EXPLORING SUB-WORD FEATURES AND LINEAR SUPPORT VECTOR MACHINES FOR GERMAN SPOKEN DOCUMENT CLASSIFICATION

Martha Larson*, Stefan Eickeler*, Gerhard Paaß**, Edda Leopold**, Jörg Kindermann**

*Fraunhofer Institute for Media Communication (IMK), Sankt Augustin, Germany
**Fraunhofer Institute for Autonomous Intelligent Systems (AIS), Sankt Augustin, Germany

ABSTRACT

Using sub-word features for spoken document classification raises two potential drawbacks. First, if the speech recognizer recognizes sub-word units directly, the risk arises that word-level discriminative features are irretrievably lost. This effect is aggravated by depressed recognition accuracy, such as that associated with speaker- and domain-independent systems. Second, if input documents are expanded by combining sub-word units into higher-level features, in compensation for lacking word-level discriminators, the size of the classifier input space expands rapidly, inviting the danger of over-fitting. This paper reports results of experiments with a simple, but real-world, binary topic classification task on a corpus of un-edited German-language radio documents. We compare a Naive Bayes classifier to a Linear Support Vector Machine (LSVM) and determine that benefits of sub-word features indeed outweigh potential drawbacks. The LSVM in particular profits from sub-word features supplemented by higher-order combinations, reflecting its ability to control input space complexity independently of dimension.

1. INTRODUCTION

In a spoken document information retrieval system the correlation between speech recognition error rate and retrieval performance is strong, but not absolute. This effect does not go unobserved in the information retrieval literature [1][2][3]. If spoken document classifiers have at least a small margin of tolerance towards speech recognition errors, recognizer modifications that optimize for criteria other than error rate become attractive subjects of investigation. This paper uses a straightforward binary classification task performed on a database of un-edited German radio programs to explore whether exploitation of this margin of tolerance will allow benefit to be derived from the introduction of sub-word features. We compare the performance of a Naive Bayes classifier to a Linear Support Vector Machine (LSVM), a classifier well-known for its ability to generalize in high dimensional spaces.

Two examples of criteria that go above and beyond error rate and that are relevant for a mutual optimization of speech recognition and document classification are of particular concern to us. First, we are interested in optimizing speech recognizer output to aid a classifier disadvantaged by low training data levels, and second, we are interested in making the spoken document classification system as a whole speaker and topic independent.

Text document classifiers profit from sub-word information, especially in cases where available training data is inadequate, and the system we test here demonstrates just such an effect. Extending this generalization to spoken document classification means using a syllable-based language model for speech recognition. Such a model implies extra noise on the syllable level because of errors due to combinations that would not have been part of the search space in a recognition system with a word-based language model. The results presented here show that document classifiers have the generalization potential to bridge this gap.

In addition to compensating for limited classifier training data, using a syllable-based language model for speech recognition has a positive effect on the domain independence of the system. A syllable-based language model can be trained on a moderate amount of data, and even on a completely different domain. Such a model is also pleasingly compact.

Although feature distributions can be estimated more robustly with shorter and therefore more numerous features, if discriminative information is concentrated at word level, sub-word feature distributions risk being quite indistinctive. Supplementing the feature inventory by combining simple features into higher-level features, helps to avoid loss of discrimination, but causes a drastic expansion of the input space, accompanied by the threat of over-fitting.

In section 2. we introduce the database on which the experiments were performed. In section 3. we describe the extraction of features from the audio documents, introducing the speech recognition system and its syllable-based language model. In section 4. the two classifiers, Naive Bayes and LSVM are discussed. Section 5. describes our experiments and presents the results. Conclusion and outlook are in section 6.

2. DEUTSCHE WELLE KALENDERBLATT DATABASE

The data corpus used for the experiments presented here consists of 841 radio programs and their accompanying transcripts downloaded from the Deutsch Welle Kalenderblatt website http://www.kalenderblatt.de. Each program is about 5 minutes long and contains approximately 650 running words. The variation present in the content of the Kalenderblatt...
database makes it an interesting resource for experimentation and ensures that even the straightforward binary classification we are looking at here is not an artificial task. The database contains programs related to a broad range of topics of current, historical or cultural interest. The programs are narrated by about 10 different speakers and are liberrally interspersed with the voices of people interviewed or with original sound footage. Sound effects and background music are frequent, as is usual in educational entertainment for radio. Preparation of the database, described in detail in [4], included downloading the audio (mono Real format, 31.1 Kbps, 22.05KHz), and re-sampling to 16kHz, downloading the transcripts, normalizing the texts and annotating each document in the database with a topic category.

As topic categories we use the top level of the International Press Telecommunications Council (IPTC) subject reference system, which contains 17 categories. The system is available at http://www.iptc.org. Each document was annotated by a pair of human annotators. On an average, the pair only agreed 70% of the time about the category of the document. A third annotator broke the ties. Although the disagreement between the annotators indicates that the classification task that we have chosen is a challenging one, it also means that the maximum performance that we can expect from the system is 70% accuracy.

Although there are 17 different categories, we focused our experimentation on building classifiers for the topic categories of Politics, a large category, and Science, a medium category, chosen to highlight the effects of limited available training data. Table 1. presents statistics concerning these categories.

<table>
<thead>
<tr>
<th></th>
<th>docs</th>
<th>words</th>
<th>syllables</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>tokens</td>
<td>types</td>
</tr>
<tr>
<td>Politics</td>
<td>220</td>
<td>143 k</td>
<td>20 k</td>
</tr>
<tr>
<td>Science</td>
<td>105</td>
<td>68 k</td>
<td>13.7 k</td>
</tr>
<tr>
<td>All</td>
<td>841</td>
<td>547 k</td>
<td>55 k</td>
</tr>
</tbody>
</table>

Table 1: Statistics of the Kalenderblatt database

3. EXTRACTING SUB-WORD FEATURES FROM AUDIO DOCUMENTS

3.1. Training the Speech Recognition System

The HMM-based speech recognition system we use for these experiments is built with the ISIP public domain speech recognition toolkit [5]. Because our goal is to implement a domain independent spoken document classifier, we do not train the speech recognition system on, or adapt it to, our test domain, the Kalenderblatt database. Instead, our inventory of 50 monophone models is trained on 33k sentences from the Phondat/Siemens100 speech database. Our language model is a bi-gram syllable model and is trained on 11 million transcribed words from the German Parliament, an Internet resource available at http://www.bundestag.de.

In order to train the syllable language model, we decompose the German Parliament corpus into syllables using the transcription module of the Bonn Open Source Synthesis System (BOSSII) [6]. This module applies a combination of linguistic rules and statistical correspondences to decompose German orthographic words into syllables. Syllables with the same pronunciations are merged. The syllable model is trained on the top 5000 syllables in the German Parliament corpus, which is a third of the total types present. Syllables are theoretically limited in length (8 phonemes in German) and thus a finite set, but should still be selected for language modeling using a cut-off frequency. Exploratory experiments indicate that by keeping the syllable inventory relatively small and using bi-grams we are able to effectively abstract away from the domain of training and achieve a general German syllable model, even on only 11 million words of training data.

3.2. Decoding

We decode the 841 audio documents in the Kalenderblatt database using the ISIP decoder with the monophone acoustic models and the syllable-bi-gram language model described in sub-section 3.1. We choose a fixed-length segmentation, dividing the audio documents into 20 second sections, leaving all music and sound effects intact. By not pre-processing audio, we shift the burden of separation of speech and non-speech onto the document classifier; the speech recognition component remains simple and domain independent.

Because our fixed-length segments, don’t correspond to sentences, we run the risk of increased error rate due to edge effects. We compensate by using segments with 50% overlap meaning that all audio frames are effectively decoded twice. Once again we shift the burden of classification away from the speech recognizer, with the idea that the document classifier will learn the relevant patterns in the decoder output. If there are regularities in the decoder error, the classifier should be able to exploit them. If there are no interesting structures in the error, the classifier should effectively ignore it as noise.

We decode in 6 times real time (effectively 12 times due to overlap) and the syllable error rate, which we measured only on a sample set of the radio programs, hovers around 70%.

3.3. Extracting features from recognizer output

The syllable output of the recognizer is mapped into the different combinations of sub-word and multi-level features we used for the experiments. First, a syllable bi-gram version of the documents is generated, where in addition to all recognized syllables, each document is expanded with features representing all syllable pairs. Second, the syllable output of the recognizer is broken down into a sequence of phonemes which are recombined into n-grams. The phoneme bi-gram version of the documents contains all bi- and uni-grams present in the output and the tri-gram version contains all tri-, bi- and uni-grams.

By adding higher order features to documents in this way, we are actually emulating the effects of string kernels, which are popular in bio-informatics, and have proven effective in text classification [7].

4. DOCUMENT CLASSIFICATION

4.1. Naive Bayes Classifier

The Naive Bayes Classifier chooses the category that the document belongs to by calculating the probability that the document belongs to the class in question \( P_{pos}(\text{cat}|\text{doc}) \) and the probability that the document belongs to the complement class \( P_{neg}(\text{cat}|\text{doc}) \) using a decomposition with the familiar Bayes Rule (1). The document is classified into the class which
yields the higher probability. The Bayes classifier is called naive since $P(\text{doc}|\text{cat})$ is assessed by calculating the individual probabilities of the features that compose the document (words, syllables, phoneme strings), under the assumption that each feature acts on its own and contributes independently to the probability of the document.

$$P(\text{cat} \mid \text{doc}) = \frac{P(\text{doc} \mid \text{cat})P(\text{cat})}{P(\text{doc})} \quad (1)$$

Our implementation of the Naive Bayes classifier discards the denominator of (1), which is constant for both values being compared, and also assumes a flat distribution of prior probabilities. Additionally the classifier uses the feature inventory of the positive class to estimate distributions for both positive and negative classifiers, an empirically justified expediency that helps circumvent normalization issues. We chose the Naive Bayes Classifier because it is a well-known, easy-to-implement parametric learning machine, that provides a convenient baseline and an interesting comparison for the SVM, a non-parametric learning machine.

4.2. Support Vector Machines (SVMs)

Support Vector Machines (SVMs) have proven to be effective text document classifiers [8][9]. Because they are able to exploit otherwise indiscernible regularities in input spaces and because they control complexity independently of dimension, SVMs are ideal candidates for classification of our sub-word and multi-level feature expanded input documents. SVMs perform a binary classification by using a kernel to map the input space into a space in which it stands to yield to linear separation. The decision boundary is the hyper-plane which maximizes its distance (margin) to all the data points, allowing SVMs to control complexity independently of the size of the input space [10]. Since text classification has been shown to be a linear problem [8], and since exploratory research with other kernels did not yield performance improvements, we use only linear kernels here, meaning that the Naive Bayes classifier reaches a maximum, and then declines with additional features. This trend attests to the build-in over-fit protection of SVMs.

Table 2 presents the results from the baseline experiments, which were performed on the text transcriptions of the Kalenderblatt database. Two interesting observations can be made about the text document classification. First, classifier performance improves when syllables instead of orthographic word forms are used. This improvement is particularly marked in the case of the topic category Science, for which less training data is available. Second, the LSVM is able to make use of the addition of higher-level features, whereas the performance of the Naive Bayes classifier reaches a maximum, and then declines with additional features. This trend attests to the build-in over-fit protection of SVMs.

<table>
<thead>
<tr>
<th>Classification of Text Documents</th>
<th>Topic: Science</th>
<th>Topic: Politics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word 1-grams</td>
<td>35.8</td>
<td>50.7</td>
</tr>
<tr>
<td>Word 2-grams</td>
<td>30.0</td>
<td>48.4</td>
</tr>
<tr>
<td>Syllable 1-grams</td>
<td>30.4</td>
<td>58.9</td>
</tr>
<tr>
<td>Syllable 2-grams</td>
<td>34.0</td>
<td>47.0</td>
</tr>
<tr>
<td>Phoneme 1-grams</td>
<td>40.7</td>
<td>54.9</td>
</tr>
<tr>
<td>Phoneme 2-grams</td>
<td>65.6</td>
<td>65.7</td>
</tr>
<tr>
<td>Phoneme 3-grams</td>
<td>54.7</td>
<td>58.8</td>
</tr>
</tbody>
</table>

Table 2: Classification of Text Documents reported as F-measure (human scores about 70 on this task)

Table 3 presents the results from the spoken document classification experiments. In these results we again observe that the LSVM benefits from the addition of higher level n-gram features, whereas the Naive Bayes becomes swamped when too many features are added.

<table>
<thead>
<tr>
<th>Classification of Spoken Documents</th>
<th>Topic: Science</th>
<th>Topic: Politics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syllable 1-grams</td>
<td>38.0</td>
<td>56.0</td>
</tr>
<tr>
<td>Syllable 2-grams</td>
<td>23.0</td>
<td>43.0</td>
</tr>
<tr>
<td>Phoneme 1-grams</td>
<td>25.5</td>
<td>44.7</td>
</tr>
<tr>
<td>Phoneme 2-grams</td>
<td>38.6</td>
<td>53.9</td>
</tr>
<tr>
<td>Phoneme 3-grams</td>
<td>28.1</td>
<td>47.3</td>
</tr>
</tbody>
</table>

Table 3: Classification of Spoken Documents reported as F-measure (human scores about 70 on this task)

The spoken document classification results demonstrate the independence of classifier performance from speech...
recognition accuracy in a striking way. Although the syllable error rate in the recognizer output is about 70%, the classifier performance did not deteriorate to anywhere near that level. The LSVM out-performs the Naive Bayes in its ability to compensate for speech recognition error in the input space. This effect again reflects the ability of SVMs to generalize effectively as the number of dimensions of the input space increases.

It is interesting that the Naive Bayes classifier actually does better than the LSVM in case of syllable 1-grams on the small topic Science. We feel that this is a case where feature frequency-based representation of text used by the Naive Bayes Classifier simply provided a better generalization than the more specialized feature weighting/selection we used for the SVM.

In a final set of experiments we explore the possibility of training the classifiers on the text transcripts of the Kalenderblatt database and using them to classify spoken audio documents. We tested this scheme on the topic Politics; the results are presented in Table 4.

<table>
<thead>
<tr>
<th>Topic: Politics</th>
<th>Classification of Audio Documents with classifiers trained on Text Documents</th>
<th>Naive Bayes</th>
<th>LSVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>syllable 1-grams</td>
<td>48.0</td>
<td>38.5</td>
<td></td>
</tr>
<tr>
<td>syllable 2-grams</td>
<td>42.1</td>
<td>61.1</td>
<td></td>
</tr>
<tr>
<td>syllable 3-grams</td>
<td>42.0</td>
<td>63.4</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Classification of Spoken Documents classified with classifiers trained on Text Documents, reported as F-measure.

Although both classifiers prove able to use generalizations about text to classify recognizer output, the LSVM really comes into its own. Starting with the syllable bi-grams the LSVM is actually able to achieve the same performance it attained when the spoken document classification system was trained with recognizer output.

6. CONCLUSIONS AND OUTLOOK

In this paper we have demonstrated that both Naive Bayes and SVMs are effective text document classifiers and that both benefit from the use of sub-word units, especially in cases where training data is limited. We have shown that the advantages of using sub-word units as input features extend to the domain of spoken document classification, where both Naive Bayes and SVM classifiers suffer only modest performance losses in the face of the 70% syllable error rate generated by a completely domain-independent speech recognition system using a small syllable bi-gram language model.

We have presented evidence that expanding the classifier input space with combinations of sub-word units also improves the performance a spoken document classification system. Our results show that the real benefit of such input space expansion is enjoyed by SVMs, a corroboration of their well-known capabilities of SVMs.

Future work will involve experiments with further features for input documents in order to ensure that we have achieved the optimum mix for the LSVM. We will also try to improve our baseline classifier by trying out some of the weighting/filtering schemes that we use in the SVM input space on the Naive Bayes classifier. Finally we would like to further develop the syllable-based language model used for the speech recognizer. We hope by optimizing the syllable vocabulary with respect to the spoken document classifier we will achieve yet better classification performance.

We feel that there is a large range of research directions that may not decrease speech recognition error rate directly, but which have the potential to improve spoken document classification, especially in combination with the generalization capabilities of SVMs.

7. References