9-2014

Word-based Probabilistic Phonetic Retrieval for Low-resource Spoken Term Detection

Di Xu
Carnegie Mellon University

Florian Metze
Carnegie Mellon University, fmetze@andrew.cmu.edu

Published In
Proceedings of INTERSPEECH.

Word-based Probabilistic Phonetic Retrieval for Low-resource Spoken Term Detection

Di Xu, Florian Metze

Language Technologies Institute, School of Computer Science, Carnegie Mellon University
Pittsburgh, PA, USA

dix@cs.cmu.edu, fmetze@cs.cmu.edu

Abstract

Two problems make Spoken Term Detection (STD) particularly challenging under low-resource conditions: the low quality of speech recognition hypotheses, and a high number of out-of-vocabulary (OOV) words. In this paper, we propose an intuitive way to handle OOV terms for STD on word-based Confusion Networks using phonetic similarities, and generalize it into a probabilistic and vocabulary-independent retrieval framework. We then reflect on how several heuristics and Machine Learning based methods can be incorporated into this framework to improve retrieval performance. We present experimental results on several low-resource languages from IARPA's Babel program, such as Assamese, Bengali, Haitian, and Lao.

Index Terms: Information retrieval, spoken term detection, out-of-vocabulary, classification, machine learning

1. Introduction

State-of-the-art approaches in Spoken Term Detection (STD) involve Large Vocabulary Continuous Speech Recognition (LVCSR) [1] that transcribes audio data into either Lattice [2] based or Word Confusion Network (WCN) [3] based hypotheses, and keyword search is conducted. LVCSR cannot handle out-of-vocabulary (OOV) terms and is therefore guaranteed to make errors on OOV terms. Moreover, the overall retrieval performance is limited by the quality of the hypotheses.

Low-resource conditions aggravate the two problems: sparse audio training data and language resources lead to poor Acoustic Models (AM) and Language Models (LM). An Automatic Speech Recognition (ASR) System will in turn generate erroneous and confusing hypotheses for keyword search. On the other hand, the rate of query terms that will be OOV with respect to the ASR system also surges due to the limited resources available. Together, the two problems make low-resource STD particularly challenging, which is the typical underlying condition of the IARPA Babel program [4].

It is not surprising that all existing solutions to the OOV problem resort to phonetic information to some extent. A relatively novel approach was proposed in [5], where a letter-to-sound system was introduced to obtain multiple hypotheses of the query. Then the search is conducted over phone based hypotheses, and their detections are merged. Most traditional approaches rely on dedicated phone lattices such as in Mamou et al. [6], where both WCN and phone lattices are used during the retrieval. Such methods are capable of retrieving OOV terms, but the advantage comes with the cost of generating separate phone transcriptions, and the scoring method is complicated as the search involves both word and phone transcripts. Many other popular approaches resort to subword representations such as in [7], where grapheme-to-phoneme (G2P) system is used and flat hybrid sub-lexical models were proposed to degrade OOV terms into sub-lexical units. However, subword-based approaches increase the risk of false detections as they appear more frequently, thus the index size and search space also increases. In addition, some approaches such as in [8], where OOV query terms are expanded into proxies that are morphologically similar to the original term. Such methods borrow the idea of Query Expansion from Information Retrieval (IR) but they are more useful in Spoken Document Retrieval, where related information in the same document can be used for query expansion. Interestingly, as we will reveal later, we also seek to bridge the gap between IR and STD in a new way that inherits the merits of many existing approaches.

To overcome these challenges under low-resource conditions, we introduce an intuitive way to handle OOV queries in a WCN framework using phonetic similarities. On top of that, we develop a novel vocabulary-independent probabilistic phonetic retrieval framework (PPR) that effectively improves the overall retrieval performance based on a strong probabilistic foundation. In addition, we exploit the potential of this retrieval framework by using Machine Learning based methods to further enhance the retrieval performance.

2. Probabilistic Phonetic Retrieval Framework

2.1. Word Confusion Networks

The hypotheses for each utterance generated by the ASR system are first assembled into a lattice, which is an acylic graph with each path representing a particular hypothesis of a utterance. The word lattices are overwhelmingly large with lots of redundant information, making them expensive to index and search over. In contrast, a WCN represents a set of hypotheses of a utterance with a chain-like structure and each segment stores hypotheses for a particular word. The algorithms of converting lattices to WCNs are presented in [3]. For hypotheses that are of different lengths, a segment may also contain a link symbol along with its word hypotheses, indicating that it is skippable.

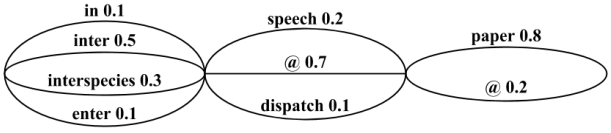

Figure 1: WCN of "interspeech paper"

Figure 1 gives an example of the WCN segments for the

phrase "interspeech paper", where @ denotes a skippable segment, and the term "interspeech" happens to be OOV.

2.2. OOV Handling

Given a random query term, we first check if the term is OOV: if so, a G2P system [9] is used to obtain the pronunciation. Otherwise, we simply look into the training lexicon to get its pronunciation(s). Then, retrieval is based on both the graphemic and phonetic representation of the terms. From this point, we will no longer distinguish between IV and OOV terms, because we define our relevance measure to be the pair-wise phonetic similarity presented in 1, later referred to as "similarity" for short. The similarity is defined as follows:

$$sim(w_1, w_2) = 1 - \frac{Lev\{G2P(w_1), G2P(w_2)\}}{\max\{|G2P(w_1)|, |G2P(w_1)|\}}, \quad (1)$$

in which *G2P* provides the phone sequence of a word, and *Lev* is the Levenshtein distance between two phone sequences. This provides a probabilistic measure of how two terms are phonetically similar, and it also generalizes graphemic similarities as the same words must have the same pronunciations. Such generalization simplifies the entire retrieval framework and allows more sophisticated methods to be used. Although one may assume non-linear relation between the similarity and level of relevance, in this paper we assume this simplified relation.

Unlike in [6] where dedicated phone lattices are generated for words missing from the WCNs, we rely only on the G2P System for OOV terms and use the training lexicon for invocabulary (IV) terms, which saves the effort of generating, indexing and searching over phone lattices. Such a light-weight solution has its advantages because in practice the decoding effort will be much more costly with hundreds or even thousands of hours of audio data. In addition, we also penalize the ASR system for deletion errors and insertion errors to ensure that most errors are phonetically similar to the truths, which is particularly important for OOV and difficult IV terms.

2.3. Indexing

The goal of generating an index is to avoid linear searches of the entire corpus. We construct a word-based inverted index, which can be viewed as a symbol table which maps a query term to its exact occurrences in the WCNs. Here, for later convenience we define an "exact match" to be a perfect string match, whereas a "fuzzy match" accounts for words that are phonetically similar to a certain level. It is tricky to construct indexes that handle fuzzy matches because the level of acceptable "fuzziness" varies upon different words in different queries. Although there are sophisticated indexing techniques proposed in [10] [11] that can be used to handle fussy indexing, since we have only a fixed set of queries, for simplicity we just cache some of the fuzzy matches that are close enough in the event of zero exact match, and this "fuzzy cache" is updated along the way.

2.4. Keyword Search

In this stage, a super set of the final detections will be generated, and we define all detections found in this stage to be "raw" detections. The upper-bound of the recall is determined by the set of raw detections, and in later stages, more sophisticated methods will be used to make the final decisions on the raw detection. In this stage the goal is to discover as many raw detections as possible, without blowing up the search space.

Searching for single-word queries consists of using the index to map their exact matches in the WCN. It will certainly happen that a term finds no exact matches, indicating it is either OOV or a difficult IV word that the ASR chose to ignore. In that case we will resort to fuzzy matches and compare its pronunciations with all terms in the index and select close fuzzy matches. This incurs a O(N) search, but once we have stored the mappings in the fuzzy cache, the mapping is O(1). Apart from using a cache, we can also cluster the terms by their pronunciations and construct a symbol table which maps a random term to a cluster of phonetically most similar terms, thus instead of linear search over the entire index, we perform linear search over a small subset of the index.

For multi-word queries, we first locate all possible exact matches of each query term, and define those matches as "pivots". For each pivot, depending on its corresponding position in the query, we find possible nearby locations in the WCN of the other query terms. To achieve this, we construct a forward and a backward partial search tree rooted at the pivot, then traverse the trees and join the partial search paths from both sides. An alternative is to perform Dynamic Programming (DP) and find one or several best partial search paths from both sides of the pivot and join them. Empirically the tree search works slightly better in terms of ATWV. It is also fast in enumerating all possible detections, yet the number of detections found by tree search is exponential with respect to query length. This is because links in the WCN segments are also considered. As the query becomes longer, a detection is allowed to span more segments, and the tree search will generate a permutation of paths involving link and non-link segments. On the other hand, although DP is more expensive when searching for the best paths as its complexity is $O(n^2)$, it returns only one or several best detections, thus the cost of scoring the detections is almost constant.

Typically, only a small number of the raw detections for multi-gram queries are exact matches, and the rests are partially fuzzy matches. If all the query terms are OOV or difficult IV terms, it is possible that none of the query terms have exact matches. In that case, we may relax to fuzzy pivots, but this is rare since the longer the query, the more likely that one query term may have a exact match. The reason we restrict to exact pivots in the first place is to prevent the search space from exploding as it is possible that some queries may find an exploding number of fuzzy matches based on fuzzy pivots.

At this point, for each query we have a number of raw detections to evaluate and our next goal is to score the detections to distinguish true detections from false alarms (FA).

2.5. Detection Scoring

In this section, we denote a query by Q, a raw detection by Det which consists of an array of ordered segments denoted by S, and each segment by s_i. Notice that S's can have different lengths as some s_i's may be links, but the number of non-link s_i's must be equal to the length of query. A query term that corresponds to a particular non-link s_i is denoted by q_{s_i}. Each s_i also consists of a bag of word hypotheses denoted by W_i, and each hypothesis is denoted by w_j.

A noisy channel model can be used to identify the intended word given a hypothesis generated from a distortion process. Here, to address the error correction problem in detection scoring, we modify the decision function of the original model and let it be our scoring function, which evaluates how likely a hypothesis is in fact a distortion of the intended word. Based on that, the relevance of each word hypothesis w_j can be measured

as the likelihood that w_j is generated from its corresponding query term q_{s_i}, as presented in equation 2.

$$P(w_j|q_{s_i}) = \frac{P(q_{s_i}|w_j)P(w_j)}{P(q_{s_i})} \propto P(q_{s_i}|w_j)P(w_i) \quad (2)$$

We drop $P(q_{s_i})$ by assuming uniform. $P(q_{s_i}|w_i)$ can be approximated by the probabilistic phonetic similarity (sim) between q_{s_i} and w_j, and the $P(w_i)$ can be viewed as the confidence score ($conf$). Without losing generality, $P(q_{s_i}|w_i)$ of each link s_i is assumed to be 1, so that equation 2 works for both link and non-link s_i's.

In each non-link s_i, there is a bag of evidences suggesting how likely s_i is generated from q_{s_i}. We aggregate these evidences by summing up $P(w_j|q_{s_i})$ over all w_j's, which provides relevance measure (R) of each s_i given q_{s_i}:

$$R_{s_i}(q_{s_i}) := \sum_{j}^{|W_i|} P(q_{s_i}|w_j)P(w_j). \quad (3)$$

However, this is no longer a probability. To normalize, we divided this value by the summation of the relevance across all segments in the WCN, which results in the probability

$$P_{s_i}(q_{s_i}) = \frac{R_{s_i}(q_{s_i})}{\sum_{s \in WCN} R_s(q_{s_i})}. \quad (4)$$

The probability of a Det being true is defined as the joint probability of each non-link s_i being generated from its corresponding q_{s_i}, and each link s_i being a link. By assuming independence, the probabilistic score of a detection is simply the product of all $P_{s_i}(q_{s_i})$'s. For the purpose of ranking, the normalization in 4 is not needed. Thus the final detection scoring formula reduces to

$$R_{Det}(Q) := \prod_{i}^{|S|} \sum_{j}^{|W|} sim(q_i, w_j) * conf(w_j). \quad (5)$$

2.6. Detection Pruning

We use the Expected Count Thresholding (ECT) proposed in [12] to compute dynamic thresholds for the detections scores of different queries. This complies with the ATWV calculations and thus the thresholds are principally optimal if the scores of the detections reflect the probabilities of being true occurrences.

3. False Alarm Filtering

The proposed approach brings a huge increase in recall, but one potential problem is the increased rate of FAs, which impair our improvements in ATWV. The number of FAs surges as fuzzy matches are considered. It is not very effective to suppress the incurred FAs by setting up a hard similarity threshold during the search, as the optimal threshold varies across queries. That means we need to resort to other information available, such as the time span of the detection, query length, the distribution of confidence scores in each segment, and the confidence scores of the most likely terms across the segments. On the other hand, many other heuristic approaches in calculating the detection score may also be considered as features to spot FAs. Naturally, this evolves into a binary classification problem.

We resort to Support Vector Machine (SVM) [13], as it is a principled method for high dimensional classification. We will use Gaussian Radial Basis Function (RBF) kernels [14] to deal with non-linearly separable data.

4. Experiments

4.1. Speech-to-Text System

The underlying ASR system is based on the Janus Toolkit [15], using the Ibis decoder [16]. The ASR system trains AMs using several aggregations of features including Fundamental Frequency Variation [17], Pitch [18], Mel-Frequency Cepstrum Coefficients [19], and log-MEL coefficients. Different training mechanisms are used, including Speaker-Adaptive Training with Maximum Likelihood and boosted Maximum Mutual Information GMMs in a Deep Bottle-neck feature space [20], along with Hybrid Deep Neural Networks [21] and Maxout Networks [22]. Moreover, Confusion Network Combination (CNC) is conducted to aggregate the hypotheses of eight individual systems. The systems have been trained on 10h of "LimitedLP" training data, which were also used exclusively to provide language model training data and dictionary.

4.2. Experimental Setup

The STD task is defined by NIST in the OpenKWS14 Evaluation Plan [23]. Table 1 summarizes the performance of our preliminary ASR systems at the time this paper was composed.

Language	Data (LimitedLP)	WER	WCN Size
Assamese	IARPA-babel102b-v0.5a	63.1%	102.5 MB
Benagli	IARPA-babel103b-v0.4b	64.7%	119.0 MB
Haitian	IARPA-babel201b-v0.2b	60.8%	104.9 MB
Lao	IARPA-babel203b-v3.1a	59.7%	96.6 MB

Table 1: Data and ASR Performance

It is worth stressing the point again that conventional STD approaches and Keyword Search (KWS) methods do not assume such difficult conditions. The WERs are high because the training resources are very limited and we lack enough audio data to train better AMs and textual data to train better LMs.

For each language, we train an ASR system to transcribe 10 hours of test data, and their hypotheses are generated as massive word lattices and then converted into compact WCNs. WERs are calculated based on the best paths in lattices or WCNs to evaluate the performance of each individual ASR system. On top of that, CNC is performed to generate the final hypotheses which are to be indexed to perform STD by PPR.

The baseline retrieval framework implements a Ranked-Boolean [24] method which by nature cannot handle OOV queries. We will focus on how PPR improves ATWV for both IV and OOV queries in various aspects.

4.3. ATWV, Recall and False Alarm Analysis

		Assamese	Bengali	Haitian	Lao
# OOV/# Total		579/1959	685/1967	448/1919	258/1925
Baseline		0.2278	0.2235	0.3583	0.2332
PPR	IV	0.2374	0.2239	0.3768	0.2574
	OOV	0.0074	0.0167	0.0122	0.0017
	All	**0.2448**	**0.2406**	**0.3890**	**0.2591**
P-value		9.9E-6	8.4E-5	2.4E-8	2.0E-3

Table 2: ATWVs for 10-hour development data using development keywords, and statistical significance

According to Table 2, we can see that PPR has significantly improved ATWV on both IV and OOV queries for all four languages. The ATWV for IV and OOV queries are calculated with respect to the entire query set, such that they sum to the overall ATWV. The improvement on IV queries also indicates that PPR is fundamentally better than the baseline method not only for OOV queries, but also in terms of the overall improvement in recall and the ability of reasonably inhibiting the FA rate. Notice that the improvement on OOV is also subject to the portion of OOV queries in the entire query set. That is why the OOV improvement for Lao is not as significant as others. The last row suggests the statistical significance of the improvements on overall ATWVs, and the p-values are computed using the Wilcoxon signed-rank test [25].

		Assamese	Bengali	Haitian	Lao
Baseline	Recall	1584	1572	2194	1672
	FA	3416	2991	3105	3227
PPR	Recall	**1694**	**1688**	**2322**	**1945**
	FA	4555	4234	3838	5301

Table 3: Recalls and False Alarms

Through Table 3, the improvements in recall is obvious. Although the increase in FA is also a problem, but the ATWV was still increased because ATWV is a metric which values recall far more than FAs. Moreover, the loss of each FA incurred also depends on the rareness of the query. In particular, the rarer the query terms, the less ATWV loss incurred by each FA and the more gains incurred by each true detection. Nevertheless, the FA issue should be considered as a major bottleneck for PPR. The advantage of PPR is its enhancements on recall for both IV and OOV queries, yet the FAs that are incurred add to the cost of this method. To improve PPR, principled methods should be considered to inhibit FAs even further.

4.4. Removing False Alarms with Support Vector Machine

The goal in this section is to discover what kinds of detections are likely to be FAs. The hypotheses that PPR works on are provided by the lattice generation, WCN generation and CNC processes, during which various decoding parameters can easily affect the properties of the resulting hypotheses, which in turn may affect the performance of the retrieval system. For example, during the WCN generation, the average number of word hypotheses in each segment can be drastically different with different parameters. Intuitively, as the average density of the segments increases, the chance of retrieving FAs from the WCN increases significantly. In addition, the WCN generation also controls the average time span of each segment. Using different parameters, the minimal time span for a segment can be exceedingly small thus the resulting WCN may have a huge number of segments confined within a very small time span. This adds difficulty to PPR as the number of possible detections may explode in certain regions. That is why it is important to look ahead at the phonetic similarity and the confidence scores, and consider other aspects informations which may indicate the reliability of certain detections. Based on this motivation, We have engineered a series of features listed as follows:

- PPR score computed according to formula (4)
- a variation of the PPR score. Instead of considering the weighted sum of term confidences in each segment, we consider term with maximum similarity, denoted by w^*. This can be calculated by the following

$$P(Det) := \prod_{i}^{|Det|} sim(w^*{}_i, q_i) * conf(w_i^*) \quad (6)$$

- mean, variance and entropy of the confidence scores and similarities to q for w_i^*'s across all segments
- length of the corresponding query
- time span of the detection
- number of links in the detection

We use RBF kernel to expand the basic features into higher dimensional spaces. We have also put different regularization weights to severely penalize misclassifying a true detection as a FA, such that the damage to recall is minimized. Below are the improved results by filtering out FAs using SVM. The models are trained and validated using 10-fold cross validation.

	Assamese	Bengali	Haitian	Lao
Recall	1688 (-0.3%)	1684 (-0.2%)	2321 (-0.0%)	1940 (-0.2%)
FA	4312 (-5.3%)	3972 (-6.2%)	3706 (-3.4%)	4875 (-8.0%)
ATWV	**0.2451**	**0.2416**	**0.3900**	**0.2627**

Table 4: Inhibiting False Alarm by SVM

According to Table 4, the benefits of using SVM to filter out FAs can be observed as the FAs are reduced significantly comparing to Table 3, and ATWVs have been further improved slightly comparing to Table 2.

5. Conclusions

In this paper, we present the "Probabilistic Phonetic Retrieval" (PPR) framework for STD, and evaluate its effectiveness. Results show that the probabilistic scoring method is a reasonable estimate of the probability of a true detection. However, this framework also tends to retrieve more FAs, which partially offsets ATWV improvements achievable through retrieval of OOV words. By extracting features from the detections, we formulate the problem of inhibiting FAs into a classification problem and demonstrate the effectiveness of using Support Vector Machines with non-linear kernel.

In future, we should resort to better phonetic similarity measures and introduce a phone confusion matrix to further enhance PPR's detection scoring quality. After all, the one proposed in this paper provides only a coarse approximation, without considering various phone-to-phone distances. We will also incorporate more features during the WCN generation in addition to the confidence score, and use better kernel functions.

6. Acknowledgements

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0015. 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.

7. References

[1] F. Triefenbach, K. Demuynck, and J.-P. Martens, "Large vocabulary continuous speech recognition with reservoir-based acoustic models," *IEEE Signal Processing Letters*, vol. 21, no. 3, pp. 311–315, 2014.

[2] F. Weng, A. Stolcke, and A. Sankar, "Efficient lattice representation and generation," in *In Proc. of ICSLP*. Citeseer, 1998.

[3] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.

[4] M. Harper, "Iarpa solicitation iarpa-baa-11-02," *IARPA BAA*, 2011.

[5] D. Can, E. Cooper, A. Sethy, C. White, B. Ramabhadran, and M. Saraclar, "Effect of pronounciations on oov queries in spoken term detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009.* IEEE, 2009, pp. 3957–3960.

[6] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '07. New York, NY, USA: ACM, 2007, pp. 615–622.

[7] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models." in *INTERSPEECH*, 2005, pp. 725–728.

[8] Y.-C. Li, W.-K. Lo, H. M. Meng, and P. Ching, "Query expansion using phonetic confusions for chinese spoken document retrieval," in *Proceedings of the fifth international workshop on on Information retrieval with Asian languages*. ACM, 2000, pp. 89–93.

[9] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.

[10] P. Yu, K. Chen, C. Ma, and F. Seide, "Vocabulary-independent indexing of spontaneous speech," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 635–643, 2005.

[11] J. Cui, X. Cui, B. Ramabhadran, J. Kim, B. Kingsbury, J. Mamou, L. Mangu, M. Picheny, T. N. Sainath, and A. Sethy, "Developing speech recognition systems for corpus indexing under the iarpa babel program," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013*. IEEE, 2013, pp. 6753–6757.

[12] D. R. 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 *INTERSPEECH*, 2007, pp. 314–317.

[13] V. Vapnik, S. E. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation, and signal processing," *Advances in neural information processing systems*, pp. 281–287, 1997.

[14] B. Schölkopf and A. J. Smola, *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[15] M. Finke, P. Geutner, H. Hild, T. Kemp, K. Ries, and M. Westphal, "The karlsruhe-verbmobil speech recognition engine," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997*, vol. 1. IEEE, 1997, pp. 83–86.

[16] H. Soltau, F. Metze, C. Fugen, and A. Waibel, "A one-pass decoder based on polymorphic linguistic context assignment," in *Automatic Speech Recognition and Understanding, 2001. ASRU'01. IEEE Workshop on*. IEEE, 2001, pp. 214–217.

[17] K. Laskowski, M. Heldner, and J. Edlund, "The fundamental frequency variation spectrum," *Proceedings of FONETIK 2008*, 2008.

[18] K. Schubert, "Grundfrequenzverfolgung und deren Anwendung in der Spracherkennung," Master's thesis, Universität Karlsruhe (TH), Germany, 1999, in German.

[19] S. Furui, "Cepstral analysis technique for automatic speaker verification," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, no. 2, pp. 254–272, 1981.

[20] J. Gehring, Y. Miao, F. Metze, and A. Waibel, "Extracting deep bottleneck features using stacked auto-encoders," in *Proc. ICASSP*. Vancouver, BC; Canada: IEEE, May 2013.

[21] J. Gehring, Q. B. Nguyen, F. Metze, and A. Waibel, "DNN acoustic modeling with modular multi-lingual feature extraction networks," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 344–349.

[22] Y. Miao, F. Metze, and S. Rawat, "Deep maxout networks for low-resource speech recognition," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, Dec 2013, pp. 398–403.

[23] NIST, "Spoken term detection (std) 2014 evaluation plan," http://www.nist.gov/itl/iad/mig/openkws14.cfm, 2014.

[24] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge university press Cambridge, 2008, vol. 1.

[25] R. H. Randles, "Wilcoxon signed rank test," *Encyclopedia of statistical sciences*, 1988.