Word and Sub-word Indexing Approaches for Reducing the Effects of OOV Queries on Spoken Audio

Beth Logan, Pedro Moreno and Om Deshmukh*
COMPAQ Cambridge Research Laboratory
One Cambridge Center
Cambridge, Massachusetts, 02142-1612, United States
Beth.Logan@compaq.com, Pedro.Moreno@compaq.com

ABSTRACT

We explore the problem of out of vocabulary (OOV) queries in audio indexing systems by comparing three indexing methods on a broadcast news repository containing 75 hours of audio. Our systems are word-based, phoneme-based and a novel system based on syllable-like units called *particles*. To better examine the performance of these three approaches we use a query set where the percentage of OOVs has been artificially increased to 50%. We additionally investigate whether the combination of the three indexing techniques can yield improvements in retrieval. We explore several simple combination strategies such as weighted combinations. We find that combining word and sub-word based systems results in improved retrieval performance.

Keywords

Speech indexing, sub-word indexing

1. INTRODUCTION

Improvements in speech recognition technology and computing power have enabled the development of usable indexes for vast spoken audio repositories (e.g. [9]). A standard technique is to use speech recognition to transcribe the audio and then to build an index using this transcription. However, this approach suffers from the fact that a speech recognizer has a limited vocabulary so the system cannot retrieve out of vocabulary (OOV) queries.

A popular technique to confront this problem is to use phoneme rather than word recognition. Here, a phoneme recognition system is used to transcribe the spoken audio. Word queries are then converted to phoneme sequences and searched for in the transcriptions.

Since phoneme recognition is less accurate than word recognition, a typical approach consists of generating for each audio segment to be indexed a lattice of phonemes encoding potentially thousands of alternative hypotheses (e.g. [5], [4]). Queries are then matched against the lattice via a scanning procedure. Improved accuracy can be obtained by converting a word recognition lattice to phonemes using a dictionary rather than performing phoneme recognition directly [5]. Additionally, phonetic confusion matrices may be used to expand the query and document representations (e.g. [7], [8]).

While the lattice approach is attractive, searching through many hypotheses and confusions is a double-edged sword. Potentially many false-positives can be generated. This can be quite significant for large repositories. For example, in [4] a false alarm rate of the order of 0.5 per hour of audio indexed is quoted for phoneme queries of length 7-11. For an index of 1,000,000 hours, this would mean that a single query might generate 500,000 or so false alarms. Even for repositories only 10,000 hours long we still would have 5,000 false alarms per query. In addition to these problems lattice based systems suffer from low speeds while searching. Since it is not possible to build a hash table structure for quick access, the cost of search is linear with the size of the audio repository. To alleviate the problem of search speed we can build an index structure of sequences of phonemes or syllables (e.g. [11]). However, the problem of false positives remains as syllable units still occur much more frequently than words.

Ideally we would like to develop systems that have the low OOV rates of lattice based systems while maintaining the good scalability, speed of search, and low false alarm rate properties of word-based of index systems. This paper represents our first attempts in this direction. In the following sections we study three approaches to audio indexing: a word-based system, a novel *particle*-based system and a phoneme-based system. The particle system is syllable-like with particles consisting of automatically determined within-word sequences of phonemes [10]. Our hope is that it can find OOV queries with less false positives than the phoneme system. We also explore some simple schemes that combine the three different indexing approaches.

2. EXPERIMENTAL CONDITIONS

We first describe our experimental setup.

2.1 Audio Databases

We use spoken audio transcribed by the Linguistic Data Consortium (LDC) [1] for our experiments. The transcripts provide us with the ground truth and allow us to estimate precision, recall and false alarm rates. The audio is from broadcast sources and is sampled at 16kHz. For training acoustic models we use 65 transcribed hours of the HUB4_96 training set. Our indexing experiments are performed on about 75 hours of audio composed of the HUB4_96

*Om Deshmukh is a graduate student at the University of Maryland.

HLT 2002, San Diego, California, USA
.

development and test data and the HUB4_97 training and test data.

2.2 Document and Relevance Definitions

In our system, we index hours-long streams of audio which do not have labeled topic boundaries. Since returning the whole stream is meaningless, we arbitrarily define documents as 10 second segments of audio, similar to the *clips* returned by the *SpeechBot*[1] user interface. If the index returns a clip in which the query word/s were spoken, the query is judged successful.

2.3 Evaluation Metrics

Our primary evaluation metric is 11-pt average precision. This is an estimate of the area under a recall-precision curve. The greater this area, the better the system[2]. It is an overall measure of the quality of a retrieval system, incorporating recall and precision.

Because we are examining sub-word-based systems for which false-positives are a major problem, we also explicitly report the number of false positives even though 11-pt average precision implicitly includes this quantity. The number of false positives for a given query is defined as the number of incorrect hits divided by the total number of hits returned. We average our results over all queries.

For completeness, we also show recall, top 5 precision, and top 10 precision. These measures are also implicitly included in 11-pt average precision since it is an overall figure of merit. For all metrics we average over all queries.

2.4 Query Selection

In [2] it is recommended that at least 25 and preferably 50 queries are used for an evaluation for which average precision is the metric. We therefore use 50 queries. Our aims in query selection are:

- to use proper names for which relevance can be determined automatically;
- to have a high proportion of OOV queries;
- to use 'real-world' queries;
- to have at least 10 hits per query, similar to a Web page of hits.

Comparison of the ground truth to the dictionary used for word recognition yields 23 suitable single word OOV queries (*i.e.* proper names with at least 10 hits). We choose the remaining 27 queries as the most frequent in-vocabulary queries to the *SpeechBot* public site which have at least 10 hits and are proper names. The majority of these queries are single-word queries with only three queries having two words. The *SpeechBot* public site has been in operation for over 18 months and is therefore a reasonable source of real-world queries. Note that our query OOV rate of about 50% is much higher than the 13% rate observed on the *Speechbot* site[6]. The queries and the number of documents in which they appear are listed in Table 4 in Appendix A.

2.5 Indexing Systems

We investigate three systems. The first uses a large vocabulary speech recognizer with a 70,000 word dictionary and a trigram language model, similar to the recognizer used by *SpeechBot* [9]. Its acoustic models are trained on the 65 hour training set. The language model is trained on the transcriptions for this set and additional text sources.

The second system uses our novel particle recognizer [10]. Particles are defined as within-word sequences of characters obtained from orthographic or phonetic transcriptions of words. Our particles are obtained from phonetic transcriptions. They are learned automatically from data. Specifically, they are determined by decomposing words into sub-sequences of phonemes so as to maximize the leaving-one-out likelihood of a particle bigram language model.

The particle dictionary consists of phoneme sequences from single phonemes to full words. Once the dictionary of valid particles is defined the word text corpora is translated into particles. This new particle corpora is used then for training a traditional language model of particles where unigram, bigram and trigram probabilities with back-off weights are learned from data. A similar translation is performed on the transcripts of the acoustic corpora and triphone based large vocabulary acoustic models are then built.

The particles representation is quite flexible. If the dictionary of particles only contains single phoneme particles then the particle recognizer behaves like a phonetic recognizer. If the particles are as long as words then it behaves like a word recognizer. In our implementation we use a dictionary of about 7,000 particles. We have found that this dictionary size with particles of length from one to three phonemes yields optimal results. In effect a particle based recognition system behaves like a syllable based speech recognizer where the basic units are automatically learned from textual data. In our system we use the same audio and text corpora for training as for the word-based recognizer.

Finally, our third system indexes phoneme sequences. We do not run a phoneme recognizer. Instead, we use a dictionary to automatically convert the transcripts from the word recognizer in our first system to phonemes. Preliminary tests indicated that this gives better results than running a phoneme recognizer.

The 75 hours of audio used for testing were transcribed by each system. The time-marked words, particles and phonemes were then fed into three separate indexes. Our index is the same as that used in *SpeechBot* which is a derivative of the AltaVista index [3]. We choose this index, which can quickly handle boolean queries, rather than a standard vector-space index because we are interested in Web-based systems which must scale to millions of queries per day.

3. RESULTS

Figure 1 and the first three lines of Table 1 show the performance of each system averaged over all queries. We see that the word-based system has the best performance overall. However, as Figures 2 and 3 and Tables 2 and 3 demonstrate, the performance of the particle and phoneme systems are worse than the word system for in-vocabulary queries and better for OOV queries. The particle system performs slightly better than the phoneme system on OOV queries.

Since particles are syllable-like, we investigate indexing sequences of phonemes as described in [11]. Specifically, we investigate indexing sequences of from 3-5 phonemes with from 1-4 phonemes overlap. The fourth line of Tables 1, 2 and 3 and Figures 4, 5 and 6 show results for the best of these systems. This indexed phoneme sequences of length 5 with overlap 4. In the figures and tables it is referred to as the 'Phoneme (5/4)' system.

From these results we see that using sequences of phonemes can improve the average precision. The system is at least as good as using particles for OOV words and equivalent to words overall.

However, although both the word index and phoneme sequence index have an average precision of 0.35, they operate at different recall and false positive levels. From Table 1, we see that using phoneme sequences rather than words improves the recall from 0.39 to 0.48. However, this comes at a cost of increasing the number of false positives from 0.08 to 0.57. In some applications this

[1] *http://www.speechbot.com*

[2] An ideal system would have precision 1.0 for all recall values i.e. every document retrieved would be relevant.

increase in false positives could be crippling. In others it might be justified by the increase in recall.

Finally, we consider combining the word-based and best phoneme-based indexes. We consider two simple strategies:

- linearly combining the scores;
- selecting the word index for in-vocabulary queries and the phoneme (5/4) index otherwise.

The last two lines of Table 1, 2 and 3 show the results of these experiments. Both strategies result in improved average precision.

For the linear combination technique, we report results from an exhaustive search of the space of all possible combination coefficients. This result is therefore an upper bound, obtainable only if the coefficients could be optimized on a development query set. The OOV-based combination technique does not rely on the use of a development set and thus could be recommended for all query types. Its performance is equivalent to the best linear combination of systems and is additionally shown in Figure 4.

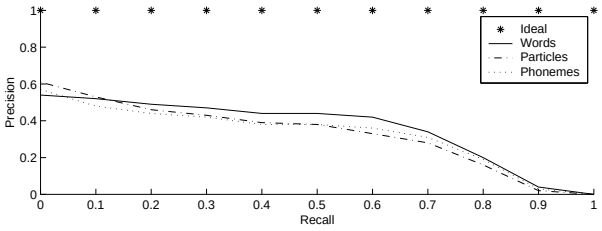

Figure 1: Precision-Recall curves averaged over all queries for the baseline systems and the ideal system.

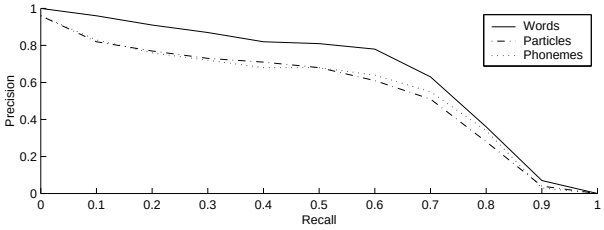

Figure 2: Precision-Recall curves for the in-dictionary queries for the baseline systems.

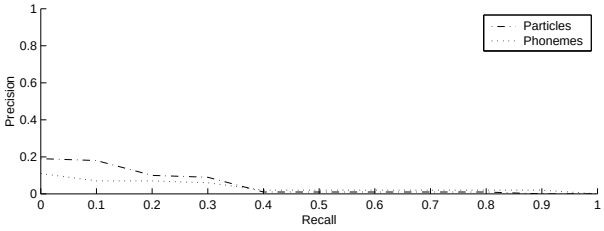

Figure 3: Precision-Recall curves for the OOV queries for the particle and phoneme baseline systems.

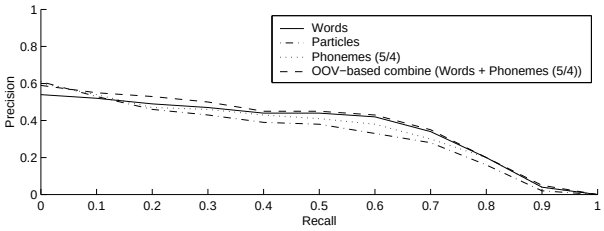

Figure 4: Precision-Recall curves for all queries showing the comparison between the word and particle baseline systems, a system indexing sequences of phonemes and a combination-based system.

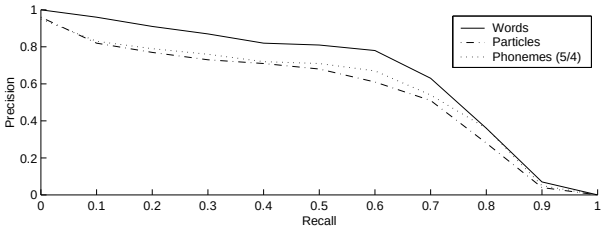

Figure 5: Precision-Recall curves for the in dictionary queries showing the comparison between the word and particle baseline systems and a system indexing sequences of phonemes.

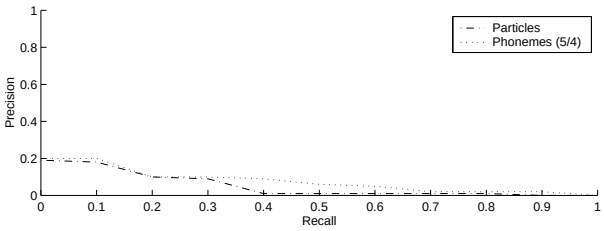

Figure 6: Precision-Recall curves for the OOV queries showing the comparison between the particle baseline system and a system indexing sequences of phonemes.

System	11-pt Avg. Precision	Recall	Top 5 Precision	Top 10 Precision	False Positives
Word	0.35	0.39	0.50	0.48	0.08
Particles	0.33	0.39	0.51	0.47	0.21
Phonemes	0.32	0.42	0.48	0.44	0.27
Phonemes (5/4)	0.35	0.48	0.48	0.45	0.57
Linear combine	0.39	0.48	0.54	0.51	0.57
OOV-based combine	0.39	0.46	0.56	0.53	0.34

Table 1: Results averaged over all queries for various indexing systems.

System	11-pt Avg. Precision	Recall	Top 5 Precision	Top 10 Precision	False Positives
Word	0.66	0.73	0.92	0.89	0.14
Particles	0.55	0.65	0.82	0.79	0.24
Phonemes	0.56	0.71	0.84	0.77	0.29
Phonemes (5/4)	0.58	0.76	0.80	0.76	0.56
Linear combine	0.66	0.76	0.92	0.87	0.14
OOV-based combine	0.66	0.73	0.92	0.89	0.14

Table 2: Results averaged over all in-dictionary queries for various indexing systems.

4. CONCLUSIONS

We have explored the problem of OOV query words in audio indexing, comparing systems based on words, particles and phonemes. While word based systems are fast and have overall good IR performance they cannot work in the presence of OOV queries. On the other hand phonetic based systems offer generally lower IR performance but can partially recover some of the audio documents with OOV queries. Finally, particles based systems exhibit an intermediate behavior both in performance and in complexity.

The IR performance of a phonetic based system can be improved by translating the word queries into a phonetic representation and creating phone sequences 5 phonemes long with a 4 phoneme overlap. However, there is a cost for this improvement. The false alarm rate, a metric often overlooked in IR, is significantly increased. False alarms are implicitly included in traditional IR precision/recall metrics. However, by explicitly measuring it we obtain a more complete view of the performance of these three different systems. In many applications high false alarm rates can render the system unusable or at the very least have a high impact on the perceived usability of the system.

Clearly no approach stands on its own as the correct answer to the audio document IR problem. The combination of word and subword indexing systems perhaps offers us a third approach. In this paper we have made a preliminary study on combining these systems with promising results. Even the simplest approach of detecting the query word as OOV and using the phonetic or particle system for that query works as well as using an optimal weighting scheme. In the future we would like to explore more sophisticated index combination techniques based on data fusion and Bayesian mixing of classifiers.

It is also important to note that the particle based system has not been deeply explored in this paper. We consider our reported results quite preliminary and intend to investigate this approach more in the future. Finally, we have not explored traditional IR techniques such as query expansion and relevance feedback. The role of these approaches in OOV retrieval remains an alternative route for further study.

5. REFERENCES

[1] S. Bird and M. Liberman. Linguistic annotation resources. University of Pennsylvania. See http://www.ldc.upenn.edu/annotation.

[2] C. Buckley and E. Voorhees. Evaluating evaluation measure stability. In *SIGIR2000*, 2000.

[3] M. Burrows. *Method for Indexing Information of a Database*. U.S. Patent 5,745,899, 1998.

[4] M. Clements, P. S. Cardillo, and M. S. Miller. Phonetic searching vs. LVCSR: How to find what you really want in audio archives. In *20th Annual AVOIS Conference*, 2001.

[5] D. A. James. A system for unrestricted topic retrieval from radio news broadcasts. In *Proc. ICASSP*, 1994.

[6] B. Logan, P. Moreno, J. V. Thong, and E. Whittaker. An experimental study of an audio indexing system for the Web. In *Proc. ICSLP*, 2000.

[7] K. Ng and V. Zue. Towards robust methods for spoken document retrieval. In *Proc. ICSLP*, 1998.

[8] S. Srinivasan and D. Petkovic. Phonetic confusion matrix based spoken document retrieval. In *SIGIR2000*, 2000.

[9] J. V. Thong, D. Goddeau, A. Litvinova, B. Logan, P. Moreno, and M. Swain. Speechbot: a speech recognition based audio indexing system for the web. In *Proc. RIAO*, 2000.

[10] E. W. D. Whittaker, J. Van Thong, and P. J. Moreno. Vocabulary independent speech recognition using particles. In *ASRU 2001*, 2001.

[11] M. Witbrock and A. G. Hauptmann. Using words and phonetic strings for efficient information retrieval from imperfectly transcribed spoken documents. In *Second ACM International Conference on Digital Libraries*, 1997.

System	11-pt Avg. Precision	Recall	Top 5 Precision	Top 10 Precision	False Positives
Word	0.00	0.00	0.00	0.00	0.00
Particles	0.06	0.09	0.15	0.10	0.17
Phonemes	0.04	0.08	0.07	0.05	0.24
Phonemes (5/4)	0.08	0.14	0.10	0.09	0.58
Linear combine	0.08	0.14	0.10	0.09	0.58
OOV-based combine	0.08	0.14	0.10	0.09	0.58

Table 3: Results averaged over all OOV queries for various indexing systems.

APPENDIX

A. LIST OF QUERIES

In dictionary	Count	Out of Dictionary	Count
bill clinton	56	cunanan	70
al gore	31	mair	57
clinton	626	fayed	52
microsoft	40	dodi	37
israel	104	tamraz	26
egypt	15	peekskill	23
montreal	23	sankara	18
china	226	plavsic	18
nasdaq	53	reineck	13
paris	101	rutan	16
christmas	97	fenphen	16
jesus	11	lia	13
kennedy	48	mcaleese	14
france	62	bilbao	13
england	86	reesjones	13
germany	37	cortisol	10
switzerland	13	onondaga	10
india	39	hightech	12
nasa	73	zorich	12
australia	25	liderman	12
mexico	121	montserrat	11
cuba	141	boughton	10
florida	198	pazuto	10
canada	106		
iran	66		
texas	151		
stock market	41		

Table 4: Queries to the system