A BAYESIAN ATTENTION NEURAL NETWORK LAYER FOR SPEAKER RECOGNITION

Weizhong Zhu, Jason Pelecanos[†]

IBM Research AI
Yorktown Heights, NY 10598, USA
zhuwe@us.ibm.com

ABSTRACT

Neural network based attention modeling has found utility in areas such as visual analysis, speech recognition and more recently speaker recognition. Attention represents a gating (or weighting) function on information and governs how the corresponding statistics are accumulated. In the context of speaker recognition, attention can be incorporated as a frame weighted mean of an information stream. These weights can be made to sum to one (the standard approach) or be calculated in other ways. If the weights can be made to represent event observation probabilities, we can extend the approach to be within a Bayesian framework. More specifically, we combine prior information with the frame weighted statistics to produce an adapted or posterior estimate of the mean. We evaluate the proposed method on NIST data.

Index Terms— attention modeling, Bayesian statistics, deep neural networks, speaker recognition

1. INTRODUCTION

Speaker recognition technology has greatly improved with the advance of computational systems and the availability of extensive speech resources. Neural networks are playing a significant role and are continuing to be integrated into different speech technologies, including speaker recognition.

Some of the earlier work in speaker recognition looked at how neural networks could be used to train frame-level speaker discriminant features using a neural network bottleneck structure [1]. More than a decade later, and with an increase in available training data, research explored the use of a segment level criterion [2]. Other work [3] investigated different configurations of bottleneck features applied to both the language and speaker identification tasks.

Over time, research also explored the training of end-to-end systems. Salmon [4] investigated possibly the first moderately successful end-to-end neural network system. It incorporated an autoencoder to improve generalization and also proposed the use of a mean and standard deviation based intermediate layer within a siamese network structure to generate low rank speaker embeddings. More recently, Snyder published a state-of-the-art result [5, 6] for a shorter duration (10-second) condition on the NIST 2010 speaker recognition data set [7]. This system is similar to the work of Salman but was trained using the cross-entropy training criterion for N speaker classes and the generated speaker embedding was later scored within a PLDA framework. For smaller data sets, neural network training needs to be carefully constrained. In contrast, for a larger data set (approx 80K speakers), it was shown that an end-to-end neural network could be effectively trained with fewer constraints [8].

Other work included the use of senone posteriors produced by a neural network as a partitioning function for generating senone conditioned i-vectors [9]. Interestingly, this represents a gating mechanism across senones and it has similarities with attention models in neural networks (for example see [10] in the visual domain and [11, 12] as applied to speaker recognition). Attention models basically represent a gating and accumulation mechanism on information whether it be across time or across nodes of a network layer. To date, attention models typically calculate a sample (or speech frame) weighted mean of a particular input of interest. This can be seen as a maximum-likelihood style estimate of the mean. In this work, and in contrast, we propose the use of Bayesian statistics [13] to provide an improved estimate of the weighted mean. By adapting the work of Gauvain [14] we develop a statistics accumulation layer that (i) regresses to the prior mean when a small *sample mass* is admitted through the gating mechanism and (ii) adapts toward the mean of the weighted input data as counts increase.

The remainder of the paper is organized as follows: Section 2 discusses the progression from maximum-likelihood based statistics to include Bayesian statistics for attention modeling. Section 3 describes the experimental setup and results and Section 4 wraps up with the conclusion.

This work was supported in part by the Air Force Research Laboratory (AFRL) under contract FA8750-16-C-0088. The views, opinions, findings and recommendations contained in this article are those of the authors and do not necessarily reflect the official policy or position of AFRL. DISTRIBUTION A. Approved for public release: distribution unlimited. Case Number 88ABW-2019-0001 2019 01 02

[†]This work was completed while Jason Pelecanos was at IBM.

2. BAYESIAN STATISTICS WITH ATTENTION MODELING

In this section, we explore the use of a different intermediate layer in the neural network; a (B)ayesian (AT)tention or BAT layer. We discuss the basic attention layer approach and later extend this with a Bayesian interpretation.

2.1. Frame-Weighted Mean Layer

The work proposed by Salman [4] and more recently by Snyder [5, 6] utilized an intermediate frame based averaging layer to collect statistics over the period of a segment. The strength of the approach is that information about the speaker is accumulated in a stable manner over the duration of the segment or recording. The drawback is that every speech frame is treated to be equally informative across all network nodes in that layer. The standard frame based average layer, at any time step t, across observations $x_1, \ldots, x_t$, may be indicated as follows:

$$\mu_t = \frac{\sum_{i=1}^{t} x_i}{t} = \frac{\sum_{i=1}^{t} x_i}{\sum_{j=1}^{t} 1} \tag{1}$$

To remedy this issue, a frame weighted average can be calculated. Two such examples in the literature include the following [11, 12]. More specifically, for a single node, the output μ_t of a node at time sample t, is given:

$$\mu_t = \frac{\sum_{i=1}^{t} \eta_i x_i}{\sum_{j=1}^{t} \eta_j} \tag{2}$$

Here η_i is the frame/node importance. For standard attention models it is typically constrained to sum to one over a speech segment. In this work, instead of constraining the sum to one, we calculate η_i as the output of a sigmoid function where bolded $\boldsymbol{w}^T$ is the transpose of a weight vector[1], $\boldsymbol{x_i}$ is the feature observation vector and b is its offset.

$$\eta_i = \frac{1}{1 + \exp -(\boldsymbol{w}^T \boldsymbol{x_i} + b)} \tag{3}$$

2.2. Frame-Weighted MAP Adaptation Layer

An important observation to note is that a weighted average can be potentially unreliable if the accumulation of the frame weights is small. We wish to have a layer that uses the frame weighted mean if the importance accumulation is large. If the importance accumulation is small then the system should rely on prior information (or a default value). We can use a Bayesian interpretation to achieve this. A version of this (in a Maximum-A-Posteriori estimation or MAP sense) was previously applied to Gaussian mixture models used in speech recognition [14] and speaker recognition [15, 16, 17]. We will use the formulation and apply it within the neural network framework. The MAP and posterior mean estimates of the mean are the same in this case and the result is denoted as μ_t^{adapt}.

$$\mu_t^{adapt} = \alpha_t \mu_t^{new} + (1 - \alpha_t)\mu_t^{old} \tag{4}$$

where

$$\alpha_t = \frac{c_t}{c_t + R} \tag{5}$$

$$c_t = \sum_{j=1}^{t} \eta_j \tag{6}$$

$$\mu_t^{new} = \frac{\sum_{i=1}^{t} \eta_i x_i}{\sum_{j=1}^{t} \eta_j} \tag{7}$$

Note that R is the relevance factor relating to the prior information.

By substituting the terms, this can be rewritten as:

$$\mu_t^{adapt} = \frac{\sum_{i=1}^{t} \eta_i x_i + R\mu_t^{old}}{\sum_{j=1}^{t} \eta_j + R} \tag{8}$$

If the parameters R and μ_t^{old} are to be learned for each node, this may be rewritten further using terms R_1 and R_2 instead. In summary, the *frame weighted MAP adaptation layer* is given as follows:

$$\boxed{\mu_t^{adapt} = \frac{\sum_{i=1}^{t} \eta_i x_i + R_1}{\sum_{j=1}^{t} \eta_j + R_2} \quad \text{where} \quad R_2 > 0} \tag{9}$$

On a per node basis, the parameters that need to be trained by back-propagation are R_1, R_2, $\boldsymbol{w}$ (or $\boldsymbol{U}$, $\boldsymbol{V}$ as per the footnote) and b. We note that extending the attention layer variant (Equation 2) to include Bayesian statistics (Equation 9) involves a relatively small increase in the number of parameters. Note that if R_1 and R_2 are set to zero, the equation resolves back to the original frame-weighted average calculation as shown earlier. i.e. only the new evidence is utilized. If there are no frames accumulated, the MAP adaptation resolves to the ratio of R_1 and R_2.

For stability purposes, we perform gradient clipping and we enforce R_2 to be greater than some small positive value (we use 10^{-4}). As a result, we use the following equation in our software implementation:

$$\boxed{\hat{\mu}_t^{adapt} = \frac{\sum_{i=1}^{t} \eta_i x_i + R_1}{\sum_{j=1}^{t} \eta_j + |R_2| + c} \quad \text{where} \quad c = 10^{-4}} \tag{10}$$

[1] The weight vector is part of a weight matrix used to produce the output of an entire layer of nodes. This weight matrix can be formulated with fewer parameters by representing the weight matrix $\boldsymbol{W}$ as $\boldsymbol{U}\boldsymbol{V}^T$. Here, $\boldsymbol{U}$ and $\boldsymbol{V}$ are two low rank rectangular matrices with a smaller dimension of 200 in our experiments.

3. EXPERIMENTS

3.1. System Overview

To evaluate our system, we use a deep neural network structure as described by Snyder [5, 6] and adapt his implementation provided in the Kaldi toolkit [18] to evaluate this work. The network has 9 stacked layers with the first 5 layers calculating statistics at the frame-based level and the remaining 4 layers involving segment level statistics. The 5 layers of the frame-based components are ReLU based layers. Each input of the first 3 layers is formed by stacking nearby frames from the output of the previous layer. The next two layers use the input from the previous layers directly. The frame-based mean and standard deviation statistics (each of dimension 1500) are then calculated from the frames. These recording level statistics feed into two intermediate 500-node ReLU layers (known in Snyder's paper as speaker embedding layers "a" and "b") and are finally passed through a linear layer followed by a softmax layer to produce the speaker posteriors. The output from the two *intermediate* layers is then separately processed using LDA to reduce the dimensionality to 150. This is followed by a PLDA scoring framework [19]. The resulting scores can be combined as "a" + "b".

For the purpose of contrasting the different systems, we switch out the mean and standard deviation statistics layer for other types of components. Here we explore 3 variations on a particular layer. For simplicity of the Bayesian estimate in this paper, we assess an intermediate layer that estimates frame-based weighted posterior information for a concatenation of x_t and x_t^2 statistics[2]. To compare with Snyder's work [5, 6] we include results using mean and standard deviation (identified as [Mean (x), SD (x)] in the experiments). However, we consider our baseline system to be based on the mean of x and x^2 [Mean (x, x^2)]. This can be extended into one form of self-attention layer according to Equation 2 to perform neuron/node specific frame-weighted averaging [ATT (x, x^2)]. This component is then modified to encompass prior information in the form of Bayes with Attention [BAT (x, x^2)].

3.2. Data Sets

Here we explored 2 categories of data for the training of a system. The first we call the NIST related data. This is the data that NIST [7] encourages its participants to use. The training data consists of admissible audio from NIST 2005-2010 and the LDC Switchboard data prior to 2010. This data is composed of mostly English speech data recorded under conversational telephony conditions. The second portion of training data is a combination of NIST data and data extracted from the OpenSLR corpus [20] and the VoxCeleb Corpus [21].

[2]For the interested reader, posterior statistics can also be determined for standard-deviation/variance (see the relevant work in [14, 17, 13] for more information).

Table 1. EER performance for four types of layers with training on the NIST related data.

Intermediate Layer	a	b	a+b
Mean (x), *SD* (x)	*11.0*	*8.3*	*7.8*
Mean (x, x^2)	10.4	9.3	8.2
ATT (x, x^2)	9.2	8.9	8.0
BAT (x, x^2)	**8.6**	**7.9**	**7.5**

The OpenSLR corpus is extracted from the LibriVox [22] website where volunteers contribute their voice talent to produce public domain audio books. The volunteers, known as Readers, record themselves on various devices speaking sections of books. The VoxCeleb Corpus [21] represents a collection of YouTube audio speech segments spoken by different celebrities. The majority of the celebrities chosen are from the United States. The systems proposed are evaluated on the NIST 2010 10-second, 10-second condition which consists of a single side of conversational telephony speech.

3.3. Results and Discussion

For the first set of NIST 2010 10second-10second experiments we assess the utility of different frame accumulation layers using standard NIST training data sets. These data sets conform to the NIST 2010 specification. Tables 1 and 2 show the EER and MinDCF10 [7] performance measures for each technique. The first result shows the performance numbers when Snyder's system (using a mean and standard deviation statistics accumulation layer) is reevaluated. It achieves an EER of 7.8% which is comparable to the result in Snyder's paper of 7.9% [5]. The next line represents the baseline system which calculates the mean of a set of first and second order statistics. The results suggest (compare 8.2% to 7.8%) that this baseline may not be as strong as the mean and standard deviation layer. However, it does provide a suitable baseline for demonstrating the evolution of the improvements to the model. The next row presents the results of the self attention model. While it does improve the EER, the minDCF result is inconsistent. The final row includes the results for the Bayes with attention component. This approach shows improvement across all EER values and in two of the three minDCF cases (i.e. for cases, "b" and "a+b").

There is a large effort to improve speaker recognition on shorter recordings. One approach is to use better modeling techniques and another is the utilization of additional speech data. In this experiment we explore the intersection of the modeling techniques discussed in this paper and the inclusion of additional data sets for training. In particular we examine the utility of including training data in addition to the standard NIST data.

In Tables 3 and 4 we present the results of the same four

Table 2. MinDCF10 performance for four types of layers with training on the NIST related data.

Intermediate Layer	a	b	a+b
Mean (x), *SD* (x)	*0.94*	*0.88*	*0.88*
Mean (x, x^2)	**0.96**	0.99	0.96
ATT (x, x^2)	**0.96**	0.97	0.97
BAT (x, x^2)	0.99	**0.93**	**0.90**

Table 3. EER (%) performance for the training set configuration consisting of the NIST related data, OpenSLR and VoxCeleb.

Intermediate Layer	a	b	a+b
Mean (x), *SD* (x)	*8.9*	*7.9*	*6.6*
Mean (x, x^2)	8.9	9.7	8.0
ATT (x, x^2)	**8.6**	**7.7**	**6.8**
BAT (x, x^2)	8.8	8.0	7.5

approaches now trained using the combination of three data sets; NIST, OpenSLR and VoxCeleb. For the (x, x^2) set of approaches, the standard attention model does well across multiple configurations. Interestingly, the BAT model obtains the best single MinDCF result of 0.85, but the overall performance on the mismatched training data is inconsistent and needs further study.

4. CONCLUSION

In this paper we proposed and explored the use of a Bayesian attention layer as a drop-in replacement for a frame average or attention layer. This layer is similar to an attention layer with the added flexibility of adapting to new information or backing-off to prior information. Experiments on the NIST 2010 10-second condition suggest some promising initial results especially on the relatively matched training condition. Opportunities for future work include (i) a better understanding of the properties of the network layer under diverse training and evaluation conditions, (ii) exploring additional frame weighting types other than sigmoid and, (iii) evaluating posterior statistics for standard deviation.

Table 4. MinDCF10 performance for the training set configuration consisting of the NIST related data, OpenSLR and VoxCeleb.

Intermediate Layer	a	b	a+b
Mean (x), *SD* (x)	*0.99*	*0.95*	*0.90*
Mean (x, x^2)	0.94	0.98	0.95
ATT (x, x^2)	0.93	**0.91**	**0.88**
BAT (x, x^2)	**0.85**	0.98	0.94

5. REFERENCES

[1] L. Heck, Y. Konig, M. Sönmez, and M. Weintraub, "Robustness to telephone handset distortion in speaker recognition by discriminative feature design," *Speech Communication*, vol. 31, no. 2-3, pp. 181–192, 2000.

[2] S. Yaman, J. Pelecanos, and R. Sarikaya, "Bottleneck features for speaker recognition," *Odyssey 2012: The Speaker and Language Recognition Workshop*, pp. 105–108, 2012.

[3] F. Richardson, D. Reynolds, and N. Dehak, "Deep neural network approaches to speaker and language recognition," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1671–1675, 2015.

[4] A. Salman, *Learning Speaker-Specific Characteristics with Deep Neural Architecture*, Ph.D. thesis, University of Manchester, 2012.

[5] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Interspeech*, 2017.

[6] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust embeddings for speaker recognition," *ICASSP*, 2018.

[7] NIST, "Speaker Recognition Evaluation 2010," *https://www.nist.gov/itl/iad/mig/speaker-recognition-evaluation-2010*, Accessed: October 2018.

[8] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," *ICASSP*, 2016.

[9] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," *ICASSP*, 2014.

[10] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *Proceedings of the 32nd International Conference on Machine Learning*, pp. 2048–2057, 2015.

[11] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification," *Interspeech*, 2018.

[12] F. Chowdhury, Q. Wang, I. Moreno, and L. Wan, "Attention-based models for text-dependent speaker verification," *ICASSP*, 2018.

[13] M. DeGroot, *Optimal Statistical Decisions*, McGraw-Hill Inc, New York City, New York, USA, 1970.

[14] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[15] J. Gauvain, L. Lamel, and B. Prouts, "Experiments with speaker verification over the telephone," *Eurospeech*, 1995.

[16] D. Reynolds, "Comparison of background normalization methods for text-independent speaker verification," in *Eurospeech*, 1997, vol. 2, pp. 963–966.

[17] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1/2/3, pp. 19–41, 2000.

[18] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[19] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," *ICCV*, 2007.

[20] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "LibriSpeech: An ASR corpus based on public domain audio books," *ICASSP*, 2015.

[21] A. Nagrani, J. Son Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," *Interspeech*, 2017.

[22] LibriVox, "LibriVox: Free public domain audiobooks," *http://www.librivox.org*, Accessed October 2018.