INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

End-to-end Language Identification using Attention-based Recurrent Neural Networks

Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai and Bo Xu

Interactive Digital Media Technology Research Center,
Institute of Automation Chinese Academy of Sciences, Beijing 100190, China
{wang.geng, wangwenfu2013, yyzhao5231, xinyuan.cai, xubo}@ia.ac.cn

Abstract

This paper proposes a novel attention-based recurrent neural network (RNN) to build an end-to-end automatic language identification (LID) system. Inspired by the success of attention mechanism on a range of sequence-to-sequence tasks, this work introduces the attention mechanism with long short term memory (LSTM) encoder to the sequence-to-tag LID task. This unified architecture extends the end-to-end training method to LID system and dramatically boosts the system performance. Firstly, a language category embedding module is used to provide attentional vector which guides the derivation of the utterance level representation. Secondly, two attention approaches are explored: a soft attention which attends all source frames and a hard one that focuses on a subset of the sequential input. Thirdly, a hybrid test method which traverses all gold labels is adopted in the inference phase. Experimental results show that 8.2% relative equal error rate (EER) reduction is obtained compared with the LSTM-based frame level system by the soft approach and 34.33% performance improvement is observed compared to the conventional i-Vector system.

Index Terms: language identification, end-to-end training, attention mechanism, recurrent neural networks

1. Introduction

Recently, end-to-end automatic speech recognition system benefites from the successful application of the deep neural networks (DNN) and connection temporal classification (CTC) loss function [1, 2, 3, 4]. However, the DNN based automatic language identification (LID) system is still short of a universal end-to-end framework though many attempts have tried to apply DNN to LID task at larger scale. The deep bottleneck feed forward neural networks (DBN) that works as a front-end feature extractor has greatly enhanced the gaussian mixture model-universal background model (GMM-UBM) based i-Vector LID system [5, 6, 7, 8, 9, 10]. Moreover, a unified DBN which covers both the front-end high-level feature extraction and back-end acoustic modeling stage is proposed to apply neural networks to LID task at larger scale [11].

However, the complex architecture of the above DNN based i-Vector framework detriments its expansibility. Particularly, the neural networks applied to LID task in previous works are either shallow architectures or developed independently from the classifier. Resent works have attempted to address this disjoint developing issue by designing models that are trained end-to-end. For instance, motivated by the powerful modeling capability and discriminative nature of DNNs, the work applied the DNN directly to the LID task at the acoustic frame level [12]. This is the first time to directly apply a unified DNN model to automatic LID at large scale. Further, LSTM RNN [13, 14] is adopted to the LID task to model the long-range dependency across the input temporal sequence with respect to feed forward neural networks [12, 15]. The above two works make a great stride forward towards to end-to-end LID system [12, 15].

At present, a unified attention model has recently shown very promising performance on a range of pattern recognition tasks, such as speech recognition [16, 17, 18], neural machine translation (NMT) [19], handwriting synthesis [13, 20], image caption generation [21], and visual object classification [22]. Such models are mainly composed of two modules: RNN-based encoder and sequence generator. These models process the sequential input by iteratively selecting relevant content through the attention mechanism. This elaborately designed attention mechanism is good at dealing with the structured problem: mapping one variable-length sequence to another variable-length sequence. But different from the CTC, there is no limitation to the length relationship between the input and output sequence with regard to the attention model. These sequence-to-sequence models with attention mechanism significantly extend the applicability of end-to-end training method [20].

Motivated by the attention mechanism, this paper proposes a novel neural network structure to implement utterance level classification for end-to-end automatic language identification. The proposed attention-based recurrent neural network is internally composed of several modules, including a LSTM-based encoder, the language category embedding module and the utterance level classifier. To the best of our knowledge, this is the first time to establish an utterance level end-to-end LID system based on a unified attention model. The LSTM RNNs encoder is used for modeling the long-range temporal dependencies across the input acoustic sequences. Language category embedding is a pretrained language category mapping matrix. It is used to provide the attentional vectors through the popular lookup table operation. The attentional vector selectively evaluates the output activations of the LSTM RNNs encoder, and then selects the relevant key frames from the input sequence to generate an utterance level vector. This utterance level real-valued vector will be fed into the classifier to implement the end-to-end LID task. Distinguished from the sequence-to-sequence structured issues, the automatic language identification task is a sequence-to-tag issue that the global statistic information is used for classification. As we all know, sequence-to-sequence learning is a framework that attempts to address the issue of learning variable-length input and predict the output sequence [23]. Each token in the output sequence is generated by the current corresponding input source and its previous output to-

The work is supported by 973 Program in China, grant No. 2013CB329302.

http://dx.doi.org/10.21437/Interspeech.2016-686

ken. While, the sequence-to-tag problem is merely related to the relevant statistic information in the input sequence. Thus, extending the attention mechanism to LID task needs specially designed neural network model.

Similar to the other attention-based models, the LSTM RNNs are adopted to encode long-span connections of the input sequence. The elaborately designed language category embedding module is motivated by neural machine translation. It is a language category mapping matrix which is used to address the two challenges when applying the the attention-based model to the LID task. The first challenge is the notable bias problem existing in the joint training procedure, the second one is the inconsistence between the model training and inference within this attention-based sequence-to-tag structure. In order to circumvent these undesired behaviors, we propose to modify the attention mechanism such that it explicitly takes into account only the acoustic input sequence in the learning precedure. This language category embedding based attention mechanism reduces impact of the utterance gold label. The attentional vector is provided by the lookup table operation to attend the encoded high-level features and then selects the key representative frames in the input sequence. Depending on whether the "attention" is paid on all of the source frames or on only a few source frames, two attention methods are developed in the paper: soft attention and hard attention approach, which is similar in the spirit to the works [13, 19]. Meanwhile, two attention scoring methods are investigated in this paper. The first scoring approach is the cosine distance (dot mode) of the attentional vector and the hidden activation, while the second alternative is the general mode learned from the work [19].

The remainder of the paper is organized as follows: Section 2 gives a description of the attention-based framework for end-to-end LID. Experimental results and analysis are presented in Section 3, and our whole work is summarized in Section 4.

2. Attention-based recurrent neural networks

The proposed unified attention-based recurrent neural network is mainly composed of three modules: the encoder LSTM, the language category embedding module and the utterance level classifier. The encoder is two stacking LSTM RNNs to model the long-term dependencies across the conventional acoustic input. The language category embedding is a pretrained language category mapping matrix for providing the corresponding attentional vectors by means of the regular lookup table operation in neural machine translation. The fixed-size representative real-valued vector is generated through the selectiveness of the attentional vector on the output activations of the 2 hidden layers LSTM RNNs encoder. The end-to-end automatic language identification task is implemented by feeding this utterance level real-valued vector into the classifier. Figure 1 illustrates the attention-based model implementation in this paper and the computation formulates are as follows:.

$$\boldsymbol{h}_t = Recurrent(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) \quad (1)$$

$$\boldsymbol{l}_k = Label2Vec(y_k) \quad (2)$$

$$a_t = Attend(\boldsymbol{l}_k, \boldsymbol{h}_t) \quad (3)$$

$$\boldsymbol{c} = \sum_{t=1}^{T} a_t \boldsymbol{h}_t \quad (4)$$

where $\boldsymbol{x}_t$ and $\boldsymbol{h}_t$ represent the t-th step input and its corresponding hidden representation. y_k and $\boldsymbol{l}_k$ are the k-th language category and its real-valued vector. $\boldsymbol{a}$ is the attentional weight vector and $\boldsymbol{c}$ is the generated utterance representation.

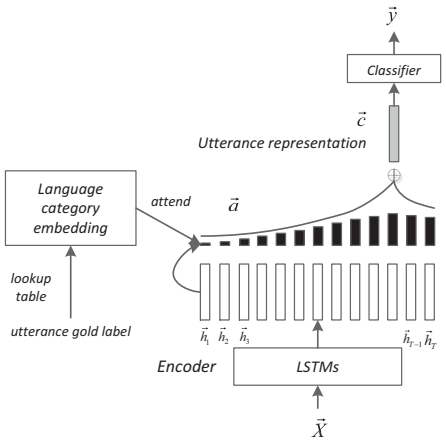

Figure 1: *The Architecture of attention-based recurrent neural network.*

2.1. Attention-based recurrent neural networks architecture

2.1.1. LSTM RNNs encoder

The encoder is implemented by a stacking LSTM RNNs. The LSTM which is equipped with various learnable gates is an enhanced RNN architecture, this elaborately designed gating unit ensures that the gradients can effectively flow back to the past. Benefitted from the powerful capability of modeling temporal dependencies across the input sequence, the LSTM RNNs have achieved great success in many pattern recognition tasks, including neural machine translation [19, 20], speech synthesis [24, 25] and speech recognition [1, 2]. The LSTM encoder models the acoustic feature $\boldsymbol{x}$ and outputs the input representation $\boldsymbol{h} = \{\boldsymbol{h}_1, ..., \boldsymbol{h}_T\}$, which is more suitable to work with the attention mechanism.

2.1.2. Language category embedding module

Different from the sequence-to-sequence structured problems, the automatic language identification task is a sequence-to-tag issue that the output posterior probability relies exclusively upon the relevant discriminative statistic content in the sequential input. The gold label of the utterance works merely for relevant information evaluation and selection. In sequence-to-sequence issue, the joint training procedure of the conventional attention-based model updates both the encoder and the decoder based on the strong interaction of the hidden states of the decoder and the sequential output of the encoder. Instead, as for the sequence-to-tag issue, this coupling operation of the encoder and decoder brings great bias problem that the model will learn more about the gold label rather than the training materials. Thus, the elaborately designed language category embedding module is adopted to reduce impact of the utterance gold label when extending the attention mechanism to LID task.

This language category embedding module is motivated by the NMT [19, 20] and it is used to provide the attentional vectors. Language category embedding converts a language gold label into a dense, fixed-size, real valued vector representation, which is similar in spirit of the word embedding [26, 27, 28, 29, 30]. This language category embedding learns compact vector representation for language categories and best fits the end-to-end utterance level classification. This language category embedding matrix L is pre-trained similar to the word embedding matrix in NMT. When it is applied to the attention-based recurrent neural networks, it can be either fixed in the w-

hole training procedure or updated along with the LSTM RNNs encoder in the last few iterations of the attention network.

When the language category embedding learning is finished, each language gold label in the category set corresponds to an attentional vector $\boldsymbol{x} \in R^D$, and all the vectors are stacked into an embedding matrix $L \in R^{D \times K}$. Thus, each language has an index k into the column of the embedding matrix L, note the K is the number of categories.

The internal attentional vector $\boldsymbol{x} \in R^D$ is given by the popular lookup table operation:

$$\boldsymbol{l}_k = Le_k \in R^D \tag{5}$$

where a binary vector $\boldsymbol{e}_k$ is used for retrieving the language vector representation through a simple multiplication with the embedding matrix L.

2.1.3. Utterance level representation

The attention-based recurrent neural networks derive utterance representation through the composition mechanism of the encoder and language category embedding. During the generation of utterance level real-valued vector, the composed model takes as input the hidden state h_t at the top layer of the stacking LSTM RNNs, and then to derive a utterance representation $\boldsymbol{c}$ which is fed into the classifier to implement an end-to-end fashion. The frame representation h_t out of the encoder serves as both the frame factor for attentional score calculation and the acoustic representation for utterance level representation accumulation. The frames belonging to the same kind of language are close to each other in the embedding space. This utterance representation is in fact performing frame clustering with respect to the utterance gold label.

2.2. Two kinds of attentional score calculation methods

This paper investigates two types of scoring methods to evaluate the attentional score between the attentional vector and the hidden state of the LSTM RNNs encoder.

The first scoring approach is the cosine distance between the attentional vector and the encoder hidden activation.

$$s_t = Score(\boldsymbol{l}_k, \boldsymbol{h}_t) = \boldsymbol{l}_k \cdot \boldsymbol{h}_t \tag{6}$$

The second alternative is the general mode inspired by the work [19].

$$s_t = Score(\boldsymbol{l}_k, \boldsymbol{h}_t) = \boldsymbol{l}_k W \boldsymbol{h}_t \tag{7}$$

$$a_t = \exp(s_t) \Big/ \sum_{t=1}^{T} \exp(s_t) \tag{8}$$

Where the a_t are the attentional weight and the softmax operation ensures that the $a_t \geq 0$ and the sum of a_t is unit. The $\boldsymbol{h}_t$ is the hidden state of the encoder and the $\boldsymbol{l}_k$ is the attentional vector.

2.3. Attention mechanism

According to whether attending to all the source frames or only a subset source frames, two attention methods are explored when implementing an end to end LID system: A soft attention approach [20] always takes into account all input frames and a hard one only focuses on a subset of the most promising frames when computing the utterance level fixed-sized vector.

The soft attention mechanism focuses on all the hidden states of the LSTM encoder when deriving the utterance representation $\boldsymbol{c}$. In this attention mode, a variable-length attentional weights vector $\boldsymbol{a}$, whose size equals the length of one utterance, is obtained by calculating the attentional score between attentional vector and each hidden state h_t. While the hard attention mechanism considers only the frames in a sampling window which covers a subset of the last few representative frames in one utterance. It is important to point out, the hard attention mode [21] is the same with the soft one except that the attentional weights vector is fixed-length and shorter than the length of the attended utterance. Further more, the hard attention mechanism best fits the LSTM RNNs encoder, since this hard attentional method takes into account of the temporal modeling property of the LSTM encoder and it selectively focuses on a small window of subset frames. Additionally, this approach holds the advantage of reducing the expensive computation complexity incurred in the soft one.

3. Experiments

3.1. Experimental setups

The NIST Language Recognition Evaluation (LRE) 2007 dataset is used for demonstrating the effectiveness of the proposed attention-based model adopted in this paper. The training dataset is composed of LRE05_OHSU, CALLFRIEND and LID05el and the experiment test corpus is a subset of the official NIST LRE 2007 3s condition evaluation set. 14 kinds of language and 2158 segments are included in the 3s evaluation data. The sequential input of the attention-based model is the 42-dimensional acoustic feature vectors that composed of 13-dimensional perceptual linear prediction coefficients (PLP) and pitch coefficient along with their first and second delta. All experiments are carried out on the open toolkit KALDI [31]. For experimental comparison, the LSTM RNNs based frame level LID system is established. The investigated LSTM RNNs model is the same configuration with the LSTM RNNs encoder in the attention model that each hidden layer contains 800 memory cells with 512 recurrent projection units. All models are optimized with the famous truncated backpropagation through time (BPTT) learning algorithm [15, 13].

3.2. Evaluation approach

Distinguished from the sequence-to-sequence issues, the attention-based sequence-to-tag structure faces the inconsistence problem between the model training and evaluation. As for the sequence-to-sequence issue like the NMT, during both of the training and inference phase, the token "BOS" is used to derive the first token in the generated sequence and the predicted "EOS" indicates the end of the generation process of the new sequence. However, with regard to the sequence-to-tag problem, in the training phase, the gold labels of the training examples are used to retrieve the corresponding attentional vectors in the language category embedding through the lookup table operation. While, when evaluating the learned model, we lack of prior knowledge (the gold label is unknown) about the test utterance. This inconsistence between the model training and inference is the most serious problem to prevent the extension of attention mechanism to sequence-to-tag tasks. Thus, based on the special working mode of the proposed attention model, a hybrid evaluation method to traverse all gold labels for generating the full score matrix M is adopted. The task-related test score is calculated by searching the most promising score row in this full score Matrix M. The M is a 14*14 score matrix. Each row represents the classification score vector when one kind of language category gold label is used to attend the test

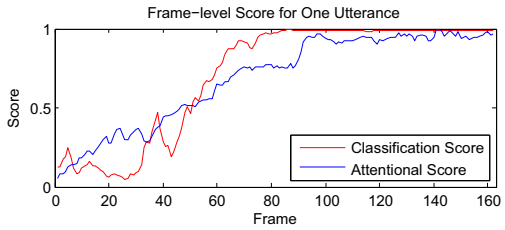

Figure 2: *Frame level attentional score and classification score.*

utterance. While, each column represents the posterior score on this category when the 14 gold labels are used for attention.

This elaborately designed attention mechanism architecture: the language category embedding module, ensures that the model learning procedure explicitly takes into account only the acoustic input sequence. Thus, it is obvious that the target category score of the utterance would mostly be higher no matter which kind of gold label to provide the attentional vector. What's more, the corresponding gold label of this utterance would lead to the highest score in the score matrix M. Based on the above observation and analysis, two types of task-related evaluation approaches are explored: majority voting for row classification result (called majority voting) and maximum score in the score matrix (called max score).

3.3. Experimental results and analysis

We evaluate attention-based end-to-end utterance level LID system in this section. The attentional score and the corresponding frame level classification score on the target category are analyzed. The two kinds of attentional score calculation methods mentioned above are discussed and two types of task-related evaluation approach are investigated.

3.3.1. Reliability and effectiveness of attention mechanism

Figure 2 illustrates the attentional score (blue line) and the corresponding classification score (red line) on the target category of each frame. Based on the long span dependencies modeling ability of the LSTMs, the sequential output of the LSTM RNNs becomes more discriminative over time in one utterance. Thus, higher attentional score and classification score should be obtained with increasing time. The red line is the frame level classification score of each frame on target category. It demonstrates the sequential modeling ability of the LSTMs that the classification score increases with time lasting in one utterance. Simultaneously, the attentional score increases over time which shows a uniform trend with the classification score as illustrated in Figure 2. The increasing attentional score over time confirms the reliability and the selectivity of the attention mechanism.

Table.1 gives a comparison between the LSTM-based frame level LID system and the attention-based end-to-end LID system on the soft average score calculation approach. The soft scoring in LSTM-based LID system is identical to the method adopted in works [12, 15] : averaging the log of the classifier output of all the frames in an utterance. The soft scoring in attention-based LID system concerns about all the source frames when deriving the utterance representation which is similar to the global attention in work [19]. Observed from Table.1, the attention-based end-to-end LID system achieves 8.2% relative EER reduction compared with the LSTM-based frame level system. The excellent performance confirms the effectiveness of the attention mechanism.

Table 1: *System performance of different models (EER %) on LRE 2007 (3s segments).*

model	EER(%)
i-Vector	20.39
LSTM RNNs	16.03
Attention model	14.72

3.3.2. Evaluation of the attention mechanism

Table.2 gives a brief description about the attention-based LID system. Where, maj_vot is the majority voting evaluation approach and max_sco is the max score approach.

Table 2: *Performance of attention-based model (EER %) on LRE 2007 (3s segments).*

model	dot		general	
	maj_vot	max_sco	maj_vot	max_sco
soft attention	15.33	14.72	15.61	15.33
hard attention	13.34	13.39	13.47	13.44

Firstly, both of the two attentional score calculation methods achieve considerable performance. Especially under the hard attention condition, both of them achieve about 13.4% EER performance that 34.33% performance improvement is observed comparing with the i-Vector system. Secondly, the hard attention mechanism outperforms the soft attention mechanism as show in Table.2. The hard attention benefits from the sequential modeling of the LSTM encoder. Since the LSTM encoder can capture long-range context information, the encoder output of one frame contains more language discriminative information if it locates at the back position in one utterance. Thus, the latter frames are representative enough while the front ones are lack of representation. Based on this property of the LSTM encoder, the hard attention approach can directly avoid the selectivity of the representation insufficiency frames to prevent bringing in noisy frames, which conduces its better performance over the soft attention. Thirdly, both of the two task-related evaluation approaches are reliable which coincides with the working mechanism of the attention-based model.

4. Conclusions

An end-to-end LID system is established based-on the LSTM RNNs with the attention mechanism. The attention-based model is composed of several modules, including a LSTM encoder, a language category embedding module and a utterance level classifier. The LSTM encoder is used for modeling the long-span contextual dependencies across the sequential input. The language category embedding is a pretrained language category mapping matrix which is used to provide the attentional vectors through the popular lookup table operation. The attentional vector selectively attends the output activations of the LSTM RNNs encoder and generates the utterance representation. The utterance representation is fed into the classifier to implement the end-to-end LID task. Two attention approaches: the soft attention approach and the hard one, are investigated in this paper. Experimental result confirms the effectiveness of the proposed attention-based end-to-end LID system. Observed from the experiment, 8.2% relative EER reduction is obtained compared with the LSTM-based frame level system by the soft attention approach and 34.33% performance improvement is obtained compared to the conventional i-Vector system.

5. References

[1] Y. Miao, M. Gowayyed, and F. Metze, "Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding," *arXiv preprint arXiv:1507.08240*, 2015.

[2] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1764–1772.

[3] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[4] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[5] W. Geng, J. Li, S. Zhang, X. Cai, and B. Xu, "Multilingual tandem bottleneck feature for language identification," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[6] Y. Song, B. Jiang, Y. Bao, S. Wei, and L.-R. Dai, "I-vector representation based on bottleneck features for language identification," *Electronics Letters*, vol. 49, no. 24, pp. 1569–1570, 2013.

[7] B. Jiang, Y. Song, S. Wei, I. V. McLoughlin, and L.-R. Dai, "Task-aware deep bottleneck features for spoken language identification." in *INTERSPEECH*, 2014, pp. 3012–3016.

[8] Y. Song, R. Cui, X. Hong, I. McLoughlin, J. Shi, and L. Dai, "Improved language identification using deep bottleneck network," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4200–4204.

[9] P. Matejka, L. Zhang, T. Ng, S. H. Mallidi, O. Glembek, J. Ma, and B. Zhang, "Neural network bottleneck features for language identification." Citeseer, 2014, pp. 299–304.

[10] F. Richardson, D. Reynolds, and N. Dehak, "A unified deep neural network for speaker and language recognition," 2015.

[11] Y. Song, X. Hong, B. Jiang, R. Cui, I. V. McLoughlin, and L. Dai, "Deep bottleneck network based i-vector representation for language identification," 2015.

[12] I. Lopez-Moreno, J. Gonzalez-Dominguez, O. Plchot, D. Martinez, J. Gonzalez-Rodriguez, and P. Moreno, "Automatic language identification using deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5337–5341.

[13] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] J. Gonzalez-Dominguez, I. Lopez-Moreno, H. Sak, J. Gonzalez-Rodriguez, and P. J. Moreno, "Automatic language identification using long short-term memory recurrent neural networks." in *INTERSPEECH*, 2014, pp. 2155–2159.

[16] 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.

[17] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: first results," *arXiv preprint arXiv:1412.1602*, 2014.

[18] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," *arXiv preprint arXiv:1508.04395*, 2015.

[19] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

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

[21] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *arXiv preprint arXiv:1502.03044*, 2015.

[22] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in Neural Information Processing Systems*, 2014, pp. 2204–2212.

[23] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[24] W. Wang, S. Xu, and B. Xu, "Gating recurrent mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *Proc. ICASSP*. IEEE, 2016.

[25] Y. Fan, Y. Qian, F.-L. Xie, and F. K. Soong, "Tts synthesis with bidirectional lstm based recurrent neural networks." in *Interspeech*, 2014, pp. 1964–1968.

[26] J. Zhang, S. Liu, M. Li, M. Zhou, and C. Zong, "Bilingually-constrained phrase embeddings for machine translation." in *ACL (1)*, 2014, pp. 111–121.

[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[28] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *INTERSPEECH*, vol. 2, 2010, p. 3.

[29] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain, "Neural probabilistic language models," in *Innovations in Machine Learning*. Springer, 2006, pp. 137–186.

[30] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[31] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.