INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Out-of-vocabulary Words Detection with Attention and CTC Alignments in an End-to-End ASR System

Ekaterina Egorova, Hari Krishna Vydana, Lukáš Burget, Jan "Honza" Černocký

Brno University of Technology, Faculty of Information Technology, Speech@FIT, Czechia
{iegorova,vydana,burget,cernocky}@fit.vutbr.cz

Abstract

This work explores the effectiveness of detecting positions of out-of-vocabulary words (OOVs) in a decoded utterance using attention weights and CTC per-frame outputs of an end-to-end system predicting word sequences. We show that the end-to-end approach can be effective for the task of OOV detection. CTC alignments are shown to provide better temporal information about the positions of OOV words than attention, and therefore are more suitable for the task. The detected positions of OOV occurrences are utilized for the recurrent OOV recovery task in which probabilistic representations of the pronunciations of the detected OOVs are clustered in order to find repeating words. Improved detection results are shown to correlate with better performance of the recovery of recurrent OOVs.

Index Terms: Speech recognition, Out-of-vocabulary, OOV, Attention, CTC, End-to-end

1. Introduction and Previous Work

Out-of-vocabulary words (OOVs) pose one of the persistent problems in automatic speech recognition (ASR) and other speech mining tasks, as language is changing and new words constantly emerge. In a widely used hybrid ANN/HMM ASR system [1], OOVs are not represented in the dictionary or language models (LMs) and therefore cannot be correctly recognized. End-to-end (E2E) approaches avoid the limitations of a dictionary by predicting character labels. However, in case an E2E system predicts words [2], OOVs pose a problem, as the network has no distinctive labels for these words in the training and therefore cannot not learn their representation.

In a classic hybrid ANN/HMM ASR system, dealing with OOVs usually involves lattices of subword units, which can be either linguistically motivated (phonemes, syllables, etc.) or data-driven (e.g. SentencePiece [3]). One approach within a hybrid framework is to have two separate systems: a word and a subword system. Decoding works on the word level and switches to the subword level only if the output does not fit the pre-set conditions (e.g. minimum confidence score etc.) [4]. Thus, detection and recovery stages are separated: low confidence of a word system means that an OOV is detected, and then its representation is recovered from the subword system. Another approach exploits the combination of word and subword units on the language model level, training a "flat" hybrid LM with both [5, 6, 7]. This approach combines detection and recovery tasks in one: an OOV is detected when the output contains a string of subword units, and these units also provide acoustic representation of the OOV.

If OOVs are repeating, which is a reasonable assumption for most new, trending words, the strings of subword units can be clustered to discover these repeating OOVs [8, 9]. The clustering criteria may include phonetic and acoustic features and context information [10]. A more effective way of finding repeating OOVs is performing clustering on lattices of subword units instead of clustering one-best strings, thus introducing pronunciation uncertainty [11].

In [11], we used a word-subword Weighted Finite State Transducer (WFST) decoder for OOV detection and recovery tasks. OOV occurrences were extracted from the time periods for which decoded hybrid word-subword lattices produced subword units. This approach had the benefit of combining detection with generating phoneme representation, as well as several drawbacks: one of them was the size of the decoding graph and the resulting output lattices; another was that detection rates were low due to the system having lower costs on shorter paths containing only words, instead of longer paths with phoneme units. Even introducing cost penalties on word arcs in decoded lattices during WFST-decoding did not produce enough improvement in terms of detection recall. The highest detection recall we were able to achieve was 26%. Low detection rate hurts recovery performance, which brings us to investigate E2E approaches in this paper to improve the detection task.

In the recent paper [12], an E2E system that predicts word sequences with an attention mechanism is used for discovering positions of OOVs. For each output label, attention points to the relevant input frames; frames with the maximum attention weight for the OOV label can be used to find OOV center frame. These positions of OOVs are then used to improve generation of subtitles with the use of subword units. In this work, we explore the effectiveness of this method in more detail and evaluate new methods of detecting OOVs using an E2E word predicting system. In particular, we show that CTC alignment information from a hybrid CTC/attention architecture [13] is more effective for OOV detection than attention information from Listen Attend and Spell (LAS) model [14]. We also process the output of the OOV detection step with an OOV recovery pipeline and assess the effect that detection has on recovery task.

Most of the aforementioned works report results on private datasets, which makes it hard to reproduce them. We report results on the well-known and publicly available Librispeech dataset [15] and propose reasonable success metrics in hope of bringing more standardization to the task.

2. Task Formulation and Success Metrics

There are two tasks in OOV processing pipeline: detection and recovery. Detection task deals with finding OOV segments in speech and should return the start and end times of where an OOV can be found. OOV recovery task aims to recover acoustic and/or graphemic representation of OOVs or link them to an existing word identity. Although detection is a challenging task on its own, our ultimate goal is to improve recovery of repeating OOVs, and we will investigate the success of the detection task mostly from the perspective of how it affects the recovery task.

Copyright © 2021 ISCA

http://dx.doi.org/10.21437/Interspeech.2021-1756

2.1. Detection Task

First, an E2E ASR system that predicts word sequences is used as a base for an OOV detector. The input to this stage are acoustic features, and as the output we get predicted word labels and also access to internal hidden representations of words. The details of our E2E system implementation are given in section 3. Then we perform OOV detection with the help of information obtained from the E2E system. The desired output of this stage is a list of timestamps marking the beginnings and ends of OOVs. In this paper, we experiment with two approaches to this task. The first approach involves estimating OOV positions from attention weights, and the second approach uses per-frame CTC predictions. Both methods are described in more detail in the experiment section.

At this stage, **detection success** is evaluated. The reference timing is obtained by force aligning the reference transcriptions containing target OOVs to the acoustic features. These force alignments have been done using a system trained as described in [11] following a standard Kaldi [16] recipe. When reporting the OOV detection, an OOV occurrence is treated as a true positive if the hypothesis overlaps with the reference for more than half of the reference duration. Thus, the detection recall is calculated as the % of true positives in the reference list of OOVs, and detection precision as the % of true positives in the list of detected OOV occurrences. As the ultimate goal of our experiments is improving recovery of repeating OOVs, the detection recall is more important – while incorrectly detected occurrences will most likely form singleton clusters and therefore be ignored after the clustering stage, the occurrences that are not detected at all have no chance to be recovered.

2.2. Recovery Task

Whichever of the two methods of detection is used, the detected time regions are passed to the next stage that provides phonetic representations of OOV occurrences for further processing. As our goal is to find recurring OOVs via clustering of these phonetic representations, the processing pipeline is the following. First, detected OOVs shorted than 0.5 second are discarded, to avoid getting too many occurrences to cluster. Moreover, shorter occurrences are usually not full words but rather suffixes and hesitations. Then, a Kaldi [16] phoneme recognizer system is used to generate decoded phoneme lattices. From the decoded phoneme lattices, 50 best phoneme strings are generated, together with their probabilities and time alignments. From these, all phoneme substrings that are within the detected start and end times are extracted. This way, we extract a number of phoneme substrings (together with their probabilities) as a set of alternative pronunciations for each detected OOV occurrence. We efficiently store the alternative pronunciations and their probabilities in the form of WFST by performing union of the linear WFST corresponding to the individual 50 alternative pronunciations followed by minimization of the resulting WFST.

In the next step, we cluster the OOV occurrences based on their pronunciation similarity using the information encoded in their corresponding WFSTs. This clustering method takes into consideration pronunciation uncertainty and in the converged state contains clusters, each corresponding to one OOV with a unique pronunciation. The clustering is based on a generative model that assumes that each OOV occurrence is a string of phonemes sampled from an infinite mixture model where each mixture component has a certain weight and a given (deterministic) pronunciation. The mixture model follows the Dirichlet Process [17] prior, whereby the base distribution is a uniform distribution over all possible sequences of phonemes. The Chinese Restaurant Process -based inference is used to infer (the posterior distribution over) the mixture components corresponding to our detected (observed) OOVs, which tell how many different OOVs there are in the data and what their pronunciations are[1]. This inference has to further take into account that we are uncertain about the input observations (i.e. each OOV occurrence is represented by a distribution (WFST) over the possible pronunciations).

After the clustering process converges, clusters of the size ≥ 2 (as we are only interested in recovering repeating OOVs) are post-processed to obtain graphemic representations of the recovered OOVs. For this purpose, the most likely phonetic pronunciation of each cluster is passed through a phoneme-to-grapheme system. We use a joint-sequence model [18], but it is trained on a reverse, phoneme to grapheme, dictionary.

The output of the whole pipeline is a list of previously unseen OOV words in their graphemic representation. When reporting the **OOV recovery success**, this list is compared to the list of the reference OOVs with the help of Levenstein distance. A word is marked as recovered (true positive) if its graphemic representation does not differ from some reference word by a distance more than 1. For example, a recovered word *"morover"* would be considered correctly recovered, but *"anctious"* – not. Thus, in our pipeline, successful OOV recovery depends on 1) enough occurrences of the same word present in the text and 2) successfully detected, 3) the clustering assigning enough of these occurrences to the same cluster, and 4) the phoneme-to-grapheme conversion correctly discovering the graphemic representation of the word from its subword unit representation. Recovery recall shows the percentage of OOVs from the reference list that were recovered, and recovery precision shows the percentage of the recovered words that belong to the reference OOV list.

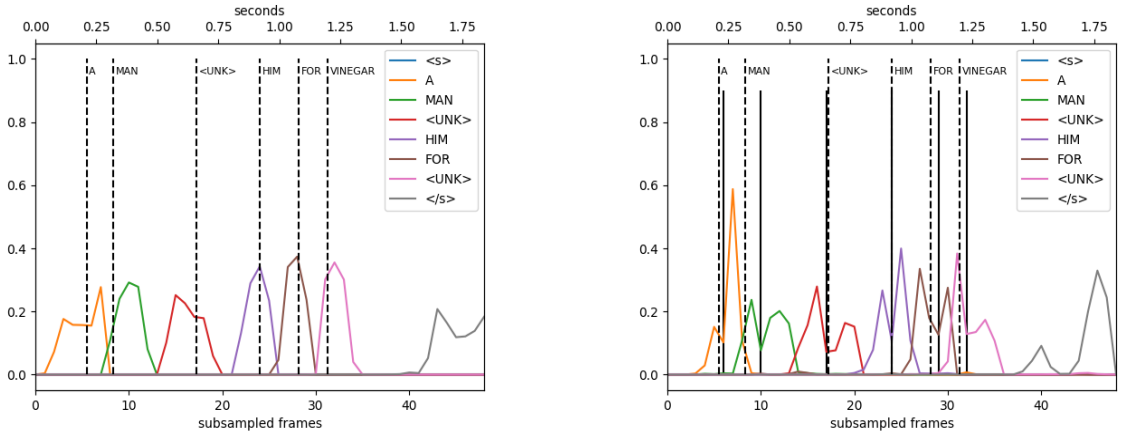
3. E2E ASR Systems for OOV Detection

Two E2E ASR systems are used in this work as the sources for OOV detection: one trained with attention, and another with a hybrid CTC/attention architecture.

For attention-based OOV detection, Listen Attend and Spell (LAS) model [14] is used. The model has an encoder with 5-layers and a single layer decoder. Each encoder layer has a bidirectional LSTM layer followed by a linear projection layer (LSTM-P) and all the encoder layers have residual connections. Dropout is applied on the output of the LSTM network. The decoder is a single unidirectional LSTM. Both the encoder and the decoder have a hidden layer size of 320 units. The LSTMs used in the encoder are bi-directional so the output size is double the hidden size and the following linear projection layer projects these representations back to the hidden layer size. The input sequence is subsampled by a factor of 2 by initial two LSTM-P layers. The network is optimized to predict words from the Mel-filter bank features.

For OOV detection with the use of CTC, E2E system is trained with a hybrid CTC/attention architecture described in [13]. The total loss used for optimizing the model has two components: the CTC loss (L_{ctc}) and label-smoothed cross-entropy between the predicted and the ground-truth label sequences (L_{Att}), which is estimated from the decoder of the LAS model. Both losses are combined as follows:

[1] `www.fit.vutbr.cz/~iegorova/public/CRP_Adaptation_for_FST_Clustering.pdf`



(a) *LAS*

(b) *hybrid attention/CTC with 0.9 CTC cost weight*

Figure 1: *Comparison of attention and CTC alignments to real OOV times. Lower x axis is in frames and the upper one is in seconds. Dashed lines with labels show reference transcriptions with reference word timings in brackets. Black lines show borders of words according to CTC. Colored plots show attention weights for corresponding output labels.*

$$loss = \beta \times L_{ctc} + (1 - \beta) \times L_{Att}, \quad (1)$$

with the label smoothing factor $\beta = 0.1$ [19, 20]. The models are trained with a uniform scheduled sampling scheme using 60% of the ground truth tokens.

The model is optimized using the ADAM [21] optimizer with the learning rate of 0.003, and the initial learning rate is halved upon encountering an increase in the validation error rate. The model is early-stopped upon encountering an increase in the validation accuracy for three epochs. The predicted label sequences are decoded with a beam size of 10.

From the input feature sequence of T frames: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_T]$, the encoder produces the encoded hidden representation $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \ldots, \mathbf{h}_T]$. Let y_i and y_{i-1} be the present and the past predicted labels, $\mathbf{s}_i$ is the state/output of the decoder RNN, and $\mathbf{c}_i$ is the context vector from the attention module. The current output label y_i is predicted by the LAS decoder as follows:

$$\mathbf{s}_i = RNN(\mathbf{s}_{i-i}, \mathbf{c}_{i-1}, y_{i-1}) \quad (2)$$

$$\mathbf{c}_i = Attention(\mathbf{s}_i, H) \quad (3)$$

$$P(y_i|\mathbf{X}; y_{i-1}, y_{i-2}, .., y_1) = Linear([\mathbf{s}_i, \mathbf{c}_i]). \quad (4)$$

with $Attention(\mathbf{s}_i, H)$ evaluated as follows:

$$f_{i,j} = \mathbf{F} * \boldsymbol{\alpha}_{i-1} \quad (5)$$

$$e_{i,j} = \mathbf{z}^T \times \tanh(\mathbf{U}\mathbf{s}_{i-1} + \mathbf{V}\mathbf{h}_{j-1} + \mathbf{W}f_{i,j} + \mathbf{b}) \quad (6)$$

$$\alpha_{i,j} = \exp(e_{i,j}) / \sum_{k=1}^{T} \exp(e_{i,k}) \quad (7)$$

$$\mathbf{c}_i = \sum_{j}^{T} (\alpha_{i,j}\mathbf{h}_j). \quad (8)$$

Here, the weights $\mathbf{U}$, $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{F}$ are trainable weight matrices and $\mathbf{z}$, $\mathbf{b}$ are trainable vectors; $\mathbf{c}_i$ is a weighted sum of $\mathbf{H}$ using the frame-level attention weights $\alpha_{i,j}$. Therefore, $\mathbf{c}_i$ can be seen as a summary vector representing a subsequence in $\mathbf{H}$ that is responsible for producing the current label y_i. Eqs. 5 and 6 are the location aware attention from [22].

4. Experiments and results

4.1. Data and OOV Simulation Setup

In continuation of the experiments in [11], the database for the current experiments was the LibriSpeech ASR corpus of audiobooks [15]. From LibriSpeech, 100 hours of clean data and 500 hours of "other" data are used for system training. OOV discovery is done on the remaining chunk of 360 hours of clean data, which we call "OOV eval data". The LibriSpeech dictionary contains 200000 words, but for E2E training the number of word labels has been reduced to 10000 most common words (minimum 32 occurrences per word). OOV symbol is also included among the E2E target labels and replaces all the less-frequent words for the training.

Since the audiobooks are predominantly from the 19th century, a real OOV scenario is impossible – there are no new words. Thus, an artificially created list of OOVs for testing has been taken from [11]. In short, it is a "reverse" choice of OOVs from outdated 19th-century words that are less likely to be seen in modern dictionaries and language models. The resulting list [2] of OOVs consists of 1000 designated OOVs, which exemplify 19-century bookish English, for instance, it includes words such as *interposed*, *hastened*, *mademoiselle*, *indignantly*, *countenance*, etc. There are enough high-repetition words that would be the targets for the recurrent OOVs recovery task: the words' frequency in the OOV eval data ranges from 0 to 296 reference occurrences, with the mean of 51 occurrences. All

[2] `www.fit.vutbr.cz/~iegorova/public/LibriSpeech_1000_OOV_list.txt`

Table 1: *Comparison of BPE baseline results with [23]. %WERs for word-predicting E2E systems on 10k vocabulary.*

system	dev_c	dev_o	test_c	test_o
[23] no LM	4.87	14.37	4.87	15.39
BPE 960hrs	4.99	15.18	5.02	15.65
LAS attention	14.21	26.61	14.58	27.19
LAS att. (OOV word)	8.66	21.78	8.79	22.36
ctc+att $\beta = 0.9$	15.38	27.29	16.00	27.85

Table 2: *OOV Detection and recovery results using methods described in subsections 4.3 and 4.4. Column "OOV occurrences" shows the amount of occurrences on the clustering input, and column "clusters" shows the amount of clusters of size 2 or more after clustering saturates.*

Detection method	OOV occurrences	Detection		Recovery		
		recall	precision	clusters	recall	precision
attention $>$ 0.5 sec	57701	33.4%	14.5%	147	0.7%	6%
att $>$ 0.5 sec 0.2 right shift	57669	34.4%	14.9%	489	5.5%	29%
ctc+att $\beta = 0.5$	134843	73.7%	32.0%	1996	8.3%	15%
ctc+att $\beta = 0.9$	148428	81.5%	35.3%	3485	14.0%	15%

recovery recall and precision metrics are calculated on this list. The resulting OOV rate (percentage of OOVs in all the words) is 1.5%. For the purpose of E2E training, all the words from the list have also been assigned the same OOV label as the words not included in the 10k vocabulary.

4.2. E2E WER baselines

Classic E2E systems do not deal with OOVs, as they are trained to predict characters or byte pair units (BPEs) [3] that represent less frequent words as word parts. Table 1 shows, that, when tested on the full LibriSpeech database (960 hours), our BPE system functions on par with the systems with no LM reported in [23]. For the OOV experiments, the training of E2E system predicting words was done on 600 hours, leaving out the OOV eval data. E2E targets are 10000 words that do not include words from the OOV list described in subsection 4.1. The third row shows WER as calculated on reference transcription; it is much higher than for the BPE experiment due to the introduction of OOVs. The fourth row shows the potential of the E2E system as an OOV detector, as it treats predicting the OOV label for an OOV word as correct, and not as substitution. Finally, the last row shows WER for an E2E system that gives most weight to the CTC training objective. For the ASR task, it performs worse than the pure attention system, but as we will see later, this system is helpful in the OOV detection task.

4.3. OOV Detection with Attention

A LAS-based E2E system is used for the OOV detection experiments described in this section. For each output label, attention vector $\boldsymbol{\alpha}_i$ is pointing to certain frames that are relevant to the current decision. However, there is no guarantee that attention will be aligned to the real position of the word in the output. If we look just at the maximum of attention weight and consider the frame it points to as the hypothesized OOV time, we discover that only 26% of these frames lie within a reference OOV timing, and the detection recall is just 32%. This contradicts [12], where the centers of attention were assumed to point to the centers of OOVs. However, if we look at whether there is an overlap between the reference timings and the frames that are responsible for 90% of attention mass, 87% of the hypothesis time spans intersects with reference timings, suggesting 75% detection recall.

If we plot LAS attention vectors $\boldsymbol{\alpha}_i$ for each predicted word label as shown in Fig.1(a), we get more insight into what happens. It can be noted that the maxima of attention lie before the centers of words and the span of attention is indeed earlier in time than the reference word alignments.

This shift has negative influence on recovery: if we convert frames receiving 90% of attention mass for OOVs into times, and then extract their pronunciations according to these times and run clustering on them, the result will be very poor (see Tab. 2, the first row). There are almost no repeating phoneme strings to cluster, hence, there are not many clusters of at least size 2 and a low recovery recall. To solve this problem, the next experiment was made with timings obtained from attention with different delays. The best results have been observed with a delay of 0.2 seconds (see Tab. 2, the second row).

4.4. OOV Detection with CTC

In an E2E system trained with a hybrid CTC/attention architecture, CTC also provides a way of obtaining timing information from the labels. Word boundaries are positioned on the frames for which a new label is predicted. When the output is a blank symbol or the same label as before, there is no word boundary.

Table 2 shows the OOV detection results for the hybrid attention/CTC system with different weights given to attention and CTC costs (β from equation 1) in rows 3 and 4, respectively. Both result in more extracted OOV occurrences than attention due to the fact that attention tends to be very spiky. These occurrences also have much better overlap with reference timings, as confirmed by the detection scores. They also cluster much better and improve the recovery recall and precision metrics.

Interestingly, while attention provides better WER in general, CTC alignments are more trustworthy for the task of OOV timings extraction. This can be seen from the results obtained by the system with 0.9 cost given to CTC (4th row in Table 2). In Fig. 1(b), alignments from CTC are drawn as black lines. It can be seen that, although attention does not reflect OOV position well, the word boundaries from CTC alignment mostly correspond to the reference word boundaries.

5. Conclusions

Our experiments have shown that the E2E approach definitely has a potential to be applied for the task of OOV detection and outperforms detection by a hybrid lattice framework presented in [11]. CTC alignments provide better temporal information about word positions than the pure attention-based E2E system and so are more suitable for extracting OOV occurrences. The improved detection results also correlate with better recovery of recurrent OOVs. For the pure attention-based E2E model it can be seen that, even though the system performs better in terms of WER, there is no guarantee that attention actually points to the positions of frames directly responsible for producing the output label in question.

6. Acknowledgements

The work was supported by Czech National Science Foundation (GACR) project "NEUREM3" No.19-26934X and by European Union's Horizon 2020 project No.870930 - "WELCOME". Part of high-performance computations ran on IT4I supercomputer and was supported by the Ministry of Education, Youth and Sports of the Czech Republic through e-INFRA CZ (ID:90140).

7. References

[1] M. Mohri, F. Pereira, and M. Riley, "Speech Recognition with Weighted Finite-State Transducers," *Handbook on Speech Processing and Speech Communication*, 2008.

[2] H. Soltau, H. Liao, and H. Sak, "Neural Speech Recognizer: Acoustic-to-Word LSTM Model for Large Vocabulary Speech Recognition," *Proceedings of INTERSPEECH*, 2017.

[3] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing," *EMNLP*, 2018.

[4] L. Lee, J. Glass, H. Lee, and C. Chan, "Spoken Content Retrieval—Beyond Cascading Speech Recognition with Text Retrieval," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, 2015.

[5] A. Yazgan and M. Saraclar, "Hybrid language models for out of vocabulary word detection on large vocabulary conversational speech recognition," *Proceedings of ICASSP*, 2004.

[6] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models," *Proceedings of INTERSPEECH*, 2005.

[7] S. Kombrink, M. Hannemann, and L. Burget, "Out-of-Vocabulary Word Detection and Beyond," *Studies in Computational Intelligence*, vol. 384, p. 57–65, 2012.

[8] L. Qin and A. Rudnicky, "Learning Better Lexical Properties for Recurrent OOV Words," *Proceedings of ASRU*, 2013.

[9] M. Hannemann, S. Kombrink, M. Karafiát, and L. Burget, "Similarity Scoring for Recognizing Repeated Out-of-Vocabulary Words," *Proceedings of INTERSPEECH*, p. 897–900, 2010.

[10] L. Qin and A. Rudnicky, "Finding Recurrent Out-of-Vocabulary Words," *Proceedings of INTERSPEECH*, 2013.

[11] E. Egorova and L. Burget, "Out-of-Vocabulary Word Recovery Using FST-Based Subword Unit Clustering in a Hybrid ASR System," *Proceedings of ICASSP*, pp. 5919–5923, 2018.

[12] S. Thomas, K. Audhkhasi, Z. Tüuske, Y. Huang, and M. Picheny, "Detection and Recovery of OOVs for Improved English Broadcast News Captioning," *Proceedings of INTERSPEECH*, 2019.

[13] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/Attention Architecture for End-to-End Speech Recognition," *Journal of Selected Topics in Signal Processing*, 2017.

[14] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[15] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "LIBRISPEECH: an ASR Corpus Based on Public Domain Audio Books," *Proceedings of ICASSP Conference*, 2015.

[16] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemanna, P. Motlícek, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi Speech Recognition Toolkit," *Proceedings of ASRU Conference*, 2011.

[17] B. A. Frigyik, A. Kapila, and M. R. Gupta, "Introduction to the dirichlet distribution and related processes," in *UWEE Technical Report*, 2010.

[18] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, p. 434–451, 2008.

[19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[20] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Advances in Neural Information Processing Systems*, 2019, pp. 4694–4703.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[23] A. Zeyer, K. Irie, R. Schlüter, and H. Ney, "Improved training of end-to-end attention models for speech recognition," *Interspeech*, 2018.