Discriminatively trained phoneme confusion model for keyword spotting

Panagiota Karanasou[1]*, Lukas Burget*[2]*, Dimitra Vergyri*[2]*, Murat Akbacak*[2]*, Arindam Mandal*[2]

[1] LIMSI/CNRS, Université Paris-Sud, BP133, 91 403 Orsay Cédex, France
[2]Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA
pkaran@limsi.fr, burget,dverg,murat,arindam@speech.sri.com

Abstract

Keyword Spotting (KWS) aims at detecting speech segments that contain a given query within large amounts of audio data. Typically, a speech recognizer is involved in a first indexing step. One of the challenges of KWS is how to handle recognition errors and out-of-vocabulary (OOV) terms. This work proposes the use of discriminative training to construct a phoneme confusion model, which expands the phonemic index of a KWS system by adding phonemic variation to handle the above-mentioned problems. The objective function that is optimized is the Figure of Merit (FOM), which is directly related to the KWS performance. The experiments conducted on English data sets show some improvement on the FOM and are promising for the use of such technique.

Index Terms: keyword spotting, confusion model, discriminative training, Figure of Merit

1. Introduction

As the amount of real-world spoken data rapidly increases, the ability to search it efficiently for particular words or phrases of interest gains more importance. KWS aims at searching audio data and detecting any given keyword, which is typically a single word or a short phrase. KWS systems build for off-line data processing usually operate in two phases: indexing and search. The system processes the audio data once, during the indexing phase, without knowledge of the query terms. This phase is done off-line and is the more time-consuming one. The output index is stored and accessed during the search phase, in order to locate the terms and link them to the original audio.

Word-level indexing may seem to be a straightforward solution, but it cannot handle OOV terms which are often named entities not covered by the automatic speech recognition (ASR) dictionaries. The OOV rate may increase with time, as dictionaries are usually fixed while the content of real-world data dynamically changes. Generating pronunciations for OOVs implies having a letter-to-sound system, which is often not accurate, especially for languages with limited resources. Augmenting the recognition system with pronunciation variants can help, but implies regenerating the index and may introduce confusability to the KWS system and increase the false alarms, especially if their weights are not properly tuned. Moreover, there are many applications where the performance of the word recognizer is severely degraded due to challenging audio conditions, and even in-vocabulary words are not successfully represented in the index. For these reasons, subword-level and particularly phonemic-based KWS systems have been used in the past [1], which do not impose any vocabulary restrictions.

This work started while the author Panagiota Karanasou was at Speech Technology and Research Laboratory in SRI International.

In the current work, a phoneme index is created from lattices generated by a phoneme recognizer, which allows us to preserve information about phonemic uncertainty of the phoneme recognition in the index. The phonemic index allows detecting any word without constraining the system to the in-vocabulary terms. However, the quality of the phoneme recognizer can be quite low, especially when dealing with data recorded under noisy or mismatched conditions. For this reason, a phoneme confusion model is introduced in this work. Its goal is to predict the deviation of the phoneme recognition output compared to the true spoken phonemes. Once applied on the index, this confusion model acts as a corrector of the recognition errors.

Our confusion model expands the index with alternative phoneme sequences, which inherently introduce additional detections when searching queries. This can be beneficial, especially if the space limitations forces us to significantly prune the phoneme lattices to keep the index to a reasonable size. However, KWS performance optimization is a counterbalancing procedure of increasing the true detections while keeping the false alarms low. To address this problem, we train the parameters of the confusion model discriminatively, where the optimized objective function is the Figure of Merit (FOM), a well-established evaluation metric of KWS performance. The FOM was also used in [2] to directly optimize the weights of the index, which had the form of a matrix of probabilistic acoustic scores. In [3], it was used to optimize an interpolation factor when alternative pronunciations were added for OOVs. To our knowledge this is the first time it is used to train the weights of a phoneme confusion model for KWS.

The rest of the paper is organized as follows. In Section 2 the KWS system is described. In Section 3 the discriminative training of the phonemic confusion model is explained. In Section 4 the experimental setup is documented, and in Section 5 the results are presented. The paper concludes in Section 6 with some discussion and future work plans. The reader is considered to be familiar with the basic concepts of the theory of finite state transducers for reading Sections 2 and 3. For more details, he is referred to [4].

2. Keyword spotting system

2.1. Indexing and searching representation

As already mentioned, our KWS system operates in two phases: indexing and search. In our work, the query terms are phoneme sequences and the index is constructed from a set of phoneme lattices generated for each input utterance using a phoneme recognizer. As proposed in [5], the index is represented as a weighted finite state transducer (WFST) allowing very efficient search for queries. The index WFST is constructed in such a

way that every subsequence of phonemes found in any path of any phoneme lattice is represented by exactly one successful path. For such path, the sequence of input symbols corresponds to the subsequence of phonemes and the output sequence identifies the lattice/utterance. The index WFST is built on the lexicographic semiring so that each path weight is a triple representing the start time, the end time and the log posterior probability of the phoneme subsequence appearing in the lattice at this time. For more details on constructing the index WFST, we refer the reader to [5].

The query terms are represented as weighted finite state acceptors (WFSA). In the simple case, where the query is a single phoneme sequence, the acceptor is a linear sequence of arcs with input symbols representing the corresponding phonemes. The acceptor can have, however, more complicated topology, where its individual paths represent a set of query terms we want to search for. The acceptor can also represent multiple pronunciation variants of a query word. In this last case, the weight of each path represents the (log) probability of the corresponding pronunciation variant.

Let the index and the query be represented by transducer I and acceptor Q, respectively. The search can be performed by composing the two automata $Q \circ I$ and sorting the paths through the resulting transducer with the shortest-path algorithm. Again, just like in the case of the original index WFST, each path through the composed transducer encodes information about the phoneme sequence, the lattice/utterance it was detected in and the timing. The composed transducer, however, contains only the phoneme sequences represented by Q. All the FST manipulations are realized using the Openfst libraries [6].

2.2. Confusion model

Now, we want to take into account the fact that the query phoneme sequence can get assigned wrong posterior probability in the index (e.g. because of a systematic error of the phoneme recognizer) or is completely missing from the index (e.g. because of lattice pruning needed to obtain an index of a reasonable size). For these reasons, a confusion model is introduced reflecting our assumption that a phoneme sequence can get (with a certain probability) misrecognized as a different sequence, which we may wish to search for instead. In this work, we consider only simple context independent confusion model, assuming that each phoneme can get with a certain probability inserted, deleted or substituted by another phoneme. The confusion model is represented as a WFST on the tropical semiring with all arcs looping in a single state. For each arc, the weight $w(i, o)$ can be interpreted as the log probability that the phoneme represented by the input symbol i gets misrecognized as the output symbol o. An empty symbol ϵ can be used as input symbol to represent insertion or output symbol to represent deletion.

A query transducer Q can be now composed with a given confusion model C to obtain an expanded query transducer $Q \circ C$. For each path through $Q \circ C$, the output symbols represent the phoneme sequence we want to search for in the index instead of the original query (input symbol sequence). The weight of the path then represents the assumed probability that the output sequence (if found in the index) is in fact the misrecognized input sequence. Finally, the search of the expanded query in the index I can be performed by composing all three transducers $Q \circ C \circ I$. Alternatively, this composition can be seen as searching the original query Q in the expanded/corrected index $C \circ I$. Again, each path through the final transducer $Q \circ C \circ I$ can be interpreted as the detection of a query phoneme sequence in a particular lattice/utterance at a particular time. The weight of the k-th such path (e.g. as obtained from shortest-path algorithm) defines the detection score s_k. Note that the score combines the weight w_k of the confusion model C and the weight (log posterior) n_k of the index I as

$$s_k = w_k + n_k, \tag{1}$$

which naturally expresses that the final score is the product (sum in log domain) of two probabilities: 1) probability of misrecognizing the original query as a different phoneme sequence and 2) probability of detecting that sequence in the index. We can further decompose the confusion model's path weight into the contributions of the individual phonemes j as

$$w_k = \sum_j w(i_{kj}, o_{kj}). \tag{2}$$

It should be noted that the confusion model's output symbols o_{kj} are not visible in the final transducer $Q \circ C \circ I$, as these are consumed by the composition of C and I. However, for the discriminative training, as will be detailed below, it is still needed to keep track of all the contributing scores $w(i_{kj}, o_{kj})$ for each detection.

2.3. Confusion model initialization

The weights of the confusion model transducer $w(i, o)$ are the discriminatively trained parameters as will be described in Section 3.2. To get reasonable initial values for the weights, we obtain the one-best phoneme recognition output from our training corpora, align it with the reference phoneme sequence, and count the number of phoneme specific insertions, deletions and substitutions. The weights are thus initialized as follows: For the insertion of the output phoneme o, the weight (log probability) is set as

$$w(\epsilon, o) = \log \frac{\text{\# of insertions of phoneme } o}{\text{\# of all recognized phonemes}}. \tag{3}$$

All other weights are initialized as

$$w(i, o) = \log \left(P(\text{non-ins}) \frac{\text{\# of } i \text{ recognized as } o}{\text{\# of } i \text{ in the reference}} \right), \tag{4}$$

where the probability of the output symbol not being inserted

$$P(\text{non-ins}) = 1 - \frac{\text{\# of all insertions}}{\text{\# of all recognized phonemes}} \tag{5}$$

and where $o = \epsilon$ corresponds to phoneme deletion.

3. Confusion model training

3.1. The Figure of Merit

As the criterion for the discriminative training of the confusion model's parameters we use the FOM. The FOM is defined as the detection rate averaged over the range of 0 to 10 false alarms per hour and over the individual queries [7]. Equivalently, it can be interpreted as the normalized area under the Receiver Operating Characteristic (ROC) curve in that false alarm range. To evaluate FOM for a given data set and a given set of queries, we enumerated detections by applying the shortest-path algorithm to $Q \circ C \circ I$ transducer. Based on the reference transcription, the detections are assigned into two sets: set of true detections R^+ and set of false alarms R^-. In accordance with the FOM

definition, only the top scoring detections are assigned to the sets R^+ and R^- corresponding to 10 false alarms per hour and query term. More precisely the top scoring detections are selected, so that $|R^-| \approx A = 10|Q|T$, where $|Q|$ is the number of query terms and T is the number of hours of speech. The FOM is defined as

$$FOM = \frac{1}{A} \sum_{j \in \mathbb{R}^-} \sum_{k \in \mathbb{R}^+} h_k H(s_k, s_j), \tag{6}$$

where s_k is the score for detection k as defined by equation (1), $h_k = 1/(|Q|Occ(k))$, $Occ(k)$ is the number of true occurrences of the query term corresponding to detection k in the reference transcript, and

$$H(s_k, s_j) = \begin{cases} 1 & s_k > s_j \\ 0 & otherwise. \end{cases} \tag{7}$$

In accordance with the FOM definition, this formula can be interpreted as a numerical integration over the ROC curve: Let the outer sum, performed over false detections j, be sorted by decreasing scores s_k. Then, the sum can be seen as an integration over the false alarm rate axis (i.e. each detection j corresponds to one more false alarm) with an appropriate integration step $1/A$ corresponding to the axis scale. Therefore, each j represents a certain false alarm rate obtained with the score threshold s_k. For a given false alarm rate j, the inner sum can be interpreted as the corresponding detection rate (averaged over queries) calculated as the appropriately normalized number of true detections exceeding the threshold s_k.

As can be seen from the formula, the FOM can be also interpreted as a metric testing the discriminative power of the detection scores by expressing the KWS as the problem of ranking hits above false alarms.

3.2. Discriminatively optimizing the Figure of Merit

Since the FOM is not a continuous differentiable function, which is required for our optimization, we closely approximate it as

$$f = \frac{1}{A} \sum_{k \in \mathbb{R}^+} h_k \sum_{j \in \mathbb{R}^-} \varsigma(s_k, s_j) \tag{8}$$

$$\varsigma(s_k, s_j) = \frac{1}{1 + exp(-\alpha(s_k - s_j))}, \tag{9}$$

where the step function $H(s_k, s_j)$ is approximated by a sigmoid $\varsigma(s_k, s_j)$ [8]. The tunable parameter a is set to $a = 1$ for this work. Note also that, compared to equation (6), we have switched the order of the two sums to allow a more efficient evaluation.

The confusion model parameters are trained to maximize the objective function f on the training data and using the training queries described in Section 4. For the optimization, we use a simple gradient descent algorithm with a fixed step, which requires the evaluation of the derivatives

$$\frac{\partial f}{\partial w(i,o)} = \frac{\alpha}{A} \sum_{k \in \mathbb{R}^+} h_k \sum_{j \in \mathbb{R}^-} \varsigma(s_k, s_j)(1 - \varsigma(s_k, s_j)) \left[\frac{\partial s_k}{\partial w(i,o)} - \frac{\partial s_j}{\partial w(i,o)} \right], \tag{10}$$

where $\partial s_k / \partial w(i,o)$ is simply the number of times the weight $w(i,o)$ occurs in the sum of equation (2).

Intuitively, in order to increase the FOM, the weights $w(i,o)$ should change such that the scores of hits increase with respect to the scores of false alarms. Note that when optimizing the weights we do not put any constraint on them, so that they may not correspond to any probabilistic model in the end.

4. Experimental setup

Experiments on English data were conducted. For the phoneme recognizer, acoustic models developed for the RT-04 evaluation, which were also used for the SRI STD-06 submission for the broadcast news task [9], were applied. In particular, gender independent cross-word triphone PLP models were trained for a set of 45 phones (including pause, reject and two hesitation specific phones). 13 PLP coefficients plus 1st, 2nd and 3rd order derivatives were used, while cepstral mean and variance normalization, vocal tract length normalization and HLDA to reduce dimentionality to 39 were applied. Decision tree state clustering was used to cluster the triphone states to about 2500 states, and 200 Gaussians per state were trained using 5 MLE training iterations and 4 discriminative (alternating MPE-MMIE) training iterations. The training data were LDC distributions including: Hub4 1996 and 1997 (200h), TDT4 (275h) TDT2 (272h) and BNr1234(2300h). The output lattices were pruned in a preprocessing step before the index construction.

The data set for the discriminative training of the confusion model's weights consisted of 8 hours of broadcast news (BN) data. The KWS performance of the proposed technique was evaluated on the NIST STD06 evaluation [10] set consisting of 3 hours of data. Two disjoint sets of query terms were used for training and evaluation. From the training data, 1217 query words were extracted and translated to their phonemic transcriptions. The transcriptions longer than 3 phonemes were kept. To avoid biased results, the 100 most frequent words but also the words that occurred less than 5 times were not selected. The evaluation queries were the same 1104 terms that were used for the NIST STD06 evaluation.

5. Results

The performance metric adopted in this work is the FOM (see Section 3.1), which is the one we try to optimize. The results will be presented in terms of the ROC curves showing the system performance for different operation points.

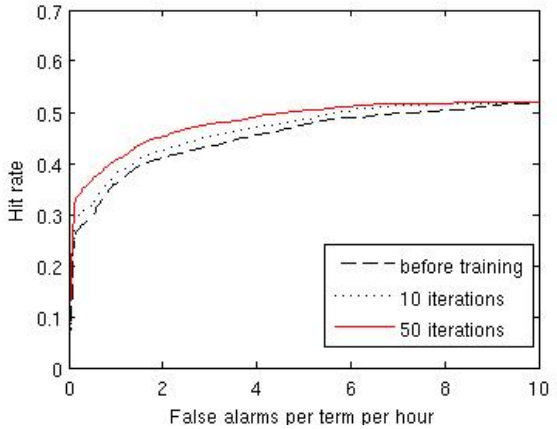

Figure 1: *ROC curves on training data*

The baseline system is the one, where the KWS search is realized on the index before applying the confusion model and expanding the search space. The initialized confusion model is then applied and discriminatively trained to maximize the FOM on the training data. Figure 1 presents the ROC curves on the training data before training, after 10 iterations and after 50 iterations of the training algorithm. It can be seen that the area under the ROC curve is indeed increased.

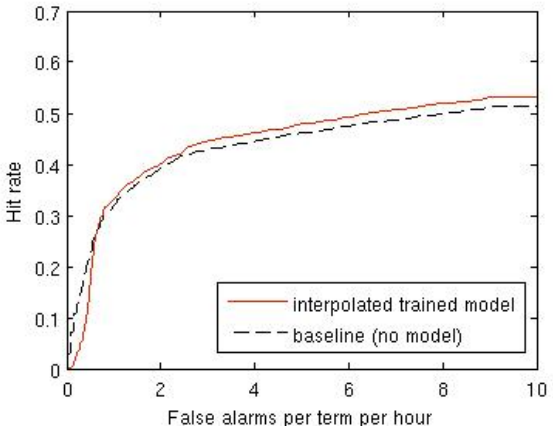

Figure 2: *ROC curves on evaluation data*

We have found that it is beneficial to smooth the trained confusion model before applying it to the evaluation data. The smoothing chosen here is the linear interpolation of the trained confusion model weights with a null model, which considers zero probabilities for substitutions, deletions and insertions and probabilities one for correct phoneme recognitions. The interpolation factor was set to 0.5. The curve for the smoothed trained model in Figure 2 corresponds to this interpolated model. Some degradation is seen for the low false alarms region (less that 1 false alarm per hour per query term), but then some improvement is observed, which increases for high false alarms numbers. For this area, if a random horizontal line is drawn in Figure 2, it can be actually seen that there is achieved an important decrease in the false alarms number for the same hit rate when the smoothed trained model is applied. For example, for a hit rate of 0.45, the false alarms number decreases from 5 (baseline curve) to 2.5 (interpolated trained curve) false alarms per hour and per query term. It should be noted that the value of the baseline performance is fairly acceptable for a phonemic KWS system.

6. Conclusion and Future work

We have presented a phoneme confusion model for the KWS that allows recovery from recognition errors, and enables detection of OOVs. A discriminative approach for training its weights was applied based on the direct optimization of the FOM. The approach was tested for English. However, it is language-independent and could be applied to other languages, potentially including languages with limited resources where the OOV problem is more extensive. In terms of FOM performance a promising improvement was observed on the evaluation set.

The confusion model is applied on the index constructed using the output lattices of a phone-loop recognizer. In the future we plan to apply it also to hybrid systems that use both word and phoneme recognition. The confusion model used in this work does not take into account any phoneme context. It is our aim to try to use at least bigram phoneme confusion models and expect to achieve better KWS results. Our aim is also a better initialization of the confusion model and we have already started working in this direction. In addition, other more complex methods to train the parameters of our model could be investigated during the FOM optimization. Last but not least, currently the confusion model just add bias to the posterior scores. Instead, more complicated confusion models could be developed that operate directly on the acoustic scores from which the posteriors are computed. In this case, the confusion model could represent a multiplicative or additive correction to the acoustic scores.

7. Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. D10PC20024. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA; or its Contracting Agent, the U.S. Department of the Interior, National Business Center, Acquisition & Property Management Division, Southwest Branch. This work is also partly realized as part of the Quaero Programme, funded by OSEO, the French State agency for innovation.

8. References

[1] Saraclar, M., "Lattice-based search for spoken utterance retrieval", in Proc. of HLT-NAACL, 129–136, 2004.

[2] Wallace, R., Baker, B., Vogt, R. and Sridharan,S., "Discriminative optimisation of the figure of merit for phonetic spoken term detection", IEEE Trans. on Audio, Speech and Language Proc., 19(6):1677–1687, 2010.

[3] Wang, D., King, S. and Frankel, J.,"Stochastic Pronunciation Modelling for Spoken Term Detection", in Proc. of Interspeech, 2009.

[4] Mohri, M., Pereira, F. and Riley, M., "Weighted finite-state transducers in speech recognition", Computer Speech & Language, 16(1):69–88, 2002

[5] Can, D. and Saraclar, M., "Lattice Indexing for Spoken Term Detection", IEEE Trans. on Audio, Speech and Language Proc., 19(8):2338–2347, 2010.

[6] Allauzen, C., Riley, M., Schalkwyk, J., Skut, W. and Mohri, M., "OpenFst: A General and Efficient Weighted Finite-State Transducer Library", in Proc. of CIAA, 4783:11–23, 2007.

[7] Rohlicek, J.R., Russell, W., Roukos, S. and Gish, H.,"Continuous hidden Markov modeling for speaker-independent word spotting", in Proc. of ICASSP, 627–630, 1989.

[8] Raykar, V., Duraiswami, R. and Krishnapuram, B., "A fast algorithm for learning large scale preference relations", in AISTATS, 2:388–395, 2007.

[9] Vergyri, D., et al, "The SRI/OGI 2006 Spokem Term Detection System", in Proc. of Interspeech, 2393–2396, 2007.

[10] NIST, "The Spoken Term Detection (STD) 2006 Evaluation Plan", 2006. Online: http://www.nist.gov/speech/tests/std/docs/std06-evalplan-v10.pdf.