SEQUENCE-TO-SEQUENCE MODELS FOR PUNCTUATED TRANSCRIPTION COMBINING LEXICAL AND ACOUSTIC FEATURES

Ondřej Klejch, Peter Bell, Steve Renals

Centre for Speech Technology Research, University of Edinburgh, Edinburgh EH8 9AB, UK
o.klejch@sms.ed.ac.uk, {peter.bell,s.renals}@ed.ac.uk

ABSTRACT

In this paper we present an extension of our previously described neural machine translation based system for punctuated transcription. This extension allows the system to map from per frame acoustic features to word level representations by replacing the traditional encoder in the encoder-decoder architecture with a hierarchical encoder. Furthermore, we show that a system combining lexical and acoustic features significantly outperforms systems using only a single source of features on all measured punctuation marks. The combination of lexical and acoustic features achieves a significant improvement in F-Measure of 1.5 absolute over the purely lexical neural machine translation based system.

Index Terms— punctuation, speech recognition, neural machine translation, rich transcription

1. INTRODUCTION

In this paper, we extend a neural machine translation (NMT) based system for punctuated transcription [1] to be able to use acoustic information. The main challenge of this extension is that the system should map between sequences with different timescales – per frame acoustic features as an input and punctuation marks as an output. We approach this challenge by replacing the traditional encoder in the encoder-decoder architecture [2] with a hierarchical encoder [3], which can map per frame acoustic features to word-level representations. Furthermore, since lexical and acoustic features cover different aspects of punctuation marks, it should be beneficial to combine them in a single system. Therefore, we evaluate several methods for combination of features from different sources using a vector concatenation and pooling operators, including stochastic mask pooling. In particular, we address the following research questions:

1. Is it sufficient to use only acoustic features for punctuated transcription?
2. Is it possible to use phonemes instead of words? Which representation leads to the most accurate punctuation of ASR output?
3. What is the best way to combine features from different sources, including acoustic and lexical features?

2. RELATED WORK

Automatic punctuated transcription is a well-studied problem, to which there have been three main approaches. First, the problem may be addressed by finding the most probable sequence of words and punctuation marks using language models [4, 5, 6, 7, 8], or finite state / hidden Markov models [9, 10]. A second approach tags each word with either the following punctuation mark or no punctuation mark [11, 12, 13, 14, 15, 10, 16, 17, 18]. Finally, punctuated transcription can be viewed as a machine translation problem in which unpunctuated text is translated to punctuated text [19, 20, 21]. For a more thorough review of these approaches to punctuated transcription see [1].

Punctuated transcription may also be categorised in terms of the features employed, primarily *lexical* features (most commonly n-gram statistics obtained from al language model, but potentially other features such as part-of-speech tags [16] or syntactic information from a sentence parse tree [15], and *acoustic prosodic* features, which might help to disambiguate between ambiguous punctuation marks, for example exclamation mark and full stop. The most important prosodic feature is a pause duration [10], but other features relating to phoneme duration, fundamental frequency, and energy [5, 7, 10, 12, 22] have been also used.

Training of neural networks with multiple sources of inputs has been addressed for many problems. Swietojanski et al. [23] used convolutional layers and max pooling operators to train a hybrid acoustic model for multichannel distant speech recognition. Zoph and Knight [24] presented a multi-source neural machine translation architecture that uses two source languages to translate to a target language (using two source languages to help translate ambiguous words). However, both these approaches were applied to input sequences of similar type and length. An example of a system that uses two different types of features is a multilingual image captioning neural sequence model [25], which is trained to describe an image given its English language description. Finally, multiple source training can be also thought of a version of many to one training [26], but this approach uses multiple sources only in training whereas we want to use multiple sources at test time.

3. NEURAL MACHINE TRANSLATION WITH HIERARCHICAL ENCODER

We have previously presented a system for punctuated transcription [1] based on a recurrent neural network (RNN) encoder-decoder architecture (with an attention layer), similar to that used for NMT [27, 28]. This system is trained to translate from sequences of words to sequences of punctuation marks (including blank). This approach is much more efficient than previous machine translation approaches that use statistical phrase based machine translation to translate from unpunctuated text to punctuated text [19, 20, 21], because our system has a much smaller output dictionary and this does not introduce new textual errors by incorrect translation.

In this paper we extend this system to use acoustic features. We use the fundamental frequency (F0) as our primary per frame acoustic feature, as it is correlated to the perceived pitch and the intonation of the sentence. Additionally, we also explored combining F0 with log mel-scale filterbank features (referred to as fbank+pitch features). To use these per frame acoustic features, we need to map them to word level representations. To this end, we replace the usual RNN encoder with a hierarchical encoder, similar to what has been successfully used in character based NMT [29] and in dialog state tracking for dialog acts representation [3]. The hierarchical encoder works as follows: First, a recurrent layer is used to obtain a frame level acoustic representation. Then, the representations of frames that correspond to word endings are used as word level acoustic embeddings, which are transformed using another recurrent layer to obtain a representation suitable for the decoder. The whole process of hierarchical encoding is illustrated in Figure 1.

4. COMBINING LEXICAL AND ACOUSTIC CUES

Since lexical and acoustic features capture different aspects of punctuation marks, it is potentially beneficial to combine them in order to create a system that achieves better performance on punctuation marks than a system which uses only lexical or acoustic features. Because lexical and acoustic features are asynchronous, it is not appropriate to simply concatenate their input features. Therefore, we decided to combine these two sources at the level of representations obtained from the encoders. We evaluated feature combination using a baseline of vector concatenation, and pooling operators such as max, average and sum.

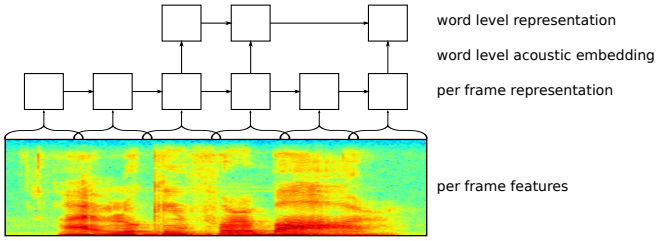

Fig. 1. Illustration of a hierarchical encoder encoding per frame acoustic features to word level features.

In our initial experiments we observed that the system which combines features in these ways is more prone to overfitting. We addressed this problem using a regularization approach based on dropout [30], which forces a neural network to learn a more robust representation by randomly resetting a portion of activations of nodes during training. This is similar to an approach for learning grounded meaning representations from images and words [31], in which a masking noise was used to enable the system to infer the missing modality from the available modality. Masking noise can be interpreted as a pooling operator, which we call *stochastic mask pooling*. At training time, stochastic mask pooling uses a random mask P sampled from a Bernoulli distribution with expected value p which selects those elements from the lexical representation X_L and those elements from the acoustic representation X_A which will be passed to the decoder.

$$R = PX_L + (1 - P)X_A; P \sim Ber(p)$$

During testing the expected representation is used, which is a weighted average of the lexical and acoustic representations with the weight set equal to the expected value of Bernoulli distribution p.

$$\mathbb{E}R = \mathbb{E}PX_L + (1 - \mathbb{E}P)X_A = pX_L + (1 - p)X_A$$

5. EXPERIMENTS

We conducted experiments on multi-genre broadcast speech data from the MGB Challenge dataset [32]. Preprocessing of the dataset for punctuated transcription is described in [1]. We used only lexical and acoustic data from the acoustic modelling training dataset with word matching error rate $< 10\%$ for training purposes. That is 162,000 sentences with 2.5 millions words and 216 hours of audio. We performed experiments on manually transcribed verbatim text (called reference in the Results section) and ASR output with a word error rate 31.6%. Our system is based on code from block.examples [33] and it is publicly available.[1] The systems used bidirectional gated recurrent units [34] with hidden layer size of 256 and was trained using AdaDelta [35] with dropout [30] for 100,000 iterations. During training we monitored F-Measure on the dev set every 5 000 iterations and we kept the best performing model for evaluation. We used a beam search with a beam size of 6 for decoding.

As a lexical baseline we used the NMT system from [1], which was trained on language modelling data, we refer to

[1] https://github.com/choko/acoustic_punctuation

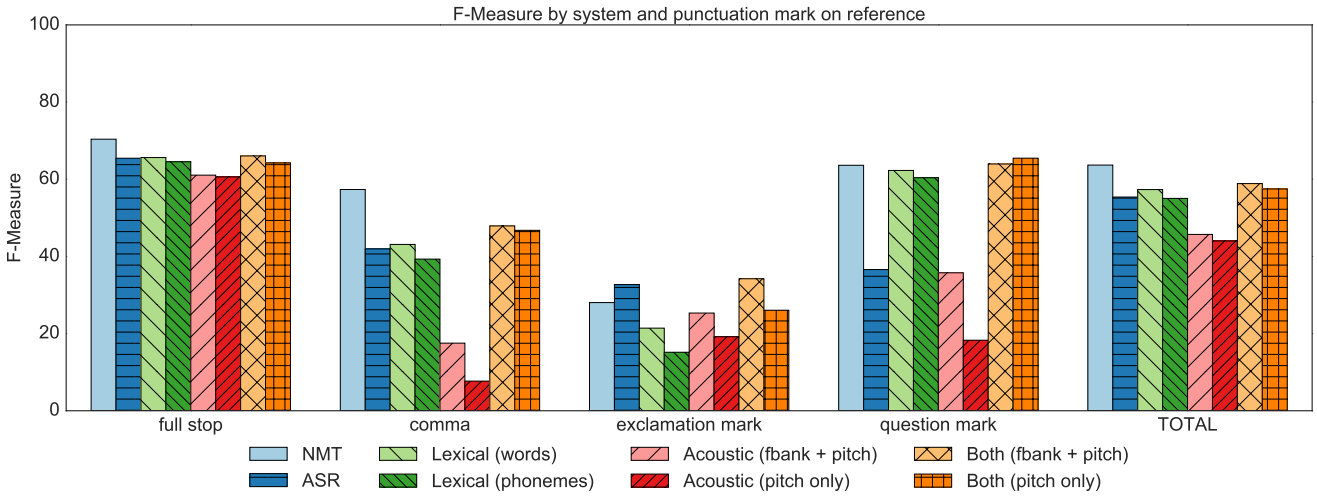

Fig. 2. F-Measure of punctuation marks on reference.

this system as **NMT**. Since the language modelling data is much larger than the data used in this paper, we also show results for a lexical system trained on the same amount of data as used in this paper. We refer to this system as **Lexical (words)**. Furthermore, we include results for a lexical baseline which uses the hierarchical encoder on phonemes referred as **Lexical (phonemes)**. As an acoustic baseline we used an ASR system from [1], which treats punctuation marks as normal tokens and uses a segment specific decoding graph to insert punctuation to already transcribed speech. We refer to this system as **ASR**. We trained a purely acoustic NMT system using the hierarchical encoder on globally normalized acoustic features obtained with the Kaldi toolkit [36]. For computation reasons we used only every third frame. We considered two types of features – 43 dimension fbank + pitch features, referred as **Acoustic (fbank + pitch)**, and 4 dimension pitch features, referred as **Acoustic (pitch only)**. Finally, we combined lexical features with acoustic features in two systems **Both (fbank + pitch)** and **Both (pitch only)** using stochastic mask pooling.

6. RESULTS

We measured the F-Measure of *full stop*, *comma*, *question mark*, *exclamation mark* and *three dots*. We used paired bootstrap resampling [37] to show that our improvements are significant. The results are summarized in Table 2, and Figures 2 and 3.

Looking at the lexical based systems, we see that **Lexical (words)** is worse than **NMT** by 6.25 absolute on reference and by 3.11 absolute on ASR output. This is due to the limited training data. Therefore, we compared all remaining systems with **Lexical (words)** in order to make a fair comparison. **Lexical (phonemes)** is worse than **Lexical (words)** by 2.27 absolute on reference and 1.26 absolute on ASR output, which suggests that although the phoneme based system is less affected by ASR errors, it still cannot achieve the performance of word based system.

Looking at the purely acoustic systems, we can see that they achieve poorer results than the lexical baseline. This is mainly because the systems cannot disambiguate full stops from other punctuation marks. Our hypothesis is that predictions of these systems are based on an inferred pause duration. The only punctuation mark for which acoustic based systems performed better is the exclamation mark. We believed that the acoustic based systems should work better than the lexical baseline on question marks, which are indicated by intonation raise, but the opposite was true. We hypothesize that lexical based systems are able to leverage the different word order in questions, and that the intonation raise is not that significant in practice. **Acoustic (fbank + pitch)** is significantly better than **Acoustic (pitch only)** with absolute improvement 1.69. But **Acoustic (pitch only)** suggests that **Acoustic (fbank + pitch)** is not learning word identities, because the difference between these systems is relatively small and it is not possible to predict words based on pitch features only.

	REF	**ASR**
Concatenation	55.90	48.92
Addition	56.62	48.50
Max pooling	57.57	49.23
Avg pooling	56.78	48.71
Stochastic mask pooling	57.88	49.24

Table 1. Comparison of methods for representation combination using fbank + pitch acoustic features.

When we look at the comparison of methods for representation combination in Table 1 we see that max pooling and stochastic mask pooling give the best results on this task. Stochastic mask pooling outperforms avg pooling with $p = 0.97$, addition with $p = 0.988$ and concatenation with

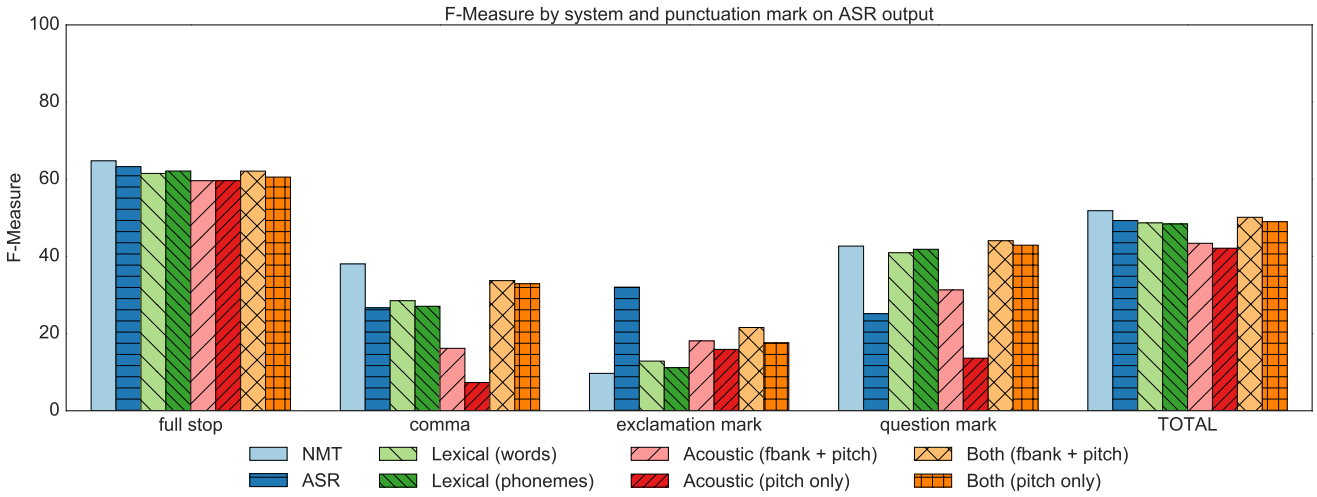

Fig. 3. F-Measure of punctuation marks on ASR output.

	full stop		comma		e. mark		q. mark		three dots		TOTAL	
	REF	ASR	REF	ASR	REF	ASR	REF	ASR	REF	ASR	REF	ASR
NMT baseline [1]	70.38	64.77	57.34	38.05	28.03	9.69	63.62	42.68	-	-	62.63	50.94
ASR baseline [1]	65.45	63.27	41.94	26.68	32.70	32.04	36.59	25.16	1.44	2.53	54.39	48.40
Lexical (words)	65.62	61.51	43.11	28.52	21.41	12.87	62.28	40.95	0.40	-	56.38	47.83
Lexical (phonemes)	64.53	60.97	39.30	26.83	15.16	10.49	60.39	40.79	0.40	-	54.11	46.57
Acoustic (fbank + pitch)	61.06	59.62	17.52	16.18	25.32	18.10	35.75	31.31	0.38	-	44.87	42.57
Acoustic (pitch only)	60.60	59.62	7.68	7.31	19.18	15.90	18.28	13.63	-	-	43.18	41.31
Both (fbank + pitch)	66.04	62.10	47.90	33.72	34.21	21.57	63.95	44.06	0.39	-	57.88	49.24
Both (pitch only)	64.23	60.56	46.76	32.91	26.05	17.58	65.46	42.91	0.39	-	56.62	48.13

Table 2. Results of the systems on reference (REF) and ASR output (ASR).

$p > 0.99$. The difference between stochastic pooling and max pooling is not statistically significant with $p = 0.717$. Looking at systems combining lexical and acoustic features we see that **Both (fbank + pitch)** is better than **Both (pitch only)**. When we compare the lexical baseline system with **Both (fbank + pitch)**, we see that incorporating acoustic features significantly improves overall performance with $p = 0.997$ with absolute improvement of 1.50 on reference and 1.41 on ASR output. Finally, we see that systems incorporating acoustic features are less affected by ASR errors than systems using only lexical features.

7. CONCLUSIONS

In this paper we presented an extension of our NMT based system for punctuated transcription. This extension allows the system to incorporate per frame acoustic features by replacing the traditional encoder with the hierarchical encoder, which can map per frame features to word-level representations. We also evaluated methods for combination of multiple sources of features in one system by using different methods including stochastic mask pooling. Our results show that a system incorporating acoustic features significantly outperforms purely lexical systems and are less affected by ASR errors.

In the future work we would like to explore ways of leveraging textual data without corresponding audio for training of the system which combines lexical and acoustic features.

8. ACKNOWLEDGMENTS

This work was supported by the H2020 project SUMMA, under grant agreement 688139.

9. REFERENCES

[1] O. Klejch, P. Bell, and S. Renals, "Punctuated transcription of multi-genre broadcasts using acoustic and lexical approaches," submitted to SLT 2016, available at `http://homepages.inf.ed.ac.uk/s1569734/papers/slt-2016.pdf`.

[2] I. Sutskever, O. Vinyals, and Q. Le, "Sequence to sequence learning with neural networks," in *NIPS*, 2014.

[3] I. V Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *AAAI*, 2016.

[4] A. Stolcke and E. Shriberg, "Automatic linguistic segmentation of conversational speech," in *ICSLP*, 1996, pp. 1005–1008.

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

[6] D. Beeferman, A. Berger, and J. Lafferty, "Cyberpunc: A lightweight punctuation annotation system for speech," in *ICASSP*, 1998.

[7] J.-H. Kim and P. C Woodland, "The use of prosody in a combined system for punctuation generation and speech recognition.," in *Interspeech*, 2001.

[8] A. Gravano, M. Jansche, and M. Bacchiani, "Restoring punctuation and capitalization in transcribed speech," in *ICASSP*, 2009.

[9] Y. Gotoh and S. Renals, "Sentence boundary detection in broadcast speech transcripts," 2000.

[10] H. Christensen, Y. Gotoh, and S. Renals, "Punctuation annotation using statistical prosody models," in *ISCA Workshop on Prosody in Speech Recognition and Understanding*, 2001.

[11] J. Huang and G. Zweig, "Maximum entropy model for punctuation annotation from speech," in *INTERSPEECH*, 2002.

[12] J. Kolář, J. Švec, and J. Psutka, "Automatic punctuation annotation in Czech broadcast news speech," *SPECOM*, 2004.

[13] J. Kolář and L. Lamel, "Development and evaluation of automatic punctuation for French and English speech-to-text.," in *Interspeech*, 2012.

[14] W. Lu and H. T. Ng, "Better punctuation prediction with dynamic conditional random fields," in *EMNLP*, 2010.

[15] N. Ueffing, M. Bisani, and P. Vozila, "Improved models for automatic punctuation prediction for spoken and written text.," in *Interspeech*, 2013.

[16] E. Cho, K. Kilgour, J. Niehues, and A. Waibel, "Combination of NN and CRF models for joint detection of punctuation and disfluencies," in *Interspeech*, 2015.

[17] O. Tilk and T. Alumäe, "LSTM for punctuation restoration in speech transcripts," in *Interspeech*, 2015.

[18] O. Tilk and T. Alumäe, "Bidirectional recurrent neural network with attention mechanism for punctuation restoration," in *Interspeech*, 2016.

[19] S. Peitz, M. Freitag, A. Mauser, and H. Ney, "Modeling punctuation prediction as machine translation.," in *IWSLT*, 2011.

[20] E. Cho, J. Niehues, and A. Waibel, "Segmentation and punctuation prediction in speech language translation using a monolingual translation system.," in *IWSLT*, 2012.

[21] J. Driesen, A. Birch, S. Grimsey, S. Safarfashandi, J. Gauthier, M. Simpson, and S. Renals, "Automated production of truecased punctuated subtitles for weather and news broadcasts," in *Interspeech*, 2014.

[22] M. Hasan, R. Doddipatla, and T. Hain, "Noise-matched training of CRF based sentence end detection models," in *Interspeech*, 2015.

[23] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *Signal Processing Letters*, 2014.

[24] B. Zoph and K. Knight, "Multi-source neural translation," *arXiv preprint arXiv:1601.00710*, 2016.

[25] D. Elliott, S. Frank, and E. Hasler, "Multi-language image description with neural sequence models," *arXiv preprint arXiv:1510.04709*, 2015.

[26] M.-T. Luong, Q. V Le, I. Sutskever, O. Vinyals, and L. Kaiser, "Multi-task sequence to sequence learning," *arXiv preprint arXiv:1511.06114*, 2015.

[27] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[28] R. Sennrich and B. Haddow, "Linguistic input features improve neural machine translation," *arXiv preprint arXiv:1606.02892*, 2016.

[29] W. Ling, I. Trancoso, C. Dyer, and A. W Black, "Character-based neural machine translation," *arXiv preprint arXiv:1511.04586*, 2015.

[30] N. Srivastava, G. E Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[31] C. Silberer and M. Lapata, "Learning grounded meaning representations with autoencoders.," in *ACL*, 2014.

[32] P. Bell, MJF Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, and P. C Woodland, "The MGB challenge: Evaluating multi-genre broadcast media recognition," in *ASRU*, 2015.

[33] B. Van Merriënboer, D. Bahdanau, V. Dumoulin, D. Serdyuk, D. Warde-Farley, J. Chorowski, and Y. Bengio, "Blocks and fuel: Frameworks for deep learning," *arXiv preprint arXiv:1506.00619*, 2015.

[34] J. Chung, C. Gülçehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," *arXiv preprint arXiv:1502.02367*, 2015.

[35] M. D Zeiler, "ADADELTA: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[36] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi speech recognition toolkit," in *ASRU*, 2011.

[37] P. Koehn, "Statistical significance tests for machine translation evaluation.," in *EMNLP*, 2004.