NORMALIZATION OF PHONETIC KEYWORD SEARCH SCORES

Damianos Karakos, Ivan Bulyko, Richard Schwartz, Stavros Tsakalidis, Long Nguyen, John Makhoul

Raytheon BBN Technologies, Cambridge, MA, USA
email: {dkarakos,schwartz,ibulyko,stavros,ln,makhoul}@bbn.com

ABSTRACT

As shown in [1, 2], score normalization is of crucial importance for improving the Average Term-Weighted Value (ATWV) measure that is commonly used for evaluating keyword spotting systems. In this paper, we compare three different methods for score normalization within a keyword spotting system that employs phonetic search. We show that a new unsupervised linear fit method results in better-estimated posterior scores, that, when fed into the keyword-specific normalization of [1], result in ATWV gains of 3% on average. Furthermore, when these scores are used as features within a supervised machine learning framework, they result in additional gains of 3.8% on average over the five languages used in the first year of the IARPA-funded project Babel.

Index Terms— Keyword search, keyword spotting, speech indexing, score normalization, phonetic matching

1. INTRODUCTION

Keyword spotting from speech is the task of determining if a word or phrase has been uttered. In most cases, it consists of performing some kind of speech recognition, and then searching the resulting output space of alternatives (e.g., lattices or n-best lists) for the keywords of interest.

Of special interest is the case where the keywords contain one or more words which are out-of-vocabulary (OOV). Such cases are especially challenging whenever the speech is "pre-indexed", which means that it has been previously processed by the ASR system using a specific pronunciation lexicon, without knowledge of the set of queries that one may want to search for. Therefore, queries which contain new or rare words (e.g., named entities) that were not part of the original pronunciation lexicon, will necessarily not be detected when using whole words as the unit of recognition or search.

To alleviate the above problem, a number of alternatives have been proposed. One of them is to use *phonetic search*. This means that query terms are converted into their phonetic representation and they are searched in a similarly-represented version of the hypothesis space at the output of the recognizer. Of course, instead of phones, one can choose to use some other representation that will achieve the goal of covering a large class of OOV words. The procedure for doing the phonetic search in this paper follows closely the one presented in [3].

To evaluate a keyword spotting system, we focus on maximizing the ATWV measure [4]. This measure, which trades off misses for false-alarms, assumes that keyword detections (also known as "hits" or "posting lists") are sorted into a global list according to their detection score. This means that all keyword scores have to be commensurate with each other, in order for the sorting of the global list to reflect correctness, irrespective of the identity of the keyword. The solution to this problem is to do some kind of "score normalization" [1] so that the resulting transformed scores are better correlated with the probability of correctness.

In [1] several normalization approaches were compared. Specifically, the unsupervised KST method, which entails computing keyword-specific thresholds and then using them in an exponential formula (presented in Section 2) resulted in very good performance, comparable to the supervised machine learning approach that was described in the same paper. The machine learning approach was tuned towards optimizing the performance measure, namely ATWV, using lots of features as input.

In this paper, we show that the output of phonetic search can benefit significantly by a linear fit method which aims at transforming the raw posteriors into an estimated probability of correctness, based on a Poisson assumption. These new scores are then shown to be much more appropriate for use in the KST method, resulting in gains of the order of 3% on average. Furthermore, the supervised learning methods of [1], which use the above score as feature, give additional gains of the order of 3.8% on average. These results prove that using well-conditioned scores that resemble posteriors can be very beneficial for both supervised and unsupervised normalization methods.

The paper is organized as follows: an overview of the performance measure, as it is used in the IARPA-funded program Babel, appears in Section 2. The KST normalization method is summarized, and a theoretical justification for using the expected count in place of the true number of references is presented in the same section. The unsupervised linear fit method is presented in Section 3. The machine learning method of [1] is briefly summarized in Section 4. Experimental results appear in Section 5 followed by conclusions in Section 6.

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

2. THE ATWV MEASURE AND ITS MAXIMIZATION WITH KEYWORD-SPECIFIC THRESHOLDS

Keyword spotting systems typically contain a speech-to-text engine that converts raw waveforms into a searchable form, such as word lattices. The probability of each detection's correctness is estimated directly from the lattices, and an index containing a precomputed list of candidate detection records (hits) for each word is generated. The index also contains phonetic transcripts to accommodate out-of-vocabulary search terms.

For each search query term (which can be a single word or a multi-word string) the generated list of detection records is sorted according to a detection score. A decision function uses a threshold and all records with scores above the threshold are postulated to be present.

Accuracy is judged relative to a time-marked reference transcript. A system detection is considered correct if a corresponding exact orthographic match of the term appears in the reference transcript within 0.5 seconds of the asserted time.

System accuracy on a given collection of query terms is measured by the Actual Term-Weighted Value (ATWV) metric, defined in [4] as

$$\text{ATWV} = 1 - \frac{1}{K}\sum_{w=1}^{K}\left(\frac{\#\textit{miss}(w)}{\#\textit{ref}(w)} + \beta\,\frac{\#\textit{fa}(w)}{T - \#\textit{ref}(w)}\right) \quad (1)$$

where K is the total number of keywords with reference tokens, $\#\textit{miss}(w)$ is the number of true tokens of keyword w that are not detected, $\#\textit{fa}(w)$ is the number of false detections of w, $\#\textit{ref}(w)$ is the number of reference tokens of w, T is the total number of trials (e.g., seconds in the audio), and β is a constant, set at 999.9.

Note that ATWV is a function of the threshold used in deciding whether a detection exists or not. The Maximum Term-Weighted Value (MTWV) is then defined as the maximum ATWV over all decision thresholds.

In [5], a formula is presented for computing a decision (Yes/No) for each detection, based on whether its posterior score is above the keyword-specific threshold

$$\textit{thr}(w) = \frac{N_{\text{true}}}{T/\beta + \frac{\beta-1}{\beta}N_{\text{true}}} \quad (2)$$

where N_{true} is the number of true tokens of keyword w which exist in the audio.

In the absence of true transcripts, $N_{\text{true}}(w)$ can be approximated by the expected count for that keyword:

$$N(w) = \sum_{j=1}^{M} p_j, \quad (3)$$

where M is the number of detections for keyword w and p_j is the posterior for the j-th detection. As we show in the rest of this section, this approximation can be justified by formulating the problem as minimizing the *expected* Bayes risk.

Under a Bayes formulation, the number of reference tokens of a keyword is considered to be a random variable (whose distribution can be approximated by an appropriate summation of the posterior scores $\{p_j\}_{j=1}^{M}$). Then, if the objective is to minimize the expected Bayes risk, where the expectation is with respect to the distribution of the number of reference tokens, the loss function becomes

- $E[L(\textit{miss}; N_t)] = E\left[N_t^{-1}\right]$, where N_t is the random number of references;
- $E[L(\textit{fa}; \beta, T, N_t)] = \beta E\left[(T - N_t)^{-1}\right]$, where T is the audio duration.

With the above modification to the Bayes risk function, it can be shown that the value of $\textit{thr}(w)$ that minimizes the expected Bayes risk is

$$\textit{thr} = \left(E[N_t^{-1}]/\beta E[(T - N_t)^{-1}] + 1\right)^{-1}. \quad (4)$$

To compute the threshold in (4) in practice, one can use the posteriors $\{p_j\}_{j=1}^{M}$ in a combinatorial expression. Specifically, given that each p_j is an estimate of the probability that the j-th hit is a true positive, the product

$$\prod_{j=1}^{M} p_j^{b_j}(1 - p_j)^{1-b_j}, \;\; b_j \in \{0, 1\},$$

expresses the probability that only the hits, whose corresponding "bits" b_j are equal to 1, are the locations of the true positives, while the rest of the hits are false alarms. Hence, by summing together all these probabilities, and under the condition that we have at least one reference token (as required by the ATWV formula), we obtain the distribution of N_t:

$$\Pr[N_t = n | N_t \geq 1] = \frac{\sum_{\{b_j\}:\sum b_j = n}\prod_{j=1}^{M} p_j^{b_j}(1 - p_j)^{1-b_j}}{1 - \prod_{j=1}^{M}(1 - p_j)}.$$

In our experiments we have found that using (2), with the expected count in the place of N_{true}, gives almost identical results as (4) which suggests that the decision threshold is quite accurately estimated.

The keyword-specific thresholds (KST) are used in this paper for normalizing the scores across all keywords in such a way that the decision threshold becomes a constant. Specifically, the formula for transforming the posterior p of a keyword w has the exponential form

$$p' = p^{\left(-\frac{1}{\log(\textit{thr}(w))}\right)}, \quad (5)$$

The constant decision threshold then becomes $1/e = 0.368$. The KST normalization method, mentioned in the rest of the paper, refers to the above formula.

3. UNSUPERVISED SCORE NORMALIZATION USING LINEAR FIT

In this section we introduce a novel pFA-based score normalization method, where we estimate the false alarm rate for a given keyword at various levels of the raw score (log probability of the DP alignment between the query and the recognition output represented by a phonetic consensus network – see [3] for details). We model the mapping from the raw log probability to the log of FA rate with a straight line, computed for each keyword separately. Note that we only need examples of false alarms to estimate this model, and the estimation is completely unsupervised if we assume that the keywords of interest are rare and the vast majority of the returned hits are false alarms (which we do in these experiments).

We found that fitting a line to the data points in log space tends to underestimate the FA rate (overestimate the confidence) of high-scoring hits, because of the high concentration of data points with low scores (hits that score poorly). We can get a better estimate of the FA rate if we fit to data points that are uniformly sampled on the log scale (Figure 1).

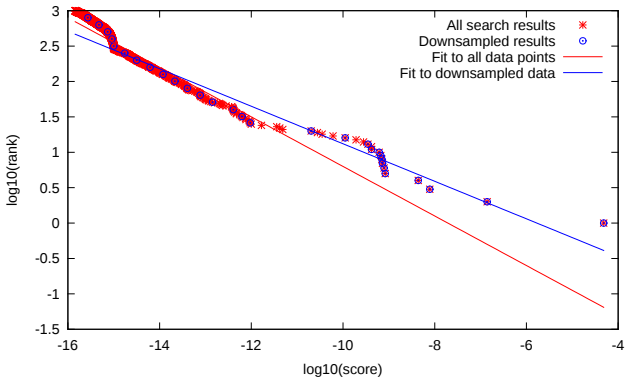

Fig. 1. Relationship of log10 of raw scores of search results for one query and log10 of the rank (we consider top 1000 hits). The blue line fitted to uniformly sampled data based on log10(rank) better captures the overall trend in log space.

We fit the line f to satisfy $f(\log(p)) \approx \log_{10}(\text{rank})$, where p is the posterior that results from the phonetic search of a keyword, and rank is the rank of the detection of that keyword.

Given a search result with the raw log probability $\log(p)$ we compute the expected rank of such result:

$$\text{expected rank} = 10^{f(\log(p))},$$

This corresponds to the false alarm rate for this result, e.g., if the expected rank is 5 than we expect 5 false alarms at this level p of posterior score. The number of false alarms is modeled with the Poisson random variable, which gives us an estimate for the probability that a certain number of events (i.e., false alarms) will be observed given the average rate λ, where $\lambda =$ expected rank:

$$\Pr(\text{number of FAs} = n) = \frac{\lambda^n \, e^{-\lambda}}{n!}.$$

What we are actually interested in is that the probability that this search result is correct (i.e., it is not a false alarm), which is equal to the probability that we do not observe any of the false alarms. Hence,

$$\Pr(\text{number of FAs} = 0) = e^{-\lambda} = e^{-\text{expected rank}}.$$

Using this formulation, we get good estimates for the confidence values that are comparable across search terms.

4. SUPERVISED SCORE NORMALIZATION USING MACHINE LEARNING

The score normalization method of [1] is based on a machine-learning framework that utilizes many features. The posterior scores scores go through a number of transformations, such as: rank-normalization (a generalization of [6]), mapping-back to posteriors, "probability of correct" normalization $p_{corr}()$, as well as non-linear functions such as $log()$, $()^{1/2}$, $()^2$, sigmoid. The $p_{corr}()$ mapping aims at transforming the scores so that they better correspond to the probability of correctness. (One way of doing that is by sorting all hits by score, defining bins, and then computing the probability that a random detection in the bin is correct.) Then, the transformed scores, together with various additional features, e.g., keyword training count, keyword length, conversation-aggregated scores, are concatenated together into a feature vector. This vector, together with a target variable denoting whether the detection is a true positive or a false alarm, is given as input to Powell's method [7], which learns a linear model using MTWV as the maximizing criterion. When multiple modalities or hit lists are available, [1] shows that, normalizing the different hit lists first, and then using another round of Powell's method to interpolate the normalized scores, results in additional gains.

5. EXPERIMENTAL RESULTS

The audio corpora and keyword sets that we considered in our research were provided by the IARPA Babel program (FullLP releases). The languages and their corresponding releases were Cantonese (IARPA-babel101b-v0.4c), Pashto (IARPA-babel104b-v0.4bY), Tagalog (IARPA-babel105b-v0.4), Turkish (IARPA-babel106b-v0.2g) and Vietnamese (IARPA-babel107b-v0.7). The condition we consider in this paper is the so-called Pre-Indexed condition, where the keywords are not known in advance of the decoding of the audio.

The training data for each language were of the order of 100 hours, and the data on which we report performance are: (i) Dev set of each language, about 10 hours each, (ii) Test set

of each language, with durations 5 hours (Cantonese, Pashto, Tagalog and Turkish) and 15 hours (Vietnamese). The test sets were supplied by NIST as "unsequestered" parts of the official evaluation sets used in the March/April 2013 Babel evaluations. The keyword sets on which we report results are the official lists provided by NIST for the evaluations; their sizes are 3762 for Cantonese, 3842 for Pashto, 3171 for Turkish, 3805 for Tagalog, and 4065 for Vietnamese.

All decodings were done with the BBN Byblos system. The BBN Byblos system uses Hidden Markov Models (HMMs), with State-Clustered-Tied Mixture (SCTM) cross-word quinphone models. The parameters for these models are estimated using the Minimum Phone Error (MPE) objective criterion. The acoustic features are based on a 6-layer stacked bottleneck neural network architecture [8].

Recognition is performed using the BBN two-pass decoder. The forward pass uses a State Tied Mixture (STM) model, and an approximate bigram LM to produce word-ending scores. The backward pass then uses the word-ending scores and associated scores from the forward pass to perform a detailed search using within-word state-clustered tied-mixture (SCTM) quinphone acoustic models and a trigram language LM to produce a lattice. Finally, lattice rescoring using a state clustered cross-word quinphone model is done. More details about the system used in the 2013 IARPA Babel evaluation can be found in [9].

5.1. Score Normalization Results

Table 1 shows the ATWV results on all five languages used in the first year of the Babel program. The rows correspond to the different normalization methods (except for "raw", which corresponds to the original, unmodified posteriors). KST+LS corresponds to the procedure of first using the linear-fit method to transform the original posteriors, and then feeding them into the KST normalization of Section 2. ML corresponds to the unmodified machine learning method of Section 4 that uses the original posteriors as input. ML+LS corresponds to a modified machine learning method that combines the normalized posteriors and the scores obtained from the linear fit method. A few observations are in order: (i) KST+LS is better than plain KST, giving, on average over the five languages, gains of 3% (absolute) on the Test data. This suggests that the raw posteriors generated through phonetic search are quite noisy. (ii) The machine learning method can still benefit from the scores obtained through the linear fit method, giving, on average over the five languages, gains of 3.8% (absolute) over KST+LS on the Test data .

	Ca	Pa	Tu	Ta	Vi
raw	20.7%	18.3%	19.4%	22.6%	20.2%
unsupervised					
KST	37.0%	32.0%	36.0%	38.3%	46.1%
KST+LS	37.6%	35.1%	41.4%	38.5%	52.8%
supervised					
ML	45.1%	37.2%	41.3%	44.4%	51.4%
ML+LS	**49.3%**	**40.0%**	**45.1%**	**47.6%**	**55.9%**

(a) ATWV Results on the Dev data using phonetic search.

	Ca	Pa	Tu	Ta	Vi
raw	22.9%	15.4%	19.2%	18.6%	18.1%
unsupervised					
KST	38.8%	30.8%	37.0%	36.5%	40.6%
KST+LS	40.7%	32.3%	40.9%	37.1%	47.9%
supervised					
ML	43.8%	33.0%	39.6%	40.5%	38.9%
ML+LS	**46.8%**	**35.5%**	**43.0%**	**43.2%**	**49.2%**

(b) ATWV Results on the Test data using phonetic search.

Table 1. Score normalization results. The best result in each column is shown in bold.

6. CONCLUDING REMARKS

In this paper we showed that, when performing phonetic search, it is important to convert the detection scores so that they accurately resemble posteriors that correspond to the probability of being correct. We presented two ways of doing this: (i) using an unsupervised linear fit method that uses a Poisson assumption; (ii) using a supervised machine learning method that reranks hits based on many features, including the ones resulting from the linear fit. In both cases, we obtained absolute ATWV gains of at least 3% (on average, over the five languages used in the first year of the Babel program), as compared to just normalizing the raw posteriors.

7. ACKNOWLEDGMENTS

We would like to thank all members of the BBN Speech and Language group for useful discussions, and especially those who work on the Babel project. We would also like to acknowledge the help and contribution of other partners of the BABELON team on the IARPA-funded Babel project.

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0013. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

8. REFERENCES

[1] D. Karakos, R. Schwartz, S. Tsakalidis, L. Zhang, S. Ranjan, T. Ng, R. Hsiao, G. Saikumar, I. Bulyko, L. Nguyen, J. Makhoul, F. Grezl, M. Hannemann, M. Karafiat, I. Szoke, K. Vesely, L. Lamel, and V.-B. Le, "Score normalization and system combination for improved keyword spotting," in *Proc. ASRU 2013*, Olomouc, Czech Republic, 2013.

[2] J. Mamou, J. Cui, X. Cui, M. J. F. Gales, B. Kingsbury, K. Knill, L. Mangu, D. Nolden, M. Picheny, B. Ramabhadran, R. Schlüter, A. Sethy, and P. C. Woodland, "System combination and score normalization for spoken term detection," in *Proc. of ICASSP*, May 2013.

[3] I. Bulyko, O. Kimball, M.-H. Siu, J. Herrero, and D. Blum, "Detection of unseen words in conversational Mandarin," in *Proc. of ICASSP*, Kyoto, Japan, Mar 2012.

[4] NIST, "OpenKWS13 keyword search evaluation plan," http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf, 2013.

[5] D. R.H. Miller, M. Kleber, C.-L. Kao, O. Kimball, T. Colthurst, S. A. Lowe, R. M. Schwartz, and H. Gish, "Rapid and accurate spoken term detection," in *Proc. of Interspeech*, 2007.

[6] B. Zhang, R. Schwartz, S. Tsakalidis, L. Nguyen, and S. Matsoukas, "White listing and score normalization for keyword spotting of noisy speech," in *Proc. of Interspeech*, Portland, Oregon, Sep 2012.

[7] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The art of Scientific Computing*, Cambridge University Press, 2007.

[8] M. Karafiat, F. Grezl, M. Hannemann, K. Vesely, and H. Cernocky, "BUT Babel system for spontaneous Cantonese," in *Proc. of Interspeech*, Lyon, France, Aug 2013.

[9] S. Tsakalidis, R. Hsiao, D. Karakos, T. Ng, S. Ranjan, G. Saikumar, L. Zhang, L. Nguyen, R. Schwartz, and J. Makhoul, "The 2013 BBN Vietnamese telephone speech keyword spotting system," in *Proc. ICASSP 2014*, Florence, Italy, 2014.