Named Entity Recognition of Spoken Documents using Subword Units

Gerhard Paaß Anja Pilz
Jochen Schwenninger
Fraunhofer Institute Intelligent Analysis and Information Systems (IAIS)
53754 St. Augustin, Germany
{Gerhard.Paass Anja.Pilz Jochen.Schwenninger}@iais.fraunhofer.de

Abstract

The output of a speech recognition system is a stream of text features that is overlayed by noise resulting from errors in the system's statistical classification of the audio input. Conditional Random Fields (CRFs), which have already proven themselves to be efficient, high-performance Named Entity Recognizers (NERs) for named entities from text, offer the promise to compensate part of these errors. In this paper we use CRFs to extract named entities from spoken audio documents. We consider a real-world audio information extraction scenario under which CRFs are trained to recognize named entities in unedited radio audio documents that have been converted into a stream of text features by a speech recognition system. The automatic speech recognition system (ASR) is able to produce word transcriptions as well as syllables. It uses general speaker-independent acoustic models and a domain-independent statistical language model, insuring that recognizer performance is not specific to the experimental domain. Using an additional syllable model increases the generality of the spoken document classification system, giving it the flexibility to handle words that are not present in the vocabulary. In this paper we apply for the first time CRFs to different features produced by German ASR. The experiments confirm that using transcribed syllables together with words can compensate for part of the NER errors caused by ASR transcription.

1. Introduction

Research and development in the area of spoken document processing is driven by the vision of achieving semantic annotation, indexing and retrieval functionality on audio data, of the same scope and scale as for text data. A spoken document processing system uses some sort of speech recognition system (ASR) to convert speech to text features, typically, but not exclusively words. The semantic interpretation of speech documents proves to be a qualitatively different problem than the interpretation of text documents. A risk remains that tried-and-true algorithms for text named entity recognition will have their performance compromised by the noise that the speech recognition system introduces into the textual representation of the spoken document in the form of misrecognized, inserted and deleted text features. The presence of this noise obscures the occurrence of text features (word stems, keywords), on which a standard named entity recognition system relies to make its classification decision.

Conditional random Fields (CRFs) have proven to be fast and effective NER methods for text documents. Since CRFs also have the advantage of being able to effectively exploit otherwise indiscernible regularities in high dimensional data, they represent an obvious candidate for spoken document NER, offering the potential to effectively circumvent the error-prone decisions characteristic of speech to text conversion. It is possible to use additional information generated during the speech recognition to make up for the errors in the final decision. A promising candidate for this approach is to produce a sub-word level transcription based on different models.

In this paper we present the results of experiments which applied NER to a real-life scenario, annotating video documents from different German TV broadcasters. The next section provides an introduction to CRFs in the context of their application to text NER. Section 3 introduces the large vocabulary continuous speech recognition and discusses the HMM-based recognition system that was used in the experiments presented here. Section 4 describes previous work in spoken document NER and makes a case for our use of CRFs. In section 5 our training and test collection is introduced and our experimental setup is described. Section 6 presents results and the final section concludes.

978-0-7695-3800-6/09 $26.00 © 2009 IEEE
DOI 10.1109/ICSC.2009.78

2. CRFs for Text Document NER

To process documents it is essential to recognize information units like names, including person, organization and location names, and numeric expressions including time, date, money, etc. Identifying references to these entities in text is called Named Entity Recognition. While early systems were making use of handcrafted rule-based algorithms, modern systems most often resort to machine learning techniques, especially supervised learning techniques like Hidden Markov Models, Decision Trees, Maximum Entropy Models, Support Vector Machines, and Conditional Random Fields (CRFs) [13].

In this paper we use the CRF for named entity recognition. We consider a task which is characterized by sequences $\mathbf{x} = (x_1, \ldots, x_T)$ of inputs. In language modeling an input x_t usually contains different features of the t-th words of a document $\mathbf{x}$. To each word x_t corresponds a state y_t which has values in a set of labels $\mathcal{Y} = \{\gamma_1, \ldots, \gamma_m\}$, e.g. indicators of different named entity classes. It is the task to predict the state sequence $\mathbf{y} = (y_1, \ldots, y_T)$.

Conditional Random fields (CRFs) [5, 17] are conditional probability distributions that factorize according to an undirected model.

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{c \in C} \phi_c(\mathbf{y}_c, \mathbf{x}) \tag{1}$$

It is assumed that there is a dependency between the states at different sequence positions. The $c \in C$ are subsets $c \subseteq \{1, \ldots, T\}$ of time instances where the states may have direct interactions. $\mathbf{y}_c = (y_t)_{t \in c}$ is a subvector of states $\mathbf{y} = (y_1, \ldots, y_T)$ with indices $t \in c \subseteq \{1, \ldots, T\}$. The $\phi_c(\mathbf{y}_c, \mathbf{x})$ are real-valued functions of these variables and $Z(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{Y}^T} \prod_{c \in C} \phi_c(\mathbf{y}_c, \mathbf{x})$ is a normalizing factor for the sequence $\mathbf{x}$.

Many applications use a linear-chain CRF, in which the sequential order of inputs is taken into account and used for a first-order Markov assumption on the dependence structure. In this case the subvectors are pairs $\mathbf{y}_c = (y_t, y_{t-1})$ and yield the *feature functions* $f_k(y_t, y_{t-1}, \mathbf{x})$ with an associated parameter λ_k.

We assume that the corresponding feature functions do not depend on the value of t, which allows weight sharing between all these components. On the other hand they may take into account the complete input vector $\mathbf{x}$. In the simplest case feature functions take the value 1 for a subset of the values $(y_t, y_{t-1}, \mathbf{x})$ and 0 otherwise.

Note that there may be different feature functions for the same variables $y_t, y_{t-1}, \mathbf{x}$. This also covers the special case of functions $g_k(y_t, \mathbf{x})$ containing only one state and can easily be extended to higher order Markov chains. Hence we arrive at

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \left(\sum_t \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \right) \tag{2}$$

If let $p(\mathbf{x})$ be the unknown marginal distribution of $\mathbf{x}$. As $p(\mathbf{y}|\mathbf{x}) = p(\mathbf{y}, \mathbf{x})/p(\mathbf{x})$ we may write the joint distribution by (2) as

$$p(\mathbf{y}, \mathbf{x}) = r(\mathbf{x}) \exp \left(\sum_t \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \right) \tag{3}$$

where $r(\mathbf{x}) = p(\mathbf{x})/Z(\mathbf{x})$ is a term depending only on $\mathbf{x}$. Then the conditional distribution is

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}) &= \frac{p(\mathbf{y}, \mathbf{x})}{\sum_{\mathbf{y} \in \mathcal{Y}^T} p(\mathbf{y}, \mathbf{x})} \\ &= \frac{1}{\tilde{Z}(\mathbf{x})} \exp \left(\sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \right) \end{aligned} \tag{4}$$

where

$$\tilde{Z}(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{Y}^T} \exp \left(\sum_t \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}) \right) \tag{5}$$

is an input-specific normalization function and $\mathcal{Y}^T$ is the set of all sequences of the form $\mathbf{y} = (y_1, \ldots, y_T)$.

Now assume we have N i.i.d. observations $(\mathbf{x}^{(n)}, \mathbf{y}^{(n)})$, $n = 1, \ldots, N$ generated according to (3). As a regularizer we introduce a penalty for large λ-values, e.g. a prior $p(\lambda) = \exp(-\sum_{k=1}^{K} \lambda_k^2 / 2\sigma^2)$ proportional to a Gaussian. Then the conditional log-likelihood $L(\lambda)$ for the vector λ of all parameters is

$$\begin{aligned} L(\lambda) &= \sum_{n=1}^{N} \log p(\mathbf{y}^{(n)}|\mathbf{x}^{(n)}, \lambda) \\ &= \sum_{n=1}^{N} \sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(y_t^{(n)}, y_{t-1}^{(n)}, \mathbf{x}^{(n)}) \\ &\quad - \sum_{n=1}^{N} \log \sum_{y \in \mathcal{Y}^T} \exp \left(\sum_t \sum_{k=1}^{K} \lambda_k f_k(y_t^{(n)}, y_{t-1}^{(n)}, \mathbf{x}^{(n)}) \right) \\ &\quad - \sum_{k=1}^{K} \frac{\lambda_k^2}{2\sigma^2} \end{aligned} \tag{6}$$

The derivative of the log-likelihood may be evaluated and used by limited memory quasi-Newton maximizers like L-BFGS [7] to find the optimal parameters.

3. Continuous speech recognition

3.1 Speech recognition system

Both acoustic and language models of the ASR have been trained using the HTK Hidden Markov Toolkit[1]. For the acoustic modeling, a set of over 80,000 utterances was extracted manually from a large database of video recordings, yielding over 80 hours of pure speech data for training. Each utterance was transcribed manually on the word level. We use a set of 50 German phonemes which are modeled by context-dependent crossword triphone HMMs. Each HMM consists of three states, connected by forward and self-transitions. At each state the probability that current state emitted the given feature vector is modeled by a probability density function composed of a mixture of Gaussians. We chose the number of of mixture components per state to 32 in order to model the large inter-speaker variability occurring in the training set. This high number of mixtures allows us to use a gender-independent model. A phonetic decision tree was applied to cluster similar triphones and thereby reduce the number of required parameters. The parameters of the remaining 18,000 triphone models were selected using Baum-Welch reestimation with the Maximum Likelihood criterion. The resulting acoustic models are used for both word and syllable decoding.

As spectral features, we extract the first 12 Mel-frequency Cepstral Coefficients (MFCC) together with the energy of the signal, and add the first and second derivatives to the vector.

For the word language models, we extracted text statistics from a large set of textual data. The corpus consists of German newswire data from 2000 to 2006, newspaper articles from 2006 to 2009 as well as the complete manual transcriptions from the acoustic training set, adding up to over 300 million words. The 200,000 most frequent words occurring in the language model training corpus were selected as the vocabulary of the recognizer, and trigram language models were trained with Good-Turing discounting as the smoothing strategy.

The large word count is necessary because of the compounding strategy inherent in the German language [10]. Grapheme-to-phoneme conversion for the word pronunciation lexicon was carried out using the transcription module of the Bonn Open Source Synthesis System (BOSSII) developed by the Institut für Kommunikationsforschung und Phonetik of Bonn University [4].

[1] http://htk.eng.cam.ac.uk/

The syllable language model was trained on the same training data set as the word language models, but the word sequences of the training text were broken down to syllable sequences first, again using BOSSII, yielding a finite set of 10,000 distinct syllables. A 4-gram syllable language model was trained, using the same setup as for the word model.

The actual decoding is carried out using the Julius[2] speech recognition engine provided by ISTC [6]. More details about the ASR setup and its use in a spoken document retrieval system are described in [15].

3.2. Errors of speech recognition systems

There are several sources for the errors made by the ASR and understanding them is essential to understanding the interface between the ASR and the CRF. First of all there are errors due to unexpected acoustics. If a phoneme in the input audio is pronounced in a way not significantly represented in the audio data used to train the ASR, the ASR probably produce a misclassification. Other unexpected acoustics such as swallowed syllables, mid-word pauses, or non-speech noises such as coughing or throat clearing are also a source of acoustic error [11]. Class-conditional probabilities are calculated on the basis of larger environments, however, and often if neighboring phonemes are pronounced consistently with well represented pronunciations, errors will be avoided. Background noises such as traffic or music, on the other hand, extend past the immediate context and can seriously affect system performance.

A second source of error relevant to the document classification task is error due to the language model. The worst case is when the word pronounced in the audio is completely missing from the language model inventory, a so called OOV (out of vocabulary) error. To reduce the impact of this dictionary limitation, a subword transcription, where OOV errors have extremely low probabilities, can be used.

The error rates produced by the ASR system on the test data are presented in section 6.

3.3. NER for speech using words and syllables

Recognition errors described above can hamper subsequent analysis, e.g. named entity recognition. A possible remedy is to use not only the recognized words but additional output from the ASR. There are several additional clues which might be used:

[2] http://julius.sourceforge.jp/

- Instead of only using the best path in the word lattice we may use the n best paths weighted by probabilities.
- Use the entire lattice of possible paths in the word lattice weighted with probabilities.
- Use one or more phonemes paths predicted by the ASR, up to the complete lattice.
- Use one or more syllables paths predicted by the ASR, up to the complete lattice.

In this paper we use the most probable syllables in conjunction with the words identified by the ASR. Note that the most probable syllables of a path in the lattice my differ from the sub-word representation of the most probable word. This is especially true if all possible decoding alternatives have low probabilities and errors might occur. Thus both items provide alternative description of the speech. A CRF including both features consequently has the potential to correct errors present in the best sequence of words. In addition sub-word text features allow the recognizer a degree of independence from the vocabulary of the topic domain over which it was trained and are far less prone to OOV errors.

As the optimal words detected by the ASR and the most probable syllables are somewhat independent they have to be aligned to be used as feature for NER. Here we use the sequence of words in a sentence as units characterized by $\mathbf{x} = (x_1, \ldots, x_T)$. An input x_t contains different features of the t-th words, e.g. the word returned by ASR, its POS-tag, as well as characteristics like prefixes, capitalization patterns and other form descriptors. We add the syllables detected by ASR as features of the word. The ASR returns the time code for the beginning and the ending of each word as well as for each syllable. We consequently assign the syllables to the word sequence according to overlap of the respective time frames. An example showing words, syllables and corresponding NER labels is presented in table 1. Note the incorrect lower case on emphschwarzen, since this is usually not a noun but an adjective.

To each word x_t corresponds a state y_t which has values in a set of labels $\mathcal{Y} = \{\gamma_1, \ldots, \gamma_m\}$. In our case these are the indicators of the different named entity classes PER (=person), ORG (=organization), and LOC (=location). Non-entity words get the label O (=other). To be able to discriminate two adjacent multiword named entities of the same type we use the labels I-PER, I-ORG, and I-LOC to mark the second, third, etc. word of a named entity. It is the task to predict the state sequence $\mathbf{y} = (y_1, \ldots, y_T)$.

Word	**Syllables**	**NER - state**
Martin	m_a6_ t_i:_n_	PER
Luther	l_U_ t_6:_	I-PER
King	k_I_N_	I-PER
der	d_e:_6:_	O
Kampf	k_a_m_p_f_	O
der	d_e:_6:_	O
schwarzen	S_v_a6_ t_s_@_n_	O
um	Q_U_m_	O
Freiheit	f_r_aI_ h_aI_t_	O

Table 1. Text transcribed by ASR (left) with corresponding syllables (middle) and the named entity classes of words (right).

4. Related work on spoken document NER

Most previous studies of the NER of speech data used generative models such as hidden Markov models (HMMs) [12, 2]. On the other hand, in text-based NER, better results are obtained using discriminative schemes such as maximum entropy (ME) models [1], support vector machines (SVMs) [3], and conditional random fields (CRFs) [9]. [18] applied a text-level ME-based NER to ASR results in realistic large-vocabulary spoken Chinese. These models have an advantage in utilizing various features, such as part-of-speech information, character types, and surrounding words, which may be overlapped, while overlapping features are hard to use in HMM-based models. They introduce a method of using n-best hypotheses that yields a small but nevertheless useful improvement in NER accuracy. To deal with ASR error problems in NER, [14] proposed an HMM-based NER method that explicitly models ASR errors using ASR confidence and rejects erroneous word hypotheses in the ASR results. Such rejection is especially effective when ASR accuracy is relatively low because many misrecognized words may be extracted as NEs, which would decrease precision.

[16] extended this approach to discriminative models and propose an NER method that deals with ASR errors as features. They use NE-labeled ASR results for training to incorporate the features into the NER model as well as the corresponding transcriptions with NE labels. In testing, ASR errors are identified by ASR confidence scores and are used for the NER. In experiments using SVM-based NER and speech data from Japanese newspaper articles, the proposed method increased the NER F-measure, especially in precision, compared to simply applying text-based NER to the ASR results.

ASR Unit	WER	Ins	Del	Sub
Word	31.3	2.7	7.3	21.4
Syllable	27.3	2.6	3.1	21.6

Table 2. ASR performance on test corpus

5. Experiments

In our experiments we employed two different German corpora. First we used different TV broadcasts from the news and magazines genre comprising a total of 4 hours. They contain speech segments of a large number of different speakers in diverse acoustic environments. This diversity makes the corpus an appealing resource since it represents a real world task. The corpus was manually transcribed in order to have a baseline for assessing the NER accuracy on error-free transcriptions as well as for analyzing ASR performance. Additionally it was processed with our large vocabulary ASR, generating words and syllables. The material was manually labeled with the named entities person, location, organization and misc.

As it proved to be impossible to annotate enough training material we used additional synthetic data to train the NER module. We chose the CoNLL 2003 corpus [3] containing about 20,000 sentences from German newswire articles. They are annotated with the named entities person, location, organization and misc. For our experiments we took the text as predicted result of the ASR, thus ignoring ASR errors for training. To generate the syllable representation required for training the CRFs we employed the transcription module of the Bonn Open Source Synthesis System (BOSSII) [4] described above.

During identification of named entities different types of errors may occur. We may annotate only part of a named entity. We may assign the wrong entity type to an entity, etc. We are strict and require that a named entity is completely recovered and annotated with the correct type. All other cases are counted as errors.

6. Results

The performance of the ASR is displayed in table 2. Taking the diversity of the test corpus with its high amount of spontaneous speech into account, the word error rates (WER) is comparable to the performance of other systems [10]. The better performance of the sub-word based recognition can be explained by out-of-vocabulary (OOV) effects in the word recognition, especially since both use the same acoustic model.

[3] available online at http://www.cnts.ua.ac.be/conll2003/ner/

Named Entities	Prec %	Rec %	F %
Genuine Text			
Persons	89.7	67.8	77.2
Locations	62.5	81.5	70.8
Organizations	83.9	40.3	54.4
Genuine Text & Syllables			
Persons	91.0	68.1	77.9
Locations	64.1	81.8	71.9
Organizations	86.2	41.1	55.7
ASR Text			
Persons	65.2	36.5	46.8
Locations	53.2	66.6	59.1
Organizations	79.2	34.2	47.8
ASR Text & Syllables			
Persons	66.1	36.7	47.2
Locations	54.6	66.9	60.2
Organizations	79.3	34.5	48.1

Table 3. Overview over NER results using different features

For evaluating the NER, precision, recall and the f-measure are defined in the usual way [8]. The broadcast corpus was used for testing. It was manually annotated with 502 persons, 97 locations, and 350 organizations. Using this as ground truth, we arrive at the following results shown in table 3.

The results for genuine test data (reference transcriptions), e.g. 77% f-value for persons are ok, but much worse than on the CoNLL test data (well above 80%). This difference shows clearly the effect of a mismatch between training and testing corpus. This mismatch is not only affecting the topics and sentence structure of the texts: Since ASR errors were not present in the training data, the CRF adapts to a perfect mapping between word and subword representation of the text.

Adding syllables as features to the genuine text helps and improves the results consistently by about 1 %.

For the ASR transcripts we observe a profound drop in performance to about 50% f-value, due to the errors made during ASR. The inclusion of syllables again improves the results slightly up to 1 %. However the significance of the improvement could not be checked.

7. Conclusion

In this paper we applied a Named Entity Recognition module to transcripts of continuous German speech. In addition to the automatically generated word transcription we used syllables as features for the NER model. It turned out that the results show a slight improvement of performance if syllables are used. This improvement is caused mainly by increasing precision, leading to a more convenient behavior for most real world tasks.

Further investigation is needed to clarify whether syllables can be used for correcting OOV-related errors in word-level ASR and subsequent analysis like extraction of named entities.

This is work in progress. In the future we will improve the CRFs by explicitly modeling the sequence of syllables. In addition we want to use confidence information generated during the ASR to lower the influence of uncertain tokens. By including real ASR output into the training data we try to model typical errors produced during transcription and arrive at a better match between training and test corpora.

8. Acknowledgement

The work presented here was funded by the German Federal Ministry of Economy and Technology (BMWi) under the THESEUS project. We would like to thank Friederike von Rantzau for annotating the test data.

References

[1] H. L. Chieu and H. T. Ng. Named entity recognition with a maximum entropy approach. In *Proc. CoNLL*, pages 160–163, 2003.

[2] B. Favre, F. Bechet, and P. Nocera. Robust named entity extraction from large spoken archives. In *Proc. HLT-EMNLP*, pages 491–498, 2005.

[3] H. Isozaki and H. Kazawa. Efficient support vector classifiers for named entity recognition. In *Proc. COLING*, pages 390–396, 2002.

[4] E. Klabbers, K. Stöber, R. Veldhuis, P. Wagner, and S. Breuer. Speech synthesis development made easy: The bonn open synthesis system,. In *Proc. EUROSPEECH 2001*, 2001.

[5] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289, 2001.

[6] A. Lee, T. Kawahara, and K. Shikano. Julius–an open source real-time large vocabulary recognition engine. In *Seventh European Conference on Speech Communication and Technology*. ISCA, 2001.

[7] D. C. Liu and J. Nocedal. On the limited memory method for large scale optimization. *Mathematical Programming B*, 45(3):503–528, 1989.

[8] C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.

[9] A. McCallum and W. Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proc. CoNLL*, pages 188–191, 2003.

[10] K. McTait and M. Adda-Decker. The 300k LIMSI German broadcast news transcription system. In *Eighth European Conference on Speech Communication and Technology*. ISCA, 2003.

[11] T. Mertens, D. Schneider, and J. Köhler. Merging search spaces for subword spooken term detection. In *Interspeech 2009*, 2009.

[12] D. Miller, R. Schwartz, R. Weischedel, and R. Stone. Named entity extraction from broadcast news. In *Proceedings of the DARPA Broadcast News Workshop*, pages 37–40, 1999.

[13] D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3–26, 2007.

[14] D. D. Palmer and M. Ostendorf. Improving information extraction by modeling errors in speech recognizer output. In *Proc. HLT*, pages 156–160, 2001.

[15] D. Schneider, J. Schon, and S. Eickeler. Towards large scale vocabulary independent spoken term detection: Advances in the fraunhofer iais audiomining system. In *Proc. Searching Spontaneous Conversational Speech (SSCS) SIGIR 2008 workshop*, 2008.

[16] K. Sudoh, H. Tsukada, and H. Isozaki. Incorporating speech recognition confidence into discriminative named entity recognition of speech data. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 617–624, 2006.

[17] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT-Press, 2007.

[18] L. Zhai, P. Fung, R. Schwartz, M. Carpuat, and D. Wu. Using n-best lists for named entity recognition from chinese speech. In *Proc. HLT-NAACL*, pages 37–40, 2004.