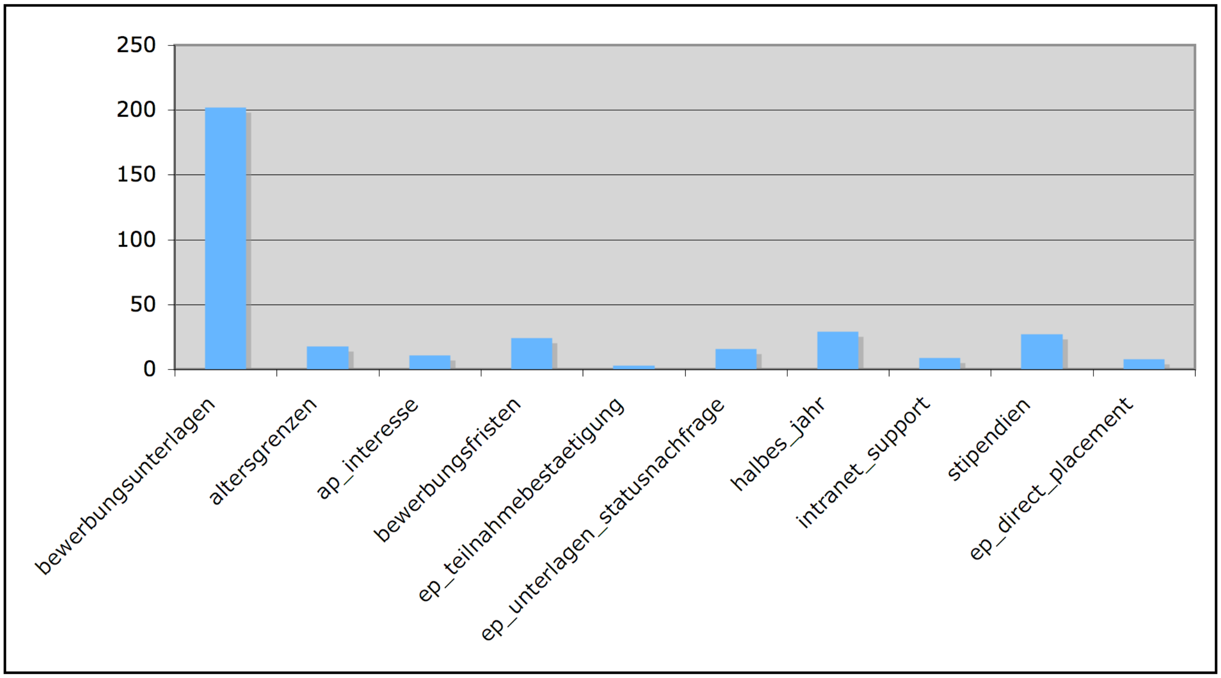


Figure 1: Distribution of category sizes

The categories' meanings were as follows:

bewerbungsunterlagen	Requests for application material and information brochures on paper
altersgrenzen	Questions concerning the age limits for eligible applicants
ap_interesse	Interest to host an exchange student
bewerbungsfristen	Questions about application deadlines
ep_teilnahmebestaetigung	Requests for a certificate documenting an exchange year
ep_unterlagen_statusnachfrage	Follow-ups on applications
halbes_jahr	Questions regarding the duration of exchange programs (typically the request if a 6-month stay is possible)
intranet_support	Questions concerning the organization's intranet
stipendien	Questions concerning scholarships
ep_direct_placement	Questions about placements in a previously known host family

FEATURE SELECTION

Prior to processing in Weka, features were selected in the preprocessing phase. Initially, stopwords were removed, using a list of 239 words. Experimentally, words were stemmed at this point, using a German version of the Porter algorithm.

Subsequently, 500 features were selected out of several thousands. Several approaches were considered:

- Weighting by term frequency: The terms most common in the collection were selected. This apparently simple selection criterion yielded the best results out of the approaches considered, corresponding to the findings in [1].
- TF-IDF weighting: Terms with the highest TF-IDF scores were selected. This did not result in a higher classification accuracy. This is likely due to the fact that high TF-IDF scores indicate that a term characterizes a single document well, but may not be informative about which category a document belongs to.
- Dispersion: As a variant to TF-IDF, terms were considered just by their dispersion over the collection, that is how many documents a term occurs in in proportion to its total number of occurences. The dispersion measure was calculated as DocumentFrequency(term) / TermFrequency(term).

Further feature reduction was applied using the weka.filters.supervised.attribute.AttributeSelection attribute filter in Weka. The filter reduced the number of features from 500 to 16. When using a NaiveBayes classifier, this improved the percentage of correctly classified emails from 67.5% to 73.9%.

When using stemming prior to other feature selection methods, accuracy improved to 76.6%.

CLASSIFIER SELECTION AND PERFORMANCE

Weka was used to experiment with different classifiers and settings. The following classifiers were considered:

- Naive Bayes classifier
- Multinomial Naive Bayes Classifier
- Bayes Network classifier
- Support Vector Machine

The best results were obtained with the Naive Bayes classifier using a reduced feature set. The classifier correctly classified 268 of the 350 messages, resulting in an overall classification accuracy of 76.6%. Cohen's Kappa value [2] was .61.

For the largest class, the precision rate was .844, and the recall rate was .931. Emails belonging to the smallest class „ep_teilnahmebestaetigung" (3 members) were all all misclassified. The complete confusion matrix is shown below:

A	B	C	D	E	F	G	H	I	J	
190	1	3	0	0	0	4	0	6	0	A: _BEWERBUNGSUNTERLAGEN
6	6	0	2	0	1	1	0	2	0	B: _altersgrenzen
5	0	4	0	0	0	0	0	0	2	C: _ap_interesse
6	0	0	15	0	1	1	0	1	0	D: _bewerbungsfristen
2	0	0	1	0	0	0	0	0	0	E: _ep_teilnahmebestaetigung
3	0	0	2	0	10	0	0	1	0	F: _ep_unterlagen_statusnachfrage
1	0	1	1	0	0	25	1	0	0	G: _halbesjahr
2	0	0	0	0	0	1	6	0	0	H: _intranet-support
10	4	0	1	0	2	3	0	8	0	I: _stipendien
0	0	2	0	0	0	1	0	1	4	J: _ep_direct_placement

Figure 2: Confusion matrix of classification results

The very uneven distribution of the category sizes in the training set appeared to be the largest problem. There seemed to be a bias toward the largest category, resulting in a high false positive rate of .24. However, reducing the size of this category by deleting most of its members impaired overall performance, rather than improving it.

In the future, using a larger training set or revising the category set may improve classification accuracy.

DISCUSSION AND OUTLOOK

The project gave interesting insights about the nature of incoming emails. While the meanings of some emails could successfully be detected by the classifier, other inquiries were too specific for automated replies. The high recall rate on the largest category with requests for application material indicates that these inquiries could in the future be processed semi-automatically. An address extraction algorithm could create database entries of requests and prepare shipping labels, reducing the need for tedious cut-and-paste operations of address data.

A reliable method for suggesting meaningful replies seems to be far away. Based on a small category set like the one used in this project, routing emails to appropriate recipients within the organization seems to be more feasible. It would, however, be interesting to examine the lexical relationships between incoming emails and the actual replies sent (which were not part of the training corpus) in a future study.

From this project, I gained detailed insights in the possibilities and limitations of text classification and the amount of human attention needed to adequately engage in interaction situations such as customer relationship management.

OTHER FINDINGS

Some other statistics on the email corpus were computed which yielded findings useful for the refinement of the website's usability.

A fair number of emails mention one or more country names. These were extracted and their distribution was matched against log files from the website. The site has a subsection with country descriptions. Traffic on these pages is a good indicator for potential applicants' interests in different countries. The distribution of emails mentioning country names and of traffic on country pages were quite similar:

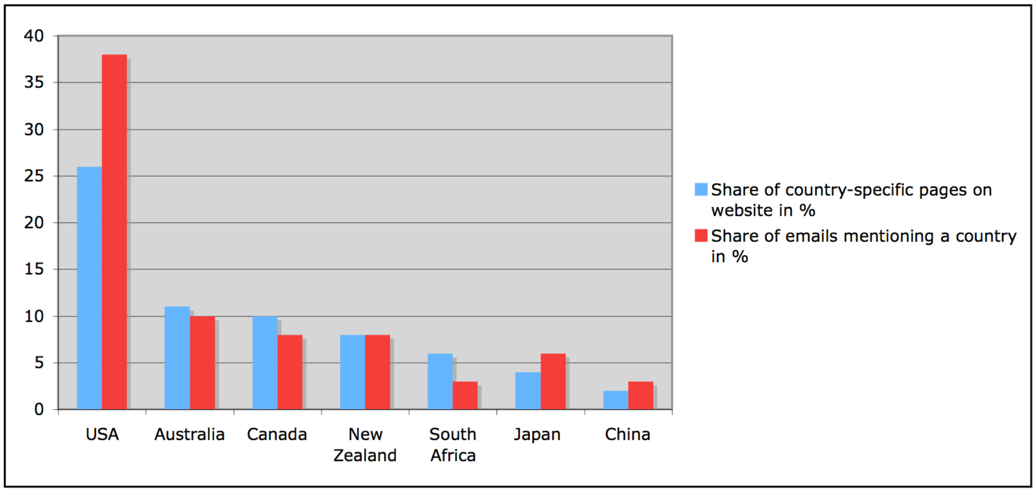
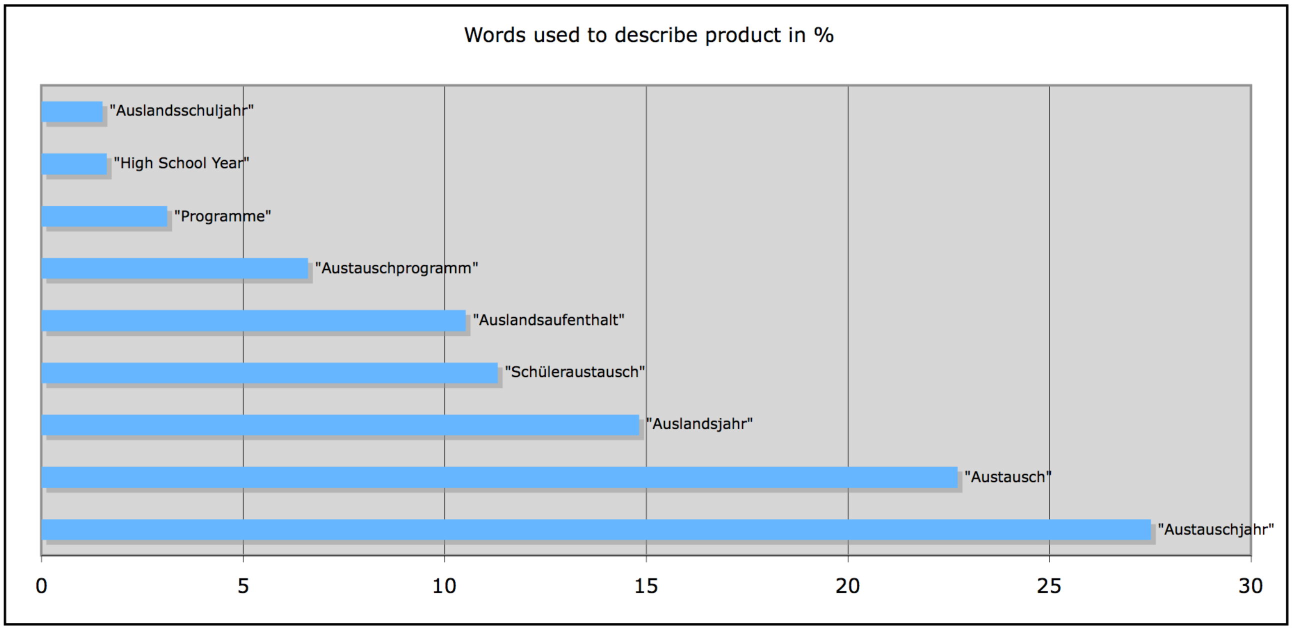


Figure 3: Comparison between mentions of countries in emails and traffic on country-specific pages of the website.

Different terms describing the product (an student exchange year) were extracted from the corpus. Their distribution reflects views prevalent among potential applicants and gives cues to improve wording on the website.



TOOLS USED

All text data was processed using Python and the nltk_lite framework. Initial data exploration was conducted with the open-source data mining framework Orange (www.ailab.si/orange) after preparing data files with selected features with Python scripts. Orange allows the user to combine a large number of statistical processors like classifiers, clustering and visualizations, using a plug-and-play-style graphical interface. Overall, the Orange software package seems well-suited for exploratory tasks and rapid prototyping of statistical methods. The processors (called „widgets“) are written in Python themselves, so they can be used outside of the Orange framework.

Most of the experimentation with classifiers and feature reduction was conducted using Weka, a Java-based data mining framework with a graphical user interface (www.cs.waikato.ac.nz/ml/weka). Data files for processing in Weka were generated using a slightly modified version of Marti Hearst's Weka tools (weka.py).

REFERENCES

[1] Yang, Y. and Pedersen, J. O. 1997. A Comparative Study on Feature Selection in Text Categorization. In Proceedings of the Fourteenth international Conference on Machine Learning (July 08 - 12, 1997). D. H. Fisher, Ed. Morgan Kaufmann Publishers, San Francisco, CA, 412-420.

[2] Cohen, Jacob 1960: A coefficient of agreement for nominal scales. In Educational and Psychological Measurement 20 (1960), pp. 37–46