Automatic Punctuation Annotation in Czech Broadcast News Speech

Jáchym Kolář, Jan Švec, Josef Psutka

University of West Bohemia in Pilsen, Department of Cybernetics,
Univerzitní 8, Plzeň, 306 14, Czech Republic

{*jachym, honzas, psutka*}*@kky.zcu.cz*

Abstract

This paper reports our initial experiments with automatic punctuation annotation from speech. We have focused on Czech broadcast news speech. The task can be defined as a classification of each inter-word boundary into one of target classes. We considered comma, sentence boundary and "no punctuation" as the target classes.

We employed two statistical models – prosodic model and language model. The prosodic model expresses relationships between prosodic quantities (such as pitch, speaking rate or loudness) and punctuation marks. We tested two implementations of this model – decision tree and multi-layer perceptron. Hidden-event N-gram models were employed for language modeling. Instead of using an ordinary word-based model, we replaced infrequent word forms by their morphological tags and trained a mixed model.

Scores from both models can be combined. The model combining language model with the decision tree yielded superior results. Testing on true words we achieved classification accuracy 95.2% and F-measure 78.2%.

1. Introduction

The automatic extraction of information from audio recordings is an important task today. As the automatic speech recognition (ASR) made a big step ahead in recent years, large volumes of audio data can now be transcribed automatically. However, these automatic transcripts are difficult to process, both for man and computer, because of missing sentence boundaries, punctuation marks and casing. The goal of our work is to improve the readability of those automatic transcripts and/or to arrange them into a form more fitting for the consequent automatic processing.

The correct determination of sentence boundaries is a crucial problem from the point of view of natural language processing (NLP), because most of NLP applications (such as information retrieval, text summarization, parsing or machine translation) require input divided into sentences.

Sentence boundaries are typically marked by full-stops and question marks in manual speech transcripts. We joined these two types of punctuation into a one class called sentence end (<sen>) due to the lack of questions in the used speech corpus. Besides the sentence boundaries, we have also focused on the automatic insertion of commas. Thus, our overall task can be described by the term "automatic punctuation annotation". This task can be defined as a classification of each inter-word boundary into one of target classes (i.e. comma, sentence boundary and "no punctuation").

The particular problems of automatic punctuation annotation differ according to the mode of analysed speech. In this paper we concentrate on the Czech broadcast news speech which is mainly read. When processing spontaneous speech some additional problems arise (e.g. speech disfluencies).

There are two sources of information that can be used to solve the task – recognized words (*what* a speaker said) and prosody (*how* the speaker said it). Thus, we can employ two statistical models – *language model* and *prosodic model*. The language model aims to provide probabilities that a punctuation mark occurs within a given word context, whereas the prosodic model expresses relationships between prosodic quantities (such as pitch, speaking rate or loudness) and punctuation marks.

In recent years, several approaches to automatic punctuation from speech exploiting lexical and prosodic features have been proposed. We have mainly benefited from insights provided by the work of Shriberg, Stolcke et al. They proposed an approach based on the "direct modeling of prosody" by decision trees and hidden-event N-gram language models [1, 2, 3]. Also Kim and Woodland adopted their approach for developing a combined punctuation generation and speech recognition system [4]. Christensen, Gotoh and Renals presented a statistical finite-state model that combined prosodic, linguistic and punctuation class features [5]. Huang and Zweig developed a maximum entropy based method for annotating spontaneous conversational speech with punctuation. They used features based on recognized words and pause lengths [6].

However, these papers deal with English speech. We have focused on our Czech language. It has, same as other Slavic languages, a highly inflectional and deriva-

tional nature, which causes additional problems with language modeling (e.g. much larger vocabulary, more difficult part-of-speech tagging etc.). Czech has also a relatively free word order which degrades the performance of N-gram language models.

The use of punctuation in Czech is similar to English, but rules for writing commas are more strict in Czech. Commas separate:

- all co-ordinate constituents unless they are connected by copulative conjunctions *a, i, nebo, či, ani* (lit. and, and, or, or, nor)
- subordinate clauses from main clauses
- all independent constituents that are inserted into a sentence (parentheses, complements, vocatives, explanatories etc.)

The rest of this paper is organized as follows. Section 2 briefly introduces the speech corpus. Section 3 presents used evaluation metrics. Sections 4 and 5 describe the prosodic and the language model respectively, Section 6 describes their combination. In Section 7, we report experimental results and finally in Section 8, we present our conclusions and future work.

2. Speech data

All experiments were performed on the Czech Broadcast News Speech Corpus which is currently available from Linguistic Data Consortium (LDC) [7]. The corpus consists of news broadcasted on 3 TV channels and 4 radio stations during the period February 1, 2000 through April 22, 2000. It contains over 50 hours of audio data which yield about 26 hours of pure transcribed speech. The broadcast news does not contain weather forecasts, sports and traffic announcements. The signal is sampled at 22kHz.

284 distinct speakers (188 males and 96 females) appear in the recordings. The transcripts contain 260k tokens (including punctuation), 16.5k sentences and 6k turns. More details about the corpus annotation are given in [8].

For our experiments, we randomly split the data into three pools (training, development and test set) that do not share any speaker. The training set contains 207k tokens spoken by 175 speakers, the development set 29k tokens by 60 speakers, and the test set 24k tokens by 49 speakers.

3. Evaluation metrics

A choice of an appropriate metric for the performance evaluation of an automatic punctuation system is difficult. There is no obviously appropriate evaluation metric. We used the overall classification accuracy (Acc) and the precision (P), recall (R) and F-measure percentages. Acc is defined as

$$Acc = \frac{C}{N_W} \tag{1}$$

where C denotes the number of correctly punctuated words and N_W denotes the total number of words. The problem is that a strong majority of words is not followed by a punctuation (in our test set it is 86.7%). Hence, one can get relatively high Acc simply by inserting no punctuation anywhere, so that the numbers can be quite misleading. In order to avoid this misinterpretation, we also report precision and recall measures well-known from information retrieval systems. The precision and recall are defined as

$$P = \frac{C}{C + FA} \tag{2}$$

$$R = \frac{C}{C + M} \tag{3}$$

where *FA* denotes number of false alarms and M denotes number of misses. To express the system performance by a single number, it is possible to use a harmonic mean of P and R

$$F = \frac{2PR}{P + R} \tag{4}$$

called F-measure. Note that the F-measure can also be misleading since it deweights errors of insertion and deletion in comparison to errors of substitution by a factor of two.

Besides the above stated metrics, we also report results measured by their modifications. The metrics denoted by a single-quote (i.e. Acc', P', R', F') are counted in such a way that a half score is given when a punctuation is located right but recognized as a wrong punctuation symbol.

4. Prosodic model

4.1. Prosodic features

The modeling of prosody is not an easy task for a number of reasons. First, prosody is influenced by an individual style and mood of the speaker. Further, prosodic (suprasegmental) features are partially affected by the segmental content of the utterance. Next, there is a trading relation between prosodic means. A weaker use of one prosodic mean can be compensated by a stronger use of another. Thus, some normalization and smoothing techniques must be applied to produce meaningful and speaker-independent features.

For deriving prosodic features we adopted direct modeling strategy of Shriberg and Stolcke [3], so that no hand-labeling of prosody (such as ToBI) was necessary. Instead, the features were extracted directly from the automatically aligned speech signal.

For features extraction, we used our speech and prosody database which is described in [9]. The database is designed in such a way that we can quickly and easily compute values of any desired set of prosodic features.

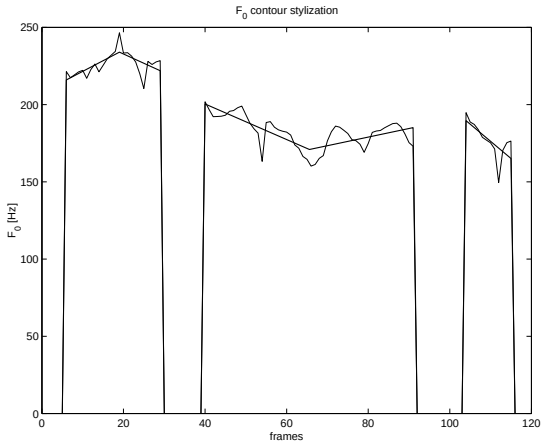

Figure 1: Raw and stylized F_0 contour

The features were utilized at a word level. For each inter-word boundary, we computed features from two words before and one word after the boundary. The positions of the boundaries were determined by a forced alignment.

The used prosodic features were related to pitch (F_0), phoneme durations, pause lengths and energy. The strategy was to create a large set of potentially useful features and then to reduce it according to a classification performance of individual features. We also added one non-prosodic feature which indicates occurrence of the end of a turn after the current word. This feature is very important since we cannot reliably measure pause durations after the last words in turns.

4.1.1. F_0 features

For F_0 tracking in voiced regions of speech, we used the RAPT algorithm [10]. The measured F_0 values must be preprocessed before extracting F_0 features. At first, we had to remove halved and doubled F_0 values, because the presence of octave errors is a typical problem of pitch trackers. For this purpose, we used the lognormal tied mixture model (LTM) described in [11]. Remaining accurate values were then filtered by a median filter.

It is also necessary to deal with the phenomenon of microintonation. The tracked, median filtered, and halved/doubled values removed pitch contour still contains a lot of local fluctuations. These fluctuations are involuntary on the speaker's part and mostly related to the physiology of speech. A common way to remove the microintonation is to stylize the pitch contour by a piecewise linear (PWL) function. The line fits better interpret pitch movements intended by the speaker [12]. An example of the F_0 contour stylization is shown in Figure 1.

Subsequently, we extracted the features from the preprocessed F_0 contour. Here we have to note that we assume that during testing, speaker tracking information is available as well as long-time means and variances of speakers' F_0. These statistics are used for the features normalization, since unnormalized values are giving no sense of relative positions in the speaker's pitch range. When assuming that these statistics are not available, different features must be used [13]. We used following F_0-derived features:

- maximum, minimum and mean
- first and last value
- first and last PWL slope
- ratio and difference between the last value in the current word and the first value in the following word
- ratio and difference between the last PWL slope in the current word and the first PWL slope in the following word
- slope of linear regression from all values in the current word

4.1.2. Phoneme duration features

This group of features describes the phenomenon of preboundary lengthening – speakers usually tend to slow down their speech toward the ends of utterance units. We have focused only on the duration of vowels, because it is known that vowels influence the overall speaking rate more significantly than consonants.

The vowel durations were obtained by the forced alignment of speech data. We also gained long-time duration statistics (mean and variance) for each particular vowel from those alignments. The statistics were speaker-independent and were used for the normalization.

For classification, we used following duration features (all features were normalized):

- average duration of vowels
- duration of the first and last vowel
- duration of the longest and shortest vowel

4.1.3. Pause features

The strongest indicators of punctuation in speech are pauses. Pause features can be extracted very easily and robustly. We have simply used raw duration of the pause after a word of interest. We also exploited a feature describing a type of the pause. The considered pause types were silent (SIL), filled with hesitation (HES) or filled with audible breath (LB).

4.1.4. Energy features

The next group of features relates to the loudness of speech. Speakers usually tend to quieten their voice toward the end of the utterance unit. We used the maximum, minimum and mean of frame level RMS energy values.

```
pause.after < 255 ms:
|   turn.end = false:
|   |   pause.after < 95 ms:
|   |   |   vow.max_dur.snorm < 1.17:  0.71 0.01 0.28 <none>
|   |   |   vow.max_dur.snorm >= 1.17:  0.45 0.03 0.52 <com>
|   |   pause.after >= 95 ms:
|   |   |   f0.last.end < 1.14:  0.04 0.52 0.44 <sen>
|   |   |   f0.last.end >= 1.14:  0.08 0.13 0.79 <com>
|   turn.end = true:  0.00 1.00 0.00 <sen>
pause.after >= 255 ms:
|   f0.last.end < 1.17:  0.01 0.93 0.06 <per>
|   f0.last.end >= 1.17:
|   |   slope.last < -0.25:
|   |   |   f0.rat.last_first < 0.94:  0.01 0.84 0.15 <sen>
|   |   |   f0.rat.last_first >= 0.94:  0.06 0.51 0.43 <sen>
|   |   slope.last >= -0.25:
|   |   |   pause.type in LB,HES :  0.04 0.18 0.78 <com>
|   |   |   pause.type in SIL :  0.02 0.69 0.29 <sen>
```

Figure 2: Top 4 levels of the CART for punctuation detection using prosodic cues (the preliminary probabilities listed in the 4th level correspond to "no punctuation", <sen> and <com> respectively)

4.2. Prosodic classification by decision tree

We tested two methods of prosodic classification – multi-layer perceptron (MLP) and CART-style (Classification and Regression Trees) decision tree. Next subsection describes the first possibility, in this subsection we discuss the latter. The use of CART provides some advantages. First, it allows to work with features that can have undefined values. Second, we can easily combine continuous and categorical features. And third, the resulting trees are human-readable and can be easily interpreted. Interested readers may consult [14] for more details on the training and use of the CART trees.

When training the CART for tasks with highly uneven class sizes (such as automatic punctuation annotation), it is often useful to downsample the data on equal class sizes. It allows the classifier to model inherent properties of smaller classes in more detail.

The maximum depth of the trained CART was 13. The top four levels of the tree are listed in Figure 2.

4.3. Prosodic classification by multi-layer perceptron

We have tried a feed-forward multi-layer perceptron (MLP) as another prosodic classifier. Our four-layer MLP had a topology 117-15-30-15-3. The neurons in the first hidden layer had a linear activation function $g(a) = a$. In the next two hidden layers, we used a function $g(a) = \tanh(a)$, and the neurons in the output layer had a soft-max activation function

$$y_j(a_j) = \frac{\exp(a_j)}{\sum_{k=1}^{K} \exp(a_k)} \tag{5}$$

where j is the index of an output layer neuron and K is the number of output neurons (in our case $K = 3$). As an error function we used a cross-entropy function

$$E = -\sum_{n} \sum_{k=1}^{K} t_k^{(n)} \cdot \log\left(\frac{y_k^{(n)}}{t_k^{(n)}}\right) \tag{6}$$

where $t_k^{(n)}$ is the requested network output and $y_k^{(n)}$ is the real network output value for the (n)-th training pattern and the k-th output neuron.

The principal components analysis (PCA) method was used for the setting up of the first hidden layer. The goal of the PCA is to find a matrix which projects input vector onto a lower dimensional vector space which is spanned by the first n principal components. During experiments, we found $n = 15$ to be a convenient value. After running the PCA, the resulting projection matrix was set into the first linear layer. This layer was then locked-up and remaining layers were trained using the scaled conjugate gradients (SCG) method. Afterwards, the first layer was unlocked and the whole MLP was retrained. In order to avoid overfitting of the training data, the early-stopping method was applied. This method stops the training process when reaching an optimum of the criterion function on held-out data.

5. Language model

The language model aims to provide a probability that a punctuation mark occurs within a given word context. For that purpose, we employed hidden-event N-gram language models [15]. These models are typically used in the following way: In the training text, the corresponding punctuation marks are replaced by <com> (comma) and <sen> (sentence end) tags. Instances of the "no punctuation" class are not explicitly marked; they are indicated simply by the absence of a tag. Then the punctuation can be treated the same way as a word token. In order to allow N-grams to span across sentence boundaries, the training text is not split into sentences as it is usual for training language models for standard ASR applications. Otherwise, standard N-gram training and smoothing techniques can be applied. During testing, the model is interpreted as a hidden Markov model (HMM). The target classes are treated as states and words are treated as observations. The requested probabilities can be computed via the forward-backward algorithm.

For training, we used newspaper texts taken from the Prague Dependency Treebank that is also available from LDC [16]. Our training text consisted of 7.4M tokens (280k distinct words). We also used manual transcripts taken from the speech training set. Although these transcripts represent only 3.5% of the whole training text, their addition significantly improved the performance of the model (Acc was increased by 1% absolute). This is probably because of the fact that the typical structure of sentences in the newspapers differs from the structure of sentences in the broadcast news. Hence, the obtaining of large amounts of news scripts from broadcast companies seems to be beneficial.

Besides word-based models, we also tested models using parts-of-speech (POS) and morphological tags. There exists a positional tag system for the Czech language. Every tag is represented by a string of 15 symbols. Each position in the string (excluding 2 reserve

Table 1: Acc, P, R and F [%] on the test set for different language models

	Acc	P	R	F
words only	91.03	70.93	41.76	52.57
tags only	89.91	67.81	33.98	45.27
tags $k = 7$	91.25	71.02	43.44	53.91
subtags $k = 7$	91.35	71.19	44.51	54.77

positions) corresponds to one morphological category. The categories are: POS, detailed POS, gender, number, case, possessor's gender, possessor's number, person, tense, grade, negation, voice, and variant (register). Positions representing categories not applicable for the tagged word are denoted by a single hyphen. For example, the word form "rezignoval" (lit. resigned) is correctly tagged as `VpYS---XR-AA---`. It means that it is a verb (V), past participle - active (p), masculine – either animate or inanimate (Y), singular (S), any person (X), past tense (R), not negated (A), and active voice (A). The number of possible distinct tags is quite high, 1362 different tags appeared in our training text that was run through an automatic tagger [16, 17].

A simple tag-based model trained on the tagged text did not perform as good as a word-based model, but we can use morphological tags in a different way. We found that it is useful to replace infrequent words (i.e. words occurring less than k times in the training text) by their tags and then to train the mixed language model on the text modified in this way. Likewise, in testing, the OOVs are replaced by their tags. We also found that instead of using the entire positional tag it is more convenient to use only a subtag containing following positions: *detailed POS*, *case* (original 7 cases were reduced to nominative, genitive, accusative, and "other"), *person*, *tense*, and *grade*. The optimal value for k was determined to be 7. The size of the vocabulary was thus reduced from 295k for the word-based model to 62k for the model mixing wordforms with subtags. The Witten-Bell discounting scheme was used for the model smoothing. The performances of various language models are reported in Table 1.

This method works better than a linear interpolation of a word-based and tag-based model. It can be viewed as a form of back off. When using it, we step back from details for rare word forms, whereas we keep the details for frequent word forms. The method also eliminates the problem of OOVs. Using the word-based model the OOV rate on the test set was 1.6%.

6. Model combination

Prosodic and lexical cues for the punctuation detection are generally considered to be largely complementary. Hence, the scores from the prosodic model and the language model can be successfully combined. Let E denote the sequence of punctuation symbols, X the prosodic features and W the corresponding sequence of words. In our task, we are looking for a sequence E_{MAP} having maximum a posteriori probability given W and X. This probability $P(E|W, X)$ can be expressed as

$$P(E|W, X) = \frac{P(X|W, E)P(E|W)}{P(X|W)} \tag{7}$$

Assuming that prosodic features depend only on the punctuation E and not on word identities, $P(X|W, E)$ in (7) can be replaced by $P(X|E)$. Note that this assumption is not always fully true, although we stylized and normalized the prosodic features to minimize their dependency on microprosody. However, this simplification is generally considered to be reasonable. Thus, equation (7) can be rewritten as

$$\begin{aligned} P(E|W, X) &\approx \frac{P(X|E)P(E|W)}{P(X|W)} \\ &= \frac{P(E|X)P(E|W)}{P(E)} \cdot \frac{P(X)}{P(X|W)} \end{aligned} \tag{8}$$

Because the fraction $\frac{P(X)}{P(X|W)}$ does not depend on E, we can search for E_{MAP} using the following proportionality

$$P(E|W, X) \propto \frac{P(E|X)P(E|W)}{P(E)} \tag{9}$$

Moreover, if we trained the prosodic classifier on data downsampled on equal class sizes, we can use

$$P(E|W, X) \propto P(E|X)^{\lambda} P(E|W) \tag{10}$$

where λ is an exponential scaling factor. Varying λ we can weight a relative contribution of either model. The optimal value of λ is determined on development data.

Assuming that the punctuation e_i depends only on the last prosodic observation x_i and that prosodic feature vectors are conditionally independent of each other given the punctuation e_i and the words W, we can use for searching for E_{MAP} a modified language model's HMM. The prosodic scores can be incorporated into the HMM as states emissions [1, 4].

7. Experimental results

The presented methods were tested on the above mentioned test set comprising 21,258 word-tokens. 1,515 (7.1%) of these word-tokens were followed by a sentence end, 1,309 (6,2%) tokens were followed by a comma, the rest (86.7%) were not followed by a punctuation mark. All tests were performed on true words (i.e. automatically aligned manual transcripts), not on ASR hypotheses.

The overall results are shown in Table 2. All tested models perform better than a priori chance. The best classification results ($Acc = 95.2\%$) were achieved by the combination of the language model and CART. The use of the MLP instead of the CART yielded slightly worse results. Also the best F-measure (78.2%) was achieved

Table 2: Overall accuracy, precision, recall and F-measure using different models [%]

	Acc	Acc'	P	P'	R	R'	F	F'
LM	91.35	91.87	71.19	76.00	44.51	46.34	54.77	57.58
CART	92.63	93.02	77.96	81.46	53.35	54.97	63.35	65.64
MLP	92.19	92.63	72.81	78.79	54.24	57.37	62.17	66.39
LM+CART	95.24	95.64	81.07	83.76	75.61	77.34	78.24	80.75
LM+MLP	95.14	95.43	87.27	89.72	68.94	70.46	77.03	78.93

Table 3: Precisions and recalls for detection of sentence ends (<sen>) and commas (<com>) using different models [%]

	P <sen>	P' <sen>	R <sen>	R' <sen>	P <com>	P' <com>	R <com>	R' <com>
LM	64.61	70.12	37.12	38.24	77.52	81.48	52.94	55.82
CART	82.55	85.63	90.50	91.95	51.07	55.75	10.92	11.44
MLP	79.55	83.50	92.91	93.69	38.48	39.94	10.72	11.32
LM+CART	85.70	88.90	90.57	92.23	74.01	75.99	58.52	61.18
LM+MLP	89.25	91.99	88.29	89.28	83.29	85.20	46.83	48.46

by the model combining the language model with the CART. The optimal exponential scaling factor for the combination of the prosodic and the language model was determined as $\lambda = 0.85$.

The separate results (P and R) for commas and sentence ends detection are reported in Table 3. These results indicate that the language model is better in detecting commas than in detecting sentence boundaries. By contrast, the prosodic model is better in detecting sentence boundaries. The reason is that the language model can use conjunctions as strong predictors of commas, whereas the prosodic model can use pauses as strong predictors of sentence ends – the commas are not so strongly prosodically marked. The detection of sentence boundaries by the N-gram language model is complicated due to the relatively free word order in Czech. Overall, the sentence boundaries were identified more accurately than commas which were detected with a significantly lower recall.

8. Conclusion and future work

In this paper, we report our initial experiments with automatic punctuation annotation from speech. We have focused on Czech broadcast news speech. We employed two statistical models – prosodic model and language model.

We tested two implementations of the prosodic model – CART and MLP. A slightly better results were achieved by the CART. For language modeling, we used hidden-event N-gram models. Instead of using an ordinary word-based model, we replaced infrequent word forms by their morphological tags and trained a mixed model. We also found that instead of using entire positional tags with 15 morphological categories, it is better to use only their subtags consisting of 5 categories.

Scores from both models can be combined. A model combining language model with the decision tree yielded superior results. Testing on true words, we achieved classification accuracy $Acc = 95.2\%$ and F-measure 78.2%.

There is a number of issues we would like to explore in the future. First of all, we know that we tested the methods on automatically aligned manual transcripts, not on hypotheses generated by an ASR system, so that the influence of ASR errors on the methods must be examined. We assume that the prosodic model will be less affected than the language model. However, the errors in word alignments will degrade it as well. The testing on ASR hypotheses also causes a problem with the performance evaluation; when the numbers of words in the reference and the ASR transcript differ, it is difficult to find the corresponding inter-word boundaries automatically.

Second, we will pay attention to the improvement of the used machine learning and feature selection techniques.

Further, we also would like to focus on the sentence boundary detection in spontaneous speech. Solving this task is crucial for the NLP applications dealing with spontaneous speech.

Acknowledgement

Support for this work was provided by the Ministry of Education of the Czech Republic, projects No. LN00A063 and MSM235200004.

References

[1] E. Shriberg, A.Stolcke, D. Hakkani-Tür, and G. Tür, "Prosody-based automatic segmentation of speech into sentences and topics," *Speech Communication*, vol. 32, no. 1-2, pp. 127–154, 2000.

[2] D. Baron, E. Shriberg, and A. Stolcke, "Automatic punctuation and disfluency detection in multi-party

meetings using prosodic and lexical cues," in *Proceedings of ICSLP*, Denver, USA, 2002, pp. 949–952.

[3] E. Shriberg and A. Stolcke, "Direct modeling of prosody: An overview of applications in automatic speech processing," in *Proceedings of International Conference Speech Prosody 2004*, Nara, Japan, 2004.

[4] J.-H. Kim and P. Woodland, "A combined punctuation generation and speech recognition system and its performance enhancement using prosody," *Speech Communication*, vol. 41, no. 4, pp. 563–577, 2003.

[5] H. Christensen, Y. Gotoh, and S. Renals, "Punctuation annotation using statistical prosody models," in *Proc. of ISCA Workshop on Prosody in Speech Recognition and Understanding*, Red Bank, NJ, USA, 2001.

[6] J. Huang and G. Zweig, "Maximum entropy model for punctuation annotation from speech," in *Proceedings of ICSLP 2002*, Denver, USA, 2002, pp. 917–920.

[7] V. Radová, J. Psutka, L. Müller, W. Byrne, J. V. Psutka, P. Ircing, and J. Matoušek, "Czech Broadcast News Speech and Transcripts," Linguistic Data Consortium, CD-ROM LDC2004S01 and LDC2004T01, Philadelphia, PA, USA, 2004.

[8] J. Psutka, V. Radová, L. Müller, J. Matoušek, P. Ircing, and D. Graff, "Large broadcast news and read speech corpora of spoken Czech," in *Proceedings of EUROSPEECH*. Aalborg, Denmark: ISCA, 2001, pp. 2067–2070.

[9] J. Kolář, J. Romportl, and J. Psutka, "The Czech speech and prosody database both for ASR and TTS purposes," in *Proceedings of EUROSPEECH'03*, vol. 2, Geneva, Switzerland, 2003, pp. 1577–1580.

[10] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," in *Speech Coding and Synthesis*. Amsterdam, Netherlands: Elsevier Science, 1995, pp. 495–518.

[11] K. Sönmez, L. Heck, M. Weintraub, and E. Shriberg, "A lognormal tied mixture model of pitch for prosody-based speaker recognition," in *Proceedings of EUROSPEECH'97*, Rhodes, Greece, 1997, pp. 1391–1394.

[12] K. Sönmez, E. Shriberg, L. Heck, and M. Weintraub, "Modeling dynamic prosodic variation for speaker verification," in *Proceedings of ICSLP*, Sydney, Australia, 1998, pp. 3189–3192.

[13] J. Buckow, V. Warnke, R. Huber, A. Batliner, E. Nöth, and H. Niemann, "Fast and robust features for prosodic classification," in *Proceedings of TSD'99 Mariánské Lázně*. Berlin: Springer, 1999, pp. 193–198.

[14] L. Breiman, J. Friedman, R. Ohlsen, and C. Stone, *Classification and Regression Trees*. Pacific Grove, CA: Wadsworth and Brooks Inc., 1984.

[15] A. Stolcke, E. Shriberg, R. Bates, M. Ostendorf, D. Hakkani, M. Plauche, G. Tür, and Y. Lu, "Automatic detection of sentence boundaries and disfluencies based on recognized words," in *Proceedings of ICSLP98*, Sydney, Australia, 1998.

[16] J. Hajič, E. Hajičová, P. Pajas, J. Panevová, P. Sgall, and B. Hladká, "Prague Dependency Treebank 1.0," Linguistic Data Consortium, CD-ROM LDC2001T10, Philadelphia, PA, USA, 2001.

[17] J. Hajič and B. Hladká, "Tagging inflective languages: prediction of morphological categories for a rich, structured tagset," in *Proceedings of the 17th international conference on Computational linguistics*, vol. 1, Quebec, Canada, 1998, pp. 483–490.