AT&T at TREC-6: SDR Track

Amit Singhal, John Choi, Donald Hindle, Fernando Pereira
AT&T Labs – Research
{singhal,choi,hindle,pereira}@research.att.com

Abstract

In the *spoken document retrieval* track, we study how higher word-recall—recognizing many of the spoken words—affects the retrieval effectiveness for speech documents, given that high word-recall comes at a cost of low word-precision—recognizing many words that were not actually spoken. We hypothesize that information retrieval algorithms would benefit from a higher word-recall and are robust against poor word-precision. Start-up difficulties with recognition for this task kept us from doing an systematic study of the effect of varying levels of word-recall and word-precision on retrieval effectiveness from speech. We simulated a high word-recall and a poor word-precision system by merging the output of several recognizers. Experiments suggest that having higher word-recall does improve the retrieval effectiveness from speech.

1 Introduction

From a retrieval system's perspective, a speech recognizer makes two types of recognition errors:

- **Omissions**: a spoken word is not recognized, and
- **Delusions**: the recognizer recognizes a word that was not spoken.

All recognition errors can be attributed to the above two types of errors, or their combination. *Omissions* reduce the word-recall, where word-recall is defined as the proportion of spoken words that are recognized; whereas *delusions* reduce the word-precision, where word-precision is defined as the proportion of recognized words that were spoken.

When speech-retrieval is done using word-based IR techniques, we hypothesize that omissions are much more hurtful than delusions. We believe that our IR techniques are quite robust against "noise" in the input text, given that there is enough "signal" in the text. High word-recall contributes to high signal in the text and high word-precision leads to low noise in the text. Therefore we want to study the effect of varying levels of word-recall and word-precision on retrieval effectiveness for speech.

Based on above hypothesis, we would like to enhance word-recall (by reducing omissions) at the cost of poorer word-precision. Two factors are responsible for omissions by a recognizer:

- Poor recognition: Often poor acoustics or language model constraints do not allow the recognizer to hypothesize a word with a reasonable confidence, even though the word is in the recognizer's vocabulary.
- Out of vocabulary (OOV): The spoken word is not in the recognizer's vocabulary, thus could never be recognized.

Using a word-based recognition system, we cannot attack the OOV problem, but we can certainly attack the other problem by generating many more words that are suggested by a recognizer even with a low confidence, and using these words for retrieval. As a recognizer suggests more and more words for a speech segment, the word-recall should improve but the word-precision should become poorer.

An attack on the OOV problem is to perform retrieval on sub-word acoustic units (phones, demi-syllables, syllables, or sequences of these units). [1, 5] For example, one might use all phone trigrams in the one-best phone transcription of the speech as the indexing units for an IR system. A user query could also be translated into a bag of phone trigrams [1]. Given that even the best phone recognizers make a large number of mistakes, to improve phone trigram recall, we can once again use phone lattices to obtain the bag of phone trigrams for each speech document. Once the recognizer outputs a phone lattice, we can simply use all possible three-phone sequences in the lattice as indexing units. A similar lattice-based approach can be used for any class of indexing units, for example syllable or demi-syllable sequences.

2 Initial Plans

Since we did not already have a recognizer trained on HUB-4 material, we were relatively unconstrained with respect to recognizer design, so we set out to build a system that would attack both the word-recall and the OOV problems. We thus decided to implement a syllabic lattice recognition system, using existing training, recognition, syllabification and language-modeling programs. However, given that we started the work on our recognizer in late June, having a complete system running before the SDR deadline was quite an ambitious task.

For syllabic language modeling, we create a word list, generate word pronunciations with a text-to-speech system, and we apply a simple rule-based maximal onset syllabifier to the result to create a translation table from words to syllables. Since the position of a syllable within a word is quite informative for language modeling, we use four position-marked versions of each syllable: word-initial, word-medial, word-final and monosyllabic word. The resulting translation table from words to position-marked syllables is then used to translate the language-model training text into syllable sequences from which the appropriate n-gram statistics are computed.

To retrieve from syllabic-recognition output, the query words would be syllabified and the resulting syllable n-grams used to look up documents also indexed by syllable n-grams from built from the corresponding recognizer output. However, various difficulties described in the next section prevented us from having the full results of syllabic recognition in time for the track deadline, and we ended up using a simplified approach described later.

3 Recognizer

For recognition, we used phone-based models, a single-pronunciation dictionary, and a syllable bigram backoff language model.

For phone models, we used 3-state, left-to-right, HMMs with triphonic context dependence, trained on 39-dimensional acoustic feature vectors of mel-frequency cepstral coefficients and their first and second time derivatives centered on 5 and 3 frame windows, respectively. These vectors were initially modeled by a single full covariance Gaussian pdf (probability distribution function) per state, which was then rotated using the eigenvectors of the covariance matrix to remove correlations between parameters. Decorrelation was followed by the estimation of a weighted mixture of Gaussian pdfs with diagonal covariance. [3]

Context-dependency was modeled using categorical decision trees based on sub-phonemic classes, which effectively results in context-dependent tying of states. The decision trees were trained only on the training speech. A separate context-dependency model was defined for each training partition.

We built three separate sets of models: one set from speech labeled as high-fidelity with no background noise, one from medium and low fidelity speech with no background noise, and one from speech labeled as having background noise. In training each of the three sets of models, we bootstrapped from a single model trained on the channel-1 data from the NAB corpus.

For language modeling, we used a standard backoff bigram language model [2] over a vocabulary of about 20,000 position-marked syllables. This vocabulary size was chosen as a compromise between expected recognition speed and OOV rate. On the development test partition, a 20,000 word vocabulary yields an OOV rate 1.7%, while that for syllables is 0.4%. Position-marked syllables are represented by the their constituent phones together with a word boundary symbol, which is used in reconstructing words from the recognized

[1] For written queries, a text-to-speech system can be used to obtain the phone string corresponding to the query.

	IBM	AT&T+IBM
Word-Recall	69.3%	82.1%
Word-Precision	65.6%	18.9%

Table 1: Word-recall and word-precision for IBM's transcription and the merged transcription.

syllables. So, for example, the syllable that is typically spelled "bob" appears in the syllable wordlist as four distinct entries – `#_B_aa_B_#, #_B_aa_B, B_aa_B, B_aa_B_#` – corresponding to its appearance in the four words "Bob", "bobcat", "discombobulate", and "shishkabob", respectively.

The language model was trained on the SDR training corpus and the data from transcribed news broadcasts, designated for use in the baseline language model (LM) for the 1996 CSR Hub-4 evaluation. The syllable inventory was defined using all pronunciation alternates generated by our text-to-speech system. All syllables in the SDR training corpus were included in the syllable inventory, and all syllables with frequency greater than 3 in the Hub-4 LM corpus were included. To train the model, each word of the training text was mapped into its component syllables (including the word boundary symbols); for words with multiple pronunciations, a single alternate was randomly chosen for each occurrence.

4 Submitted Runs

Since this was our first experience with this particular material (AT&T had not participated in the HUB4 evaluations) and with a such a large material to be recognized, we encountered several difficulties that seriously curtailed our original experimental design.

First, we did not have at the time a reliable enough means of segmenting the test material into reasonably-sized segments of uniform type that could then be given to the appropriate one of our three recognizers (high quality, mid-low quality, and noisy). Therefore, we had to adopt the expedient of segmenting the test material into evenly-sized overlapping segments, and running all three recognizers on each segment. Second, the lattice recognizers that we had at the time were too slow to be able to recognize the whole test material in the available time and computing resources. Finally, the time and resources available to us were eroded further by a slew of unexpected systems problems.

Therefore, to submit a run we had to scale back our plans radically. Instead of lattice recognizers, we ran one-best recognizers (2-3 times real time) for the three models on all the test segments. Furthermore, even though we had all the machinery in place for extracting indexing units — syllable n-grams — from lattices, this machinery was not of any use for one-best transcriptions.

Both the "ad hoc" segmentation and the limited predictive power of the bi-syllable language model certainly contributed to the resulting poor recognition accuracy. While the segmentation into overlapping segments prevented us for computing word-error rates precisely, we estimated the word-error rate as high as 60%.[2]

Given all the problems we had with the recognizer, we had not time left to test our syllabic retrieval system. So we had to give up on attacking the OOV problem and revert back to using English words for retrieval. But since our recognition was syllabic, we had to translate all the "syllabic words" (mono-syllabic words and any syllable sequence that starts with a word-starting syllable, has any number of word-medial syllables, and ends in a word-ending syllable) into all possible English words using a pronunciation dictionary. This resulted in a mono-syllabic word `#_s_eh_n_t_#` generating the English words `cent, scent`, and `sent`. We applied this transformation to the recognizer output from each of our three acoustic models, resulting in three homophone-rich wordlists for every story. We then merged all three lists to get the final text to be indexed for a story, forming a coarse simile of a lattice.

The first run **att97sS1** was done using this merged list of words as a document and the user queries. To further simulate lattices, we created another set of words for every document by further merging the above merged list of words with the words that appeared in IBM's transcription of the speech. Our second retrieval run **att97sS2** was done using this longer list of words for a document, with higher word-recall and poorer

[2] In contrast, with a recently developed segmenter and a 20,000-word bigram model, the word-error rate went down to 40% even without changing the acoustic models.

word-precision. Table 1 shows the recall and the precision figures for the baseline (IBM's) transcription, and the merged (AT&T+IBM) transcription used in att97sS2. These figures were computed using non-stop words (because only they matter in retrieval), and by ignoring word frequency (since we use binary tf weighting). The word-recall and the word-precision was computed for every story and was further averaged across stories. We observe that the merged list does exhibit a higher word-recall and has a much poorer word-precision than IBM's transcription. Our main motivation for doing this merging was that if the merged retrieval run works better than both att97sS1, and att97sB1 (which is a retrieval run done solely on IBM's baseline word transcriptions), then our hypothesis that improving word-recall should help speech retrieval effectiveness will be supported.

We use an internally modified version of Cornell's SMART system for retrieval. We used standard inner-product similarity to rank the *bnu* weighted documents using *ltu* weighted queries within the SMART system. [4] Where the weight of a word in a document (bnu) is:

$$\frac{1}{0.8 + 0.2 \times \frac{number\ of\ unique\ words\ in\ document}{average\ number\ of\ unique\ words\ per\ document}}$$

and the weight of a query word is (ltu):

$$\frac{1 + log(tf) \times log(\frac{N+1}{df})}{0.8 + 0.2 \times \frac{number\ of\ unique\ words\ in\ query}{average\ number\ of\ unique\ words\ per\ document}}$$

5 Results

Out of the three evaluation measures being used for known-item searching — mean rank, mean reciprocal rank, and counts of how many known items were found within top 5, 10, 20 and 100 documents — the first two (mean rank and mean reciprocal rank) have problems in our view. Mean rank is heavily influenced by even a single miss (very poorly ranked document, an outlier). For example, if the known item for a query is ranked 200, the mean rank for the entire collection of 49 queries drops by almost 4, irrespective of how well the system is retrieving for the other 48 queries. However, if outliers are removed, *i.e.*, all queries for which a system has extremely poor results (under some definition of extremely poor), then average rank might yield meaningful results. Mean reciprocal rank, on the other hand, differentiates too much between a known-item being ranked at rank 1 vs. if the known-item is ranked at rank 2. From a user's perspective, we believe that ranking the known-item at rank 1 is not 100% better than ranking it at rank 2, a ratio assigned by mean reciprocal rank. We believe that counts of how many known items were found within top 5, 10, 20 and 100 documents is the most meaningful measure out of the above three evaluation measures. If one has to compare only two runs, another meaningful comparison would be a query-by-query comparison of the two systems on a scatter plot. This would enable us to view which system is performing better on most of the queries and by how much.

Figure 1 shows a histogram of how document are ranked when different texts — the human transcription (Human), IBM's transcription (IBM), our merged list of words (AT&T), and our list merged with IBM's words (AT&T+IBM) — are used in retrieval for the 49 user queries. Our first observation from Figure 1 is that retrieval done over the output of a speech recognizer using conventional IR techniques is quite respectable. This agrees with the observation of other researchers who have worked with other speech corpora. As expected, our internal recognition does not perform as well as the other transcriptions. We are actually surprised that it works as well as it does. Given the recognition difficulties described above, it is somewhat surprising that our system still retrieves thirty three answer documents within top five using our merged list of words, suggesting that the task at hand was rather easy.

More interestingly, we observe that once we merge the baseline transcription provided by IBM and our list of words, even though we retrieve the answer document in the top five documents for fewer queries (which we believe is a reflection upon the poor quality of our recognition), if we look in the top ten documents, retrieval from the merged transcription (AT&T+IBM) outperforms retrieval from IBM's transcription alone. This is true even when we look in the top twenty documents. Actually, when looked in the top twenty documents, the merged transcription works as well as the human transcription. Forty six out of forty nine queries have their

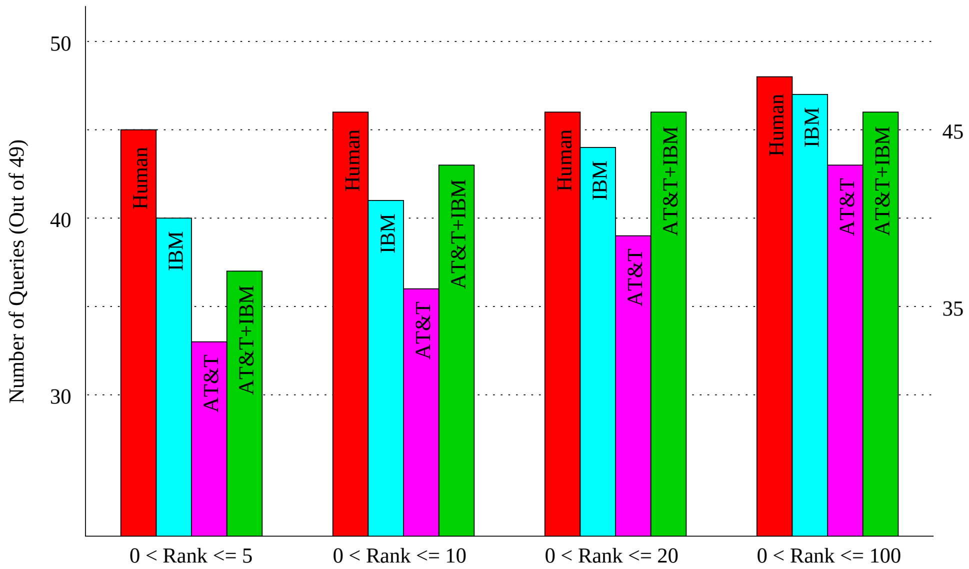

Figure 1: Comparison of retrieval from various transcriptions.

answers listed in the top twenty for retrieval from both the human and the merged transcription. Of course, where the answer is within the top twenty is also important. We believe that if we had a better transcription internally, these results could have been better. This results are encouraging for further experimentation using word and sub-word lattices.

If we remove the outlier queries, *i.e.*, query 3 (for which the answer article is ranked at ranks 236, 389, and 178 for the human, IBM, and AT&T+IBM transcriptions, respectively), query 23 (known item rank is 222 for AT&T+IBM), and query 42 (IBM's transcription does not retrieve the answer at all), then the average rank of the known item for the human, IBM, and AT&T+IBM transcription are 2.65, 4.15, and 3.04, respectively. This once again indicates that retrieval from AT&T+IBM transcription is somewhat better than retrieval from IBM's transcription alone. This lends further support to possibility of improved retrieval using lattices.

Another evidence that retrieval from AT&T+IBM transcription is better than the retrieval from the IBM's transcription alone is shown in Figure 2. Figure 2 shows what the rank of an answer document is using the AT&T+IBM transcription vs. the rank of the corresponding document using IBM's transcription alone. The x-axis is the rank of the answer document as retrieved from IBM's transcription (log-scale), and the y-axis is the rank of the answer document as retrieved from AT&T+IBM transcription (log-scale). A point below the diagonal line indicates that the rank of the answer document was lower (better retrieval) for the AT&T+IBM transcription. This scatter plot shows that, in general, the merged transcription has better results. 24 of the 49 queries have their known-item retrieved at identical ranks for the two system. For 16 queries, retrieval from AT&T+IBM transcription is better, and for 9 queries retrieval from AT&T+IBM transcription is worse than retrieval from IBM's transcription alone.

6 Directions

We have recently finished implementing a fast lattice recognizer, and are currently in the process of training new acoustic models. We have also developed a speech segmenter internally that assigns portions of the test speech to one of several possible acoustic categories in our system. We plan to investigate lattice based recognition in a much more organized manner in the near future.

Even though known-item retrieval is a fine task for initial evaluation of speech retrieval system, the small size of speech corpora (as compared to more traditional information retrieval corpora) makes this task artificially easy. There is very little noise in the corpora. Any user query hits just a few documents, if at

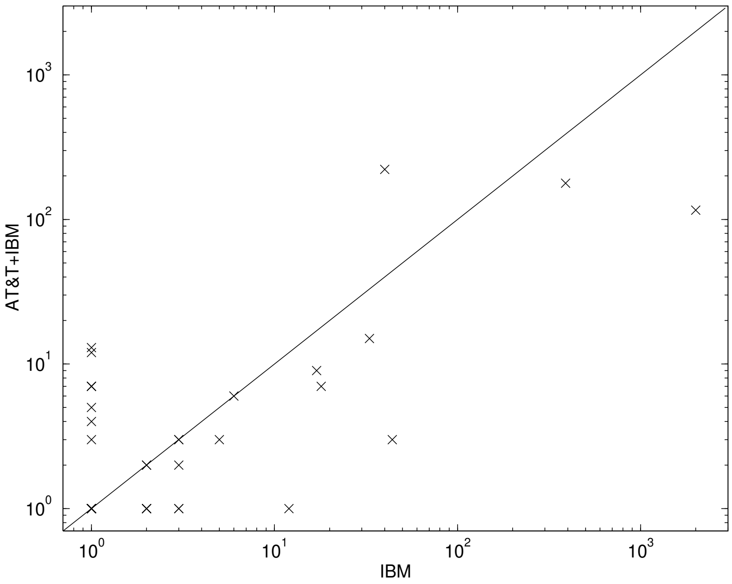

Figure 2: Comparison of retrieval from IBM's transcription and AT&T+IBM transcription.

all it hits any. Therefore, larger speech databases are always desirable in a speech retrieval task. Moving to a more traditional, ranking evaluation using average precision might also exemplify some strengths and shortcomings of various approaches of speech retrieval.

Acknowledgments

We are very grateful to Andrej Ljolje, Mehryar Mohri, and Michael Riley for all their help in building the recognizer for this data.

References

[1] G.J.F. Jones, J.T. Foote, K. Sparck Jones, and S.J. Young. Retrieving spoken documents by combining multiple index sources. In Hans-Peter Frei, Donna Harman, Peter Schauble, and Ross Wilkinson, editors, *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 30–38. Association for Computing Machinery, New York, August 1996.

[2] S.M. Katz. Estimation of probabilities from sparse data from the language model component of a speech recognizer. *IEEE Transactions of Acoustics, Speech and Signal Processing*, pages 400–401, 1987.

[3] Andrej Ljolje. The importance of cepstral parameter correlations in speech recognition. *Computer Speech and Language*, 8, 1994.

[4] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In Hans-Peter Frei, Donna Harman, Peter Schauble, and Ross Wilkinson, editors, *Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–29. Association for Computing Machinery, New York, August 1996.

[5] M. Wechsler and P. Schauble. Indexing methods for a speech retrieval system. In C.J. van Rijsbergen, editor, *Proceedings of the MIRO Workshop*, 1995.