HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Predicting Prosodic Prominence from Text with Pre-trained Contextualized Word Representations

Aarne Talman

Antti Suni, Hande Celikkanat, Sofoklis Kakouros, Jörg Tiedemann and Martti Vainio

November 14, 2019

Department of Digital Humanities

University of Helsinki

FOTRAN Found in Translation

Outline

Introduction: Prosody

Prosodic prominence: the amount of emphasis that a speaker gives to a word.

- Prosody has been widely studied in phonetics and speech processing.
- Research on text-based natural language processing (NLP) methods is somewhat limited, even in text-to-speech synthesis domain.
- The main reason is lack of suitable and large enough datasets for the modern data-hungry approaches.

Predicting prosodic prominence from text:

- The task of predicting the prominence of each word in a sentence either as a continuous value or a discrete value.

Example:

One way led to the left and the other to the right **straight** up the mountain.

Predicting Prosody from Text

Can we predict prosodic prominence from text only?

(without any speech data)

Predicting Prosody from Text

Can we predict prosodic prominence from text only?

(without any speech data)

- Would be important for text-to-speech applications.
- For this we need a lot of text, annotated with prosodic prominence values.
- The only possible dataset available was the Boston University Radio Corpus, which is too small.

We decided to create a new dataset for the task.

LibriTTS

As a starting point we use the recently published LibriTTS corpus (Zen et al., 2019).

LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, Yonghui Wu

Google AI

heigazen@google.com

- LibriTTS is a multi-speaker English corpus of approximately 585 hours of read English speech and the corresponding text.
- It's a derived from the mp3 audio files from LibriVox and text files from Project Gutenberg.

Instead of manually annotating the texts we decided to use the acoustic signal from the speech data to automatically generate labeled training data for training, dev and test.

Helsinki Prosody Corpus

We introduce a new NLP benchmark and the largest annotated dataset for predicting prosodic prominence from text, with automatically generated high-quality annotations for the recently published LibriTTS corpus (Zen et al., 2019).

- For annotation we used the Wavelet Prosody Analyzer toolkit[1] which implements the method described in Suni et al. (2017).
  1. Extraction of pitch and energy signals from the speech data and duration from the word level alignments.
  2. Filling the unvoiced gaps in extracted signals by interpolation followed by smoothing and normalizing.
  3. Combining the normalized signals by summing or multiplication.
  4. Performing a continuous wavelet transform (CWT) on the composite signal and extracting continuous prominence values as lines of maximum amplitude across wavelet scales.
- The method assumes that the louder, the longer, and the higher the acoustic signal for a word is, the more prominent it is.

[1] https://github.com/asuni/wavelet_prosody_toolkit

Continuous Wavelet Transform Annotation Method

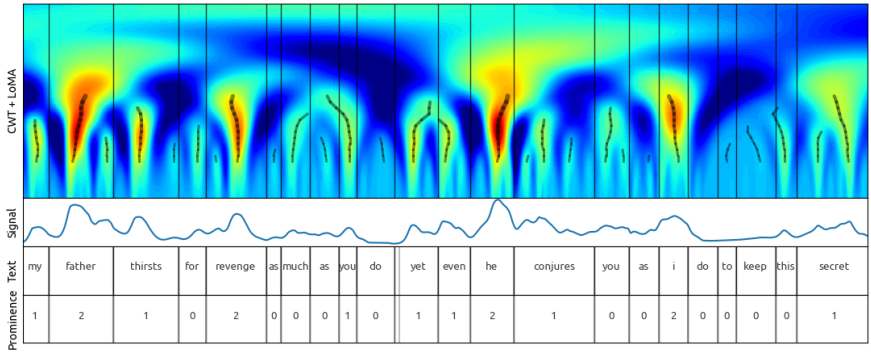

Figure 1: Continuous Wavelet Transform Annotation method.

Helsinki Prosody Corpus: Statistics

The resulting dataset contains over 2.8 Million annotated tokens divided into two training sets, a dev set and a test set.

				non-prominent	prominent	
sets (clean)	**speakers**	**sentences**	**words**	**0**	**1**	**2**
train-100	247	33,041	570,592	274,184	155,849	140,559
train-360	904	116,262	2,076,289	1,003,454	569,769	503,066
dev	40	5,726	99,200	47,535	27,454	24,211
test	39	4,821	90,063	43,234	24,543	22,286
total:	1230	159,850	2,836,144	1,368,407	777,615	690,122

Table 1: Dataset statistics

Helsinki Prosody Corpus

https://github.com/Helsinki-NLP/prosody

- Text files with one token per line.
- Sentences separated with a line: `<file> file_name.txt`, referring to the source file in LibriTTS.
- Each line has five items separated with tabs (with NA for punctuation):
  1. Token
  2. Discrete prominence label: 0 (non-prominent), 1 (prominent), 2 (highly prominent).
  3. Discrete word boundary label: 0, 1, 2.
  4. Continuous prominence value.
  5. Continuous word boundary value.

- Example sentence:

```
<file>    6829_68769_000053_000002.txt
That's     1     1     0.984     0.842
how        2     0     2.122     0.000
all        1     1     0.463     1.411
the        0     0     0.009     0.432
trouble    2     1     1.549     0.634
came       0     0     0.144     0.097
about      1     2     0.948     2.0
.          NA    NA    NA        NA
```

Experiments

The task of discrete prominence prediction from text is: *given text, predict for each word in each sentence the corresponding discrete prominence label*.

- We performed experiments from with multiple feature-based and neural models.
  - BERT-base uncased (Devlin et al., 2019).
  - 3-layer 600D BiLSTM (Hochreiter and Schmidhuber, 1997).
  - Minitagger (SVM) (Stratos and Collins, 2015) + GloVe (Pennington et al., 2014).
  - MarMoT (CRF) (Mueller et al., 2013).
  - Majority class per word.
- All systems except the Minitagger and CRF are our implementations using PyTorch and are available on GitHub: `https://github.com/Helsinki-NLP/prosody`.

Results

Experimental results for different models trained on the `train-360` dataset.

Model	Test accuracy (2-way)	Test accuracy (3-way)
BERT-base	**83.2%**	**68.6%**
3-layer BiLSTM	82.1%	66.4%
CRF	81.8%	66.4%
SVM+GloVe	80.8%	65.4%
Majority class per word	80.2%	62.4%
Majority class	52.0%	48.0%
Random	49.0%	39.5%

Table 2: Experimental results (%) for the 2 and 3-way classification tasks.

Results: learning curves

BERT outperforms the other models with just 5% of the training examples in the 2-way classification case and with 10% of the training data in the 3-way classification case.

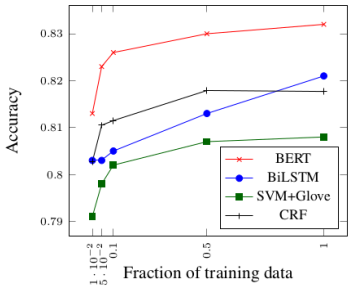

Figure 2: Test accuracy with different size subsets of the training data for the 2-way classification task.

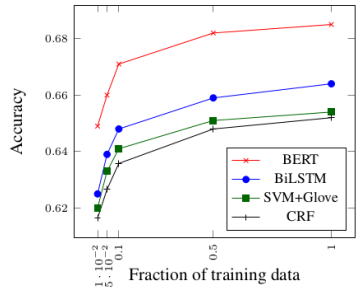

Figure 3: Test accuracy with different size subsets of the training data for the 3-way classification task.

Discussion

Although our experiments show that prosodic prominence can reasonably well be predicted from text, the scores are still quite low.

Some reasons could be:

- Errors in automatic alignment, signal processing, and quantization introduce noise to the labels.
- Different speakers have different accents, varying reading proficiency, and reading tempo, which all impact the consistency of the labeling.
- The source speech data contains genres ranging from non-fiction to metric poems with fixed prominence patterns and children's stories with high proportion of words emphasized. The difference in genres could impact the test results.
- The books included in the source speech data are all from pre-1923, whereas BERT and GloVe are pre-trained with contemporary texts.

Modeling the relationship between other linguistic properties and prosody

Can we model the relationship between other linguistic properties and prosody?
Can prosody features improve models for other tasks?
Can other linguistic features improve the prosody models?

- Textual entailment / NLI and prosody? (current work with Marianna Apidianaki)
- Syntactic structures and prosody? (future)

Unsupervised prosody prediction with NLI models

We used FaceBook's InferSent sentence encoder (Conneau et al., 2017) which is trained on the SNLI corpus (Bowman et al., 2015).

- We take the "importance" of each word in a sentence by analyzing which words contribute the most to the max pooled output.
- For each dimension in the InferSent's hidden BiLSTM layer we look at which word has the max value.
- We count the number each word contributes the max pooled output.

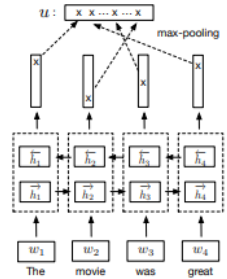

Figure 4: InferSent's BiLSTM max-pooling architecture.

Unsupervised prosody prediction with NLI models

- We use different thresholds to decide how much "word importance" is needed.
- We manually tune the threshold for labels 0, 1 and 2 based on the dev set accuracy.
- We then predict the prosody labels given the threshold.

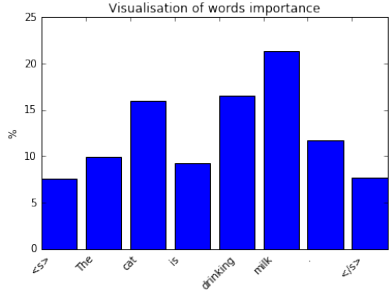

Figure 5: Visualization of the word importance.

Unsupervised prosody prediction with NLI models

The results are not great but they do beat the majority class baseline.

Model	Acc (2-way)	Acc (3-way)
InferSent	68.6%	51.5%
BERT-prosody (supervised)	83.2%	68.6%
BiLSTM-prosody (supervised)	82.1%	66.4%
Majority class	52.0%	48.0%
Random	49.0%	39.5%

Table 3: Experimental results (accuracy %) for unsupervised 2 and 3-way classification tasks.

Prosody prediction with a combined Prosody & NLI model

Can we utilize the learned features of an NLI model in prosody prediction?

Prosody prediction with a combined Prosody & NLI model

We combine the BiLSTM prosody model with InferSent.

- We pass the input sentences through the BiLSTM prosody model and the InferSent model, while freezing the InferSent.
- We concatenate the hidden states of the two models and pass the concatenated layer to a linear classifier.

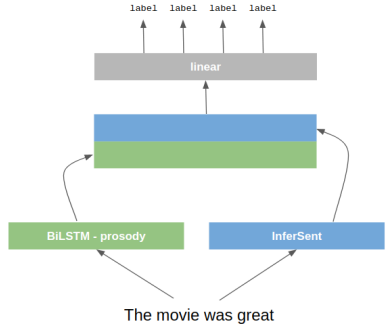

Figure 6: Concatenation model architecture.

Prosody prediction with a combined Prosody & NLI model

We combine the BiLSTM prosody model with InferSent.

- We pass the input sentences through the BiLSTM prosody model and the InferSent model, while freezing the InferSent.
- We concatenate the hidden states of the two models and pass the concatenated layer to a linear classifier.

Model	Acc (2-way)	Acc (3-way)
InferSent + BiLSTM-prosody concat	82.5%	67.0%
BiLSTM-prosody	82.1%	66.4%
BERT-prosody	83.2%	68.6%

Table 4: Experimental results (accuracy %) for supervised 2 and 3-way classification tasks.

The combined model seems to improve over the original BiLSTM prosody model (but not much).

Conclusion

- We introduced a new interesting NLP benchmark for predicting prosodic prominence from text.
- We published the largest publicly available datasets with prosodic labels.
- We tested wide variaty of models and show that BERT outperforms the other approaches even with a very small subset of training data.
- Work in progress to combine prosody prediction with other tasks.

Data and code:

https://github.com/Helsinki-NLP/prosody

References I

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

References II

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Mueller, T., Schmid, H., and Schütze, H. (2013). Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332. Association for Computational Linguistics.

References III

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Stratos, K. and Collins, M. (2015). Simple semi-supervised POS tagging. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*.

Suni, A., Šimko, J., Aalto, D., and Vainio, M. (2017). Hierarchical representation and estimation of prosody using continuous wavelet transform. *Computer Speech & Language*, 45:123–136.

Zen, H., Dang, V., Clark, R., Zhang, Y., Weiss, R. J., Jia, Y., Chen, Z., and Wu, Y. (2019). LibriTTS: A corpus derived from LibriSpeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.