The SRI/OGI 2006 Spoken Term Detection System

Dimitra Vergyri[1], *Izhak Shafran*[2], *Andreas Stolcke*[1]
Ramana R. Gadde[1], *Murat Akbacak*[1], *Brian Roark*[2], *Wen Wang*[1]

[1]SRI International, Menlo Park, CA
[2]OGI School of Science & Engineering, OR
{dverg,stolcke,murat,wwang}@speech.sri.com, {zak,roark}@cslu.ogi.edu

Abstract

This paper describes the system developed jointly at SRI and OGI for participation in the 2006 NIST Spoken Term Detection (STD) evaluation. We participated in the three genres of the English track: Broadcast News (BN), Conversational Telephone Speech (CTS), and Conference Meetings (MTG). The system consists of two phases. First, audio indexing, an offline phase, converts the input speech waveform into a searchable index. Second, term retrieval, possibly an online phase, returns a ranked list of occurrences for each search term. We used a word-based indexing approach, obtained with SRI's large vocabulary Speech-to-Text (STT) system.

Apart from describing the submitted system and its performance on the NIST evaluation metric, we study the tradeoffs between performance and system design. We examine performance versus indexing speed, effectiveness of different index ranking schemes on the NIST score, and the utility of approaches to deal with out-of-vocabulary (OOV) terms.
Index Terms: spoken term detection, audio indexing

1. Introduction

Spoken communication is a major source of information. Vast amounts of audio data are being created and digitally stored daily. Since information processing has become a primary economic activity in the world, there is a pressing need for intelligent retrieval of information from the ever-growing archives of recorded speech. Spoken term detection (STD) aims to make this information available by locating a specified *term* rapidly and accurately in large heterogeneous audio archives, to be used ultimately as input to more sophisticated retrieval technologies. A term is defined as a sequence of one or more words (as many as five words for the NIST evaluation task).

Unlike spoken document retrieval [1] or spoken utterance retrieval [2], the STD task was formulated as a detection task [3], requiring each occurrence to be specified in terms of its start and end times. In addition, systems need to provide a score and a hard decision that indicates the correctness of each occurrence. The input for the task consists of raw audio files segments and a list of search terms. Although the evaluation actually uses only modest amounts of data, it is structured to simulate the very large data situation. Therefore, the system was required to be implemented in two phases: indexing and searching. The system processes the audio data once, during the indexing phase, without knowledge of the terms. The output index is stored and accessed during the searching phase, in order to locate the terms and link them to the original audio. The searching phase may be repeated multiple times for different terms so the efficiency of its implementation is very important. Because of the vast amount of data typically needed to be indexed, the runtime of the indexing phase and the final size of the stored index are also important.

2. The STD Task

2.1. Data

The test data consists of audio waveforms, an excerpt list and the query terms. NIST provided development (dev06) and dry-run (dry06) test sets, however, the audio was common to both sets. Development set was used to tune the parameters of our system, and consisted of about 3 hrs of BN, 3 hrs of CTS and 2 hrs of MTG. The dev06 and dry06 sets contained 1099 and 1107 query terms respectively. The STT components of the system was trained using corpora available from the Linguistic Data Consortium (LDC). However, data generated after December 2003 was excluded from training all STT and STD components to comply with evaluation requirements.

Additionally, we created a separate devset (sri_dev) for system development, which consisted of data previously distributed by NIST for speech recognition evaluations, namely, BN RT-02 data (3 hours), CTS dev-04 data (3 hours) and MTG RT-04s and dev-04 data (4 hours). For a more balanced set we only used the Fisher portion of the dev-04 data from the CTS extra development data (1.5 hours). We generated a query list of about 6K terms for the dev2-06 set using the NIST termlist selection tools [4].

2.2. Evaluation Metric

Since this is a detection task, performance can be characterized by detection error tradeoff (DET) curves of miss (P_{miss}) versus false alarm (P_{fa}) probabilities, or by a weighted function of the two probabilities. For the NIST STD06 evaluation the primary evaluation metric was the actual term-weighted value (ATWV), which is defined as follows [3].

$$ATWV = 1 - \frac{1}{T}\sum_{t=1}^{T}(P_{miss}(t) + \beta P_{fa}(t)) \quad (1)$$

$$P_{miss}(t) = 1 - \frac{N_{corr}(t)}{N_{true}(t)}\ ,\ P_{fa}(t) = \frac{N_{spurious}(t)}{Total - N_{true}(t)} \quad (2)$$

where T is the total number of terms, β is set to approximately 1000, N_{corr} and $N_{spurious}$ are the total number of correct and spurious (incorrect) term detections, N_{true} is the total number of true term occurrences in the corpus, and $Total$ is the duration (in seconds) of the indexed audio corpus.

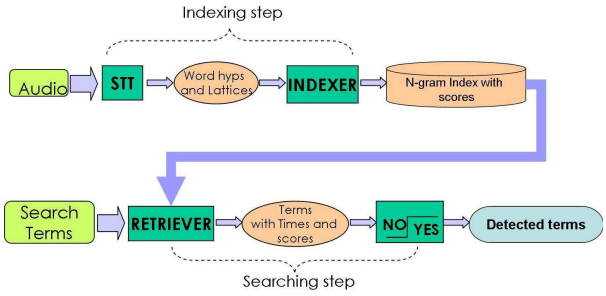

Figure 1: *Spoken Term Detection system at SRI/OGI*

3. SRI/OGI STD System Description

Indexing consists of two major steps in our system, as seen in Figure 1. First, audio input is run through the STT system that produces word recognition hypotheses and word lattices. These are converted into a candidate term index with times and detection scores (posteriors). At the retrieval step, first the search terms are extracted from within the candidate term list, and then a decision function is applied to accept or reject the candidate based on its detection score. In this section we describe in detail each of the components of the STD system.

3.1. Speech-to-Text Systems

The STT systems used for this task were a sped-up version of the STT systems used in the NIST evaluation for 2004 Rich Transcription (RT-04) [5]. The MTG system as described in [6] was based on the CTS system using domain-adapted acoustic and language models. We are using multipass [7] systems, and thus generate STT outputs with different accuracies at different stages of the system run:

- STT1: Bigram decoding with within-word models generating 1-best or word lattices.
- STT2: Lattice expansion with higher-order n-gram, followed by word posterior and 1-best generation from expanded lattices.
- STT3: Rescore expanded lattices with adapted cross-word models, and update word posteriors and 1-best generation.

All systems are using the SRI Decipher(TM) speaker-independent continuous speech recognition system, which is based on continuous-density, state-clustered hidden Markov models (HMMs), with vocabularies optimized for each genre. In the front end MFCC and/or PLP features are employed with 13 coefficients plus first, second ,and third derivatives (52 dimensions). Cepstral mean and variance normalization, vocal tract length normalization, and model-based speaker-specific feature normalization were applied. HLDA is used for dimensionality reduction (52$-$ $>$39). All acoustic models (AMs) are using decision tree state clustering, and models are trained with MLE followed by alternating MPE-MMI training [8]. In the following we give a more detailed description of the training data and models used for each of the three genres.

BN System: For acoustic training we used the data distributed from LDC as Hub-4 1996,1997 (200 h), TDT4 (274 h), TDT2 (272 h), and BNr1234 (2300 h). We trained gender-independent within-word and cross-word models with about 500K gaussians each. For the language model (LM) the training data was partitioned based on type, date, and source. Separate component LMs were generated from each partition, and then interpolated for the final LM, using a vocabulary of 58K words. In the first-pass decoding (STT1), a PLP front-end was used with within-word MPE-trained AM and a bigram LM. The lattices were rescored using a 5-gram word LM and within-word duration model to produce 1-best hypotheses for adapting a cross-word MPE-trained PLP model (STT2). The adapted cross-word model was then used to decode 4-gram expanded lattices and generate the second-pass lattices, which were again rescored with a duration model (STT3).

Table 1: *STT system results with word error rate (%WER) and runtime. Runtime is defined as how much longer than real audio time duration (xRT) it takes to complete the recognition*

		BN	CTS	MTG
dev06	STT1	29.9 (??xRT)	22.9 (??xRT)	51.0 (??xRT)
	STT2	24.7 (2.5xRT)	19.2 (1.8xRT)	48.6 (5.4xRT)
	STT3	23.2 (5.4xRT)	17.4 (2.5xRT)	43.3 (6.8xRT)
sri_dev	STT1	14.7	23.5	47.2
	STT2	12.2	19.1	44.3
	STT3	10.7	17.0	37.0

CTS System: Acoustic training uses all of Hub5 CTS, CTRAN Switchboard 2, and 2000 hours of Fisher data. MFCC and PLP models are trained, each using complementary halves of the Fisher corpus. LM training uses all CTS, UW-web data, and BN '96 transcripts, interpolated and entropy pruned. The recognition vocabulary consists of 48K words. The system uses dual front ends: PLP and MFCC with voicing and ICSI Tandem/HATS multi layer perceptron (MLP) features [7]. HLDA and feature-space SAT is applied for both. Two gender-dependent model sets are used: MFCC within-word (640K gaussians) and PLP cross-word (768K gaussians). For STT1 the phone-loop MLLR adapted within-word MFCC model is used with a bigram LM. The lattices are rescored with a 4-gram LM to generate 1-best hypotheses for SAT and MLLR adaptation of the cross-word PLP model (STT2). Trigram rescored lattices are used for constrained decoding with the SAT+MLLR adapted PLP cross-word model to generate the second pass lattices which are rescored using prosodic duration models and the 4-gram LM (STT3).

MTG System: The system architecture is identical to that of CTS, but for this system the MFCC+MLP-based models were originally trained on CTS data, and the PLP models were originally trained on BN data. Acoustic model and feature adaptation was performed using distant-microphone meeting recordings as described in [6]. The LMs used were linearly interpolated mixtures of component LMs trained from various sources: Switchboard CTS, Fisher CTS, Hub-4 and TDT4 BN, meetings (AMI, CMU, ICSI, and NIST), and wed data newly collected to match different topics and styles. The mixture weights were tuned to minimize perplexity on held-out meeting transcripts. The LM vocabulary consisted of 54K.

In Table 1 we see the STT system performance on the development sets. We note that the references provided for dev06 were not meant for STT purposes and were not marking regions to be excluded for scoring (for BN and MTGS). Therefore, we had a very high insertion rate that causes the unnatural high WER for BN.

3.2. N-gram Indexing

The baseline approach is to take the 1-best recognition output from the STT system, extract all the n-gram sequences with time information, and use this as a candidate term index. We

refer to this as *1-best indexing*. All candidates have score 1, so no threshold is applied at the retrieval step. Without further processing on either the search or the index terms, a recognition mistake concerning a search-term word leads to a detection error, either a miss or a false alarm.

To avoid misses (which have a higher cost in the evaluation score than false alarms) several studies have used the whole hypothesized word lattice [9, 2] to obtain the searchable index. We used the `lattice-tool` in SRILM [10] (version 1.5.1) to extract the list of all word n-grams (up to n=5). The term posterior for each n-gram is computed as the forward-backward combined score (acoustic, language, and prosodic scores were used) through all the lattice paths that share the n-gram nodes. We used a 0.5 second time tolerance to merge same n-grams with different times. All the n-gram terms with posterior score higher than 0.001 were retained, sorted alphabetically, and used as the index candidate list. Pronunciations of the words were retained in the index to use for OOV word retrieval.

Since the posteriors obtained from the lattices were not tuned to the detection task, we also trained a classifier to predict the correctness of a term, using the posterior probability as input along with a few other term features: audio-source (bnews/cts/mtg), LM joint n-gram probability, LM n-gram length, number of words in the term, term time duration. We used an MLP with eight input features, one hidden layer with ten nodes, and two class output (correct/incorrect) using cross entropy as an objective function. The STD dev-06 data and term list were used as training data for the MLP. The NIST scoring tool was used to align the term list with the lattice-obtained candidate index, and mark each of them as correct or incorrect (false alarm). Since the objective function ATWV considers the contribution of each term equally, regardless of the frequency of the term in the data, we found that in order to approximate this objective function we had to resample the MLP training data, so that we have an equal number of occurrences for each term.

3.3. Term Retrieval

The term retrieval was implemented using the Unix command, `join`, which concatanates the lines of the sorted term list and the index file for the terms common to both. No effort was spent on optimizing the runtime of the retrieval component. This computational cost of this simple retrieval mechanism depends only on the size of the index.

The correctness of a retrieved term is marked with a hard decision (YES/NO) and we investigated three mechanisms for it. Two of the techniques rely on the posterior probability generated by the STT system. When only the 1-best STT hypothesis is indexed, the posterior probability is always unity and the all retrieved terms are accepted (YES). In the case of STT lattices, we determined a global threshold for posterior probability (GL-TH) by maximizing the ATWV score, which for this task was found to be 0.3. An alternative strategy can be formulated that computes a term-specific threshold (TERM-TH), which has a simple analytical solution [11]. Based on decision theory the optimal threshold θ for each candidate should satisfy

$$\theta \cdot V_{hit} - (1-\theta) \cdot C_{fa} = 0 \quad \longleftrightarrow \quad \theta = \frac{C_{fa}}{V_{hit} + C_{fa}} \quad (3)$$

where V_{hit} is the value of a correct detection, and C_{fa} is the cost for a false alarm. For the ATWV metric we have

$$V_{hit} = \frac{1}{N_{true}(t)} \quad , \quad C_{fa} = \frac{\beta}{Total - N_{true}(t)}. \quad (4)$$

Since the number of true occurrences of the term is unknown we approximate it for the calculation of the optimal θ by sum of the posterior probabilities of the term in the corpus.

A third strategy builds a regression to predict the correctness from a set of features (REMAP-GL-TH). The input consists of features related to the retrieved term and the predictor, in our case, a neural network, predicts the correctness which is then converted into a hard decision (YES/NO) using a global threshold. We found the optimal global threshold to be at 0.5, which is an indication that the predicted values are better tuned to the detection task.

4. ATWV Results

In Figure 2 we compare the STD performance on the dev06 and sri_dev sets for different indexing and thresholding approaches. The three STT outputs STT1, STT2 and STT3 have increasing word accuracy as seen in Table 1. The STD systems, built from the 1-best STT hypothesis show that STD performance increases with STT accuracy for all subsets. Lattice-based indexing approaches provide for the three different thresholding schemes described in Section 3.3 are also presented as well as the oracle ATWV score, which marks the upper bound on achievable STD performance based on the STT lattices.

We see that the GL-TH approach provides only minor improvements over the 1-best indexing one. The more informed REMAP-CL-TH thresholding scheme provides further improvements while TERM-TH, described in Equation 3 yields the best results among the three lattice-based thresholding schemes. Since the MLP for remapping the posterions in REMAP-GL-TH is trained on the dev06 data the results between that and TERM-TH are very similar in that set. But it seems that the MLP does not generalize so well on new data (sri_dev) where it is clear that the TERM-TH scores outperform the REMAP-GL-TH ones. Also the dependence on the STT performance is smaller with the TERM-TH approach were there is practically no difference between the STT2 and STT3 results.

4.1. Error Analysis

The detection errors in the system are due to either false alarm or misses. Misses could be due to deletion or substitution errors in ASR. Alternatively, they could also be due to terms falling below the decision threshold. A rescoring of the retrieved terms such as that used in REMAP-GL-TH can potentially fix the latter type of misses as well as false alarms. The oracle results in Figure 2, show that the rescoring paradigm can improve the ATWV by about 0.04 for both testsets. The oracle score on conference meeting is surprisingly higher than the comparable system performance, illustrating the fact that these terms could be recovered using an improved thresholding scheme.

4.2. Term Mapping

When a spoken word is out-of-vocabulary, ASR often substitutes a similar sounding word or sequence of words. If we could learn the patterns of phonetic substitutionsm, it may be possible to recover such OOV terms. For example, in the dev06 test set, the term *Hanson* was out of vocabulary of the ASR system, even though a close variant *Hansen* was in vocabulary. Sometimes, even though a query may be present in the vocabulary, ASR may produce a variant of the term (e.g., *Mr.* in dev06 vs. *Mr* in the ASR lattice). While language and task-specific scripts and filters can mitigate this effect, we developed a single unified approach to tackle these problems.

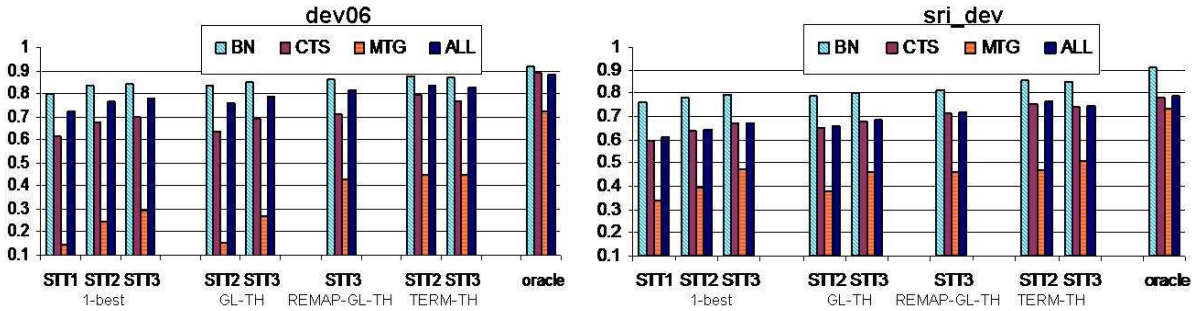

Figure 2: ATWV results on dev06 using different indexing and thresholding approaches

A query term is first mapped to its pronunciation using general techniques from speech synthesis, specifically *festival* [12]. This phone sequence, $\mathcal{P}$, is then composed with a phones-to-word transducer, $\mathcal{L}$, which is learned from the ASR lattices. The outputs are additionally weighted by the expected unigram occurrences in the ASR lattices, $\mathcal{G}$. Thus, the whole operation can be summarized as the output label of the best path in $\mathcal{P} \circ \mathcal{L} \circ \mathcal{G}$. A conservative version of this approach was applied on the test sets. No phone deletion or insertions were allowed. In addition, we experimented with restricting the substitutions to only vowels and allowing no substitutions at all. Since the test set has very low OOV rates, less than 1% on average, only small improvements were expected. The conservative implementation effected the BN subset, improving the associated ATWV score of BN by 0.007 for dev06 and 0.013 for sri_dev, making this a promising approach for future exploration.

5. Retrieval of Out-of-Vocabulary Terms

5.1. STD with Reduced STT Vocabulary

The disadvantage of word-based audio indexing, compared to phone-based approaches, is that words not included in the STT vocabulary cannot be detected. Since in this task the baseline system had a very small OOV rate, the effect of the OOV words was not significant. In order to investigate what the real effect of OOV words is, and evaluate approaches to compensate this effect, we built a system with an artificially high OOV rate. For each genre we subset the vocabulary keeping the highest frequency words in the training data, that result in a OOV rate of about 3.5on the dev06 testset. That reduced the vocabulary to 20K, 7.7K and 9.5K for each of BN, CTS and MTG tasks. The WER increased for each genre between 4-5% absolute (compared to that in Table 1), and the best ATWV result reduced by 0.1-0.2 absolute (biggest degradation was found for the CTS system).

5.2. Graphone system

In order to compensate for the OOV words we used an approach presented in [13] where *graphones*, sub-word units, are used to model OOVs. We used the 50K words (excluding the 10K most frequent ones) in our vocabulary to train the graphone module, with maximum window length set to 4. We then used a reduced word vocabulary, replacing the rest of the words in the LM training data with their graphone representations, and trained a hybrid word+graphone LM which was used for recognition. We evaluated the approach on the BN task, where we added 15K graphone units, and experimented with vocabulary sizes of 20K and 10K. The document OOV rate was 0.5% for the baseline system and increased to 1.6% and 3.3% for 20K and 10K vocabulary respectively. The OOV rate computed on the term lists went from 0.03% to 0.06% and 0.18% respectively. In Table 2 we see how the ATWV score (computed with GL-TH) changes for the reduced vocabulary systems, and how the hybrid system compensates for some of the performance loss.

Table 2: *Results for BN STT2 systems (word-based (w) and hybrid (h)), with changing vocabulary sizes. ATWV scores are reported for the threshold values of 0.3*

Vocab. Size	60K (w)	20K (w/h)	10K (w/h)
ATWV-InVocab	0.869	0.877/	0.853/
ATWV-OOV	0.000		
ATWV-all	0.836	0.807/	0.725/

6. References

[1] J. Garofolo et al., "Trec-6 1997 spoken document retrieval track overview and results", *in NIST Special Publication*, number 240, pp. 83–92, 1998.

[2] C. Allauzen, M. Mohri, and M. Saraclar, "General indexation of weighted automata – application to spoken utterance retrieval", *in Proc. of HLT/NAACL*, pp. 33–40, Boston, MA, 2004.

[3] NIST, "The Spoken Term Detection (STD) 2006 evaluation plan", `http://www.nist.gov/speech/tests/std/docs/std06-evalplan-v10.pdf`, 2006.

[4] NIST, "Nist term selection tool", `http://www.nist.gov/speech/tests/std/std2006/tools/TermSelectionTools.20061020.tgz`, 2006.

[5] A. Stolcke et al., "STT Research and Developement at SRI-ICSI-UW", `http://www.sainc.com/richtrans2004/uploads/monday/SRI-ISCI-UW.ppt`, 2004.

[6] A. Janin et al., "The ICSI-SRI spring 2006 meeting recognition system", 2007.

[7] A. Stolcke et al., "Recent innovations in speech-to-text transcription at SRI-ICSI-UW", *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, pp. 1729–1744, Sep. 2006.

[8] J. Zheng and A. Stolcke, "Improved discriminative training using phone lattices", *in Proc. Eurospeech*, pp. 2125–2128, Lisbon, Sep. 2005.

[9] D. James and S. Young, "A Fast Lattice-Based Approach to Vocabulary Independent Word Spotting", *in Proc. of ICASSP*, pp. 1029–1032, Istanbul, Turkey, 2000.

[10] A. Stolcke, "The SRI language modeling toolkit", `http://www.speech.sri.com/projects/srilm/manpages`.

[11] D. Miller, M. Kleber, C. Kao, and O. Kimball, "Rapid and accurate spoken term detection", *in Proc. Interspeech (submitted)*, 2007.

[12] A. Black et al., "The Festival speech synthesis system", `http://www.cstr.ed.ac.uk/projects/festival`, 1998.

[13] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models", *in Proc. of Interspeech*, pp. 725–728, Istanbul, Turkey, 2005.