Modeling Topic Coherence for Speech Recognition

Satoshi Sekine
Computer Science Department
New York University
715 Broadway, 7th floor
New York, NY 10003, USA
sekine@cs.nyu.edu

Abstract

Statistical language models play a major role in current speech recognition systems. Most of these models have focussed on relatively local interactions between words. Recently, however, there have been several attempts to incorporate other knowledge sources, in particular longer-range word dependencies, in order to improve speech recognizers. We will present one such method, which tries to automatically utilize properties of topic continuity. When a base-line speech recognition system generates alternative hypotheses for a sentence, we will utilize the word preferences based on topic coherence to select the best hypothesis. In our experiment, we achieved a 0.65% improvement in the word error rate on top of the base-line system. It corresponds to 10.40% of the possible word error improvement.

1 Introduction

Statistical language models play a major role in current language processing applications. Most of these models have focussed on relatively local interactions between words. In particular, large vocabulary speech recognition systems have used primarily bi-gram and tri-gram language models. Recently, however, there have been several attempts to incorporate other knowledge sources, and in particular longer-range word dependencies, in order to improve speech recognizers. Here, 'longer-range dependencies' means dependencies extending beyond several words or beyond sentence boundaries.

There have been several attempts in the last few years to make use of these properties. One of them is the "cache language model" (Kuhn, 1988) (Jelinek et al., 1991) (Kupiec, 1989). This is a dynamic language model which utilizes the partially dictated document ("cache") in order to predict the next word. In essence, it is based on the observation that a word which has already appeared in a document has an increased probability of reappearing. Jelinek showed the usefulness of this method in terms of speech recognition quality. For short documents, however, such as newspaper articles, the number of words which can be accumulated from the prior text will be small and accordingly the benefit of the method will generally be small.

Rosenfeld proposed the "trigger model" to try to overcome this limitation (Rosenfeld, 1992). He used a large corpus to build a set of "trigger pairs", each of which consists of a pair of words appearing in a single document of a large corpus. These pairs are used as a component in the probabilistic model. If a particular word w appears in the preceding text of document, the model will predict a heightened probability not just for w but also for all the words related to w through a trigger pair.

Our approach can be briefly summarized as follows. The topic or subject matter of an article influences its linguistic properties, such as word choice and co-occurrence patterns; in effect it gives rise to a very specialized "sublanguage" for that topic. We try to find the sublanguage to which the article belongs based on the sentences already recognized. At a certain stage of the speech recognition processing of an article, words in the previous utterances are selected as keywords. Then, based on the keywords, similar articles are retrieved from a large corpus by a method similar to that used in information retrieval. They are assembled into a sublanguage "mini-corpus" for the article. Then we analyze the mini-corpus in order to determine word preference which will be used in analyzing the following sentence. The details of each step will be described later.

Our work is similar to that using trigger pairs. However, the trigger pair approach does a very broad search, retrieving articles which have *any* word in common with the prior discourse. Our approach, in contrast, makes a much more focussed search, taking only a small set of articles most similar to the prior discourse. This may allow us to make sharper predictions in the case of well-

defined topics or sublanguages, and reduce the problems due to homographs by searching for a conjunction of words. (Rosenfeld has indicated that it may be possible to achieve similar results by an enhancement to trigger pairs which uses multiple triggers (Rosenfeld, 1992).) In addition, our approach needs less machine power. This was one of the major problems of Rosenfeld's approach.

Sekine has reported on the effectiveness of sublanguage identification measured in terms of the frequency of overlapping words between an article and the extracted sublanguage corpus (Sekine, 1994). In this paper, we report on its practical benefits for speech recognition.

2 Speech Recognition System

This research is being done in collaboration with SRI, which is providing the base of the combined speech recognition system. (Digalakis et.al., 1995). We use the N-best hypotheses produced by the SRI system, along with their acoustic and language model scores. There are two acoustic scores and four language scores. Language scores are namely the word trigram model, two kinds of part of speech 5-gram model and the number of tokens. Note that none of their language models take long-range dependencies into account. We combine these scores with the score produced by our sublanguage component and our cache model score, and then select the hypothesis with the highest combined score as the output of our system. The system structure is shown in Figure 1. The relative weights of the eight scores are determined by an optimization procedure on a training data set, which was produced under the same conditions as our evaluation data set, but has no overlap with the evaluation data set. The actual conditions will be presented later.

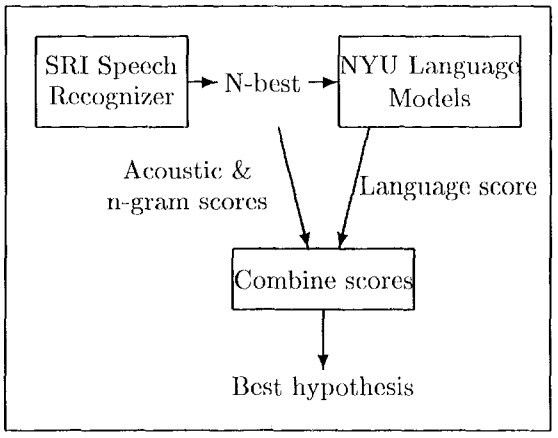

Figure 1: Structure of the system

3 Sublanguage Component

The sublanguage component performs the following four steps:

1. Select keywords from previously uttered sentences
2. Collect similar articles from a large corpus based on the keywords
3. Extract sublanguage words from the similar articles
4. Compute scores of N-best hypotheses based on the sublanguage words

A sublanguage analysis is performed separately for each sentence in an article (after the first sentence). There are several parameters in these processes, and the values of the parameters we used for this experiment will be summarized at the end of each section below. We generally tried several parameter values and the values shown in this paper are the best ones on our training data set.

We used a large corpus in the experiment as the source for similar articles. This corpus includes 146,000 articles, or 76M tokens, from January 1992 to July 1995 of North American Business News which consists of Dow Jones Information Services, New York Times, Reuters North American Business Report, Los Angeles Times, and Washington Post. This corpus has no overlap with the evaluation data set, which is drawn from August 1995 North American Business News.

Now, each step of our sublanguage component will be described in detail.

Select Keywords

The keywords which will be used in retrieving similar articles are selected from previously dictated sentences. The system we will describe here is an incremental adaptation system, which uses only the information the system has acquired from the previous utterances. So it does not know the correct transcriptions of prior sentences or any information about subsequent sentences in the article.

Not all of the words from the prior sentences are used as keywords for retrieving similar articles. As is the practice in information retrieval, we filtered out several types of words. First of all, we know that closed class words and high frequency words appear in most of the documents regardless of the topic, so it is not useful to include these as keywords. On the other hand, very low frequency words sometimes introduce noise into the retrieval process because of their peculiarity. Only open-class words of intermediate frequency (actually frequency from 6 to 100000 in the corpus of 146,000 articles) are retained as keywords and used in finding the similar articles. Also, because the N-best sentences inevitably contain errors, we set a threshold for the appearance of words in the N-best sentences. Specifically, we require that a

word appear at least 15 times in the top 20 N-best sentences (as ranked by SRI's score) to qualify as a keyword for retrieval.

Parameter	Value
Max frequency of a keyword	100000
Min frequency of a keyword	6
N-best for keyword selection	20
Min word appearances in N-best	15

Collect Similar Articles

The set of keywords is used in order to retrieve similar articles according to the following formulas. Here $Weight(w)$ is the weight of word w, $F'(w)$ is the frequency of word w in the 20 N-best sentences, M is the total number of tokens in the corpus, $F(w)$ is the frequency of word w in the corpus, $AScore(a)$ is article score of article a, which indicates the similarity between the set of keywords and the article, and $n(a)$ is the number of tokens in article a.

$$Weight(w) = F'(w) * log(\frac{M}{F(w)}) \quad (1)$$

$$AScore(a) = \frac{\sum_{w \in a} Weight(w)}{log(n(a))} \quad (2)$$

Each keyword is weighted by the product of two factors. One of them is the frequency of the word in the 20 N-best sentences, and the other is the log of the inverse probability of the word in the large corpus. This is a standard metric of information retrieval based on the assumption that the higher frequency words provide less information about topics (Sparck-Jones, 1973). Article scores ($AScore$) for all articles in the large corpus are computed as the sum of the weighted scores of the selected keywords in each article, and are normalized by the log of the size of each article. This score indicates the similarity between the set of keywords and the article. We collect the most similar 50 articles from the corpus. These form the "sublanguage set", which will be used in analyzing the following sentence in the test article.

Parameter	Value
Number of articles in sublanguage set	50

Extract Sublanguage words

Sublanguage words are extracted from the collected sublanguage articles. This extraction was done in order to filter out topic-unrelated words. Here, we exclude function words, as we did for keyword selection, because function words are generally common throughout different sublanguages. Next, to find strongly topic related words, we extracted words which appear in at least 3 out of the 50 sublanguage articles. Also, the document frequency in sublanguage articles has to be at least 3 times the word frequency in the large corpus:

$$\frac{DF(w)/50}{F(w)/M} > 3 \quad (3)$$

Here, $DF(w)$ is the number of documents in which the word appears. We can expect that these methods eliminate less topic related words, so that only strongly topic related words are extracted as the sublanguage words.

Parameter	Value
Min num of documents with the word	3
Threshold ratio of word in the set and in general	3

Compute Scores of N-best Hypotheses

Finally, we compute scores of the N-best hypotheses generated by the speech recognizer. The top 100 N-best hypotheses (according to SRI's score) are re-scored. The sublanguage score we assign to each word is the logarithm of the ratio of document frequency in the sublanguage articles to the word frequency of the word in the large corpus. The larger this score of a word, the more strongly the word is related to the sublanguage we found through the prior discourse.

The score for each sentence is calculated by accumulating the score of the selected words in the hypothesis. Here $HScore(h)$ is the sublanguage score of hypothesis h.

$$HScore(h) = \sum_{w \ in \ h} \log(\frac{DF(w)/50}{F(w)/M}) \quad (4)$$

This formula can be motivated by the fact that the sublanguage score will be combined linearly with general language model scores, which mainly consist of the logarithm of the tri-gram probabilities. The denominator of the log in Formula 4 is the unigram probability of word w. Since it is the denominator of a logarithm, it works to reduce the effect of the general language model which may be embedded in the trigram language model score. The numerator is a pure sublanguage score and it works to add the score of the sublanguage model to the other scores.

4 Cache model

A cache model was also used in our experiment. We did not use all the words in the previous utterance, but rather filtered out several types of words in order to retain only topic related words. We actually used all of the "selected keywords" as explained in the last section for our cache model. Scores for the words in cache ($CScore(w)$) are computed in a similar way to that for sublanguage words. Here, N' is the number of tokens in the previously uttered N-best sentences.

$$CScore(h) = \sum_{w \ in \ cache} \log(\frac{F'(w)/N'}{F(w)/M}) \quad (5)$$

5 Experiment

The speech recognition experiment has been conducted as a part of the 1995 ARPA continuous

speech recognition evaluation under the supervision of NIST(NIST, 1996). The conditions of the experiment are:

- The input is read speech of unlimited vocabulary texts, selected from several sources of North American Business (NAB) news from the period 1-31 August 1995
- Three non-close talking microphones are used anonymously for each article
- All speech is recorded in a room with background noise in the range of 47 to 61 dB (A weighted)
- The test involves 20 speakers and each speaker reads 15 sentences which are taken in sequence from a single article
- Speaker gender is unknown

The SRI system, which we used as the base system, produces N-best (with N=100) sentences and six kinds of scores, as they are explained before. We produce two additional scores based on the sublanguage model and the cache model. The two scores are linearly combined with SRI's six scores. The weights of the eight scores are determined by minimizing the word error on the training data set. The training data set has speech data recorded under the same conditions as the evaluation data set. The training data set consists of 256 sentences, 17 articles (a part of the ARPA 1995 CSR "dev test" data distributed by NIST) and does not overlap the evaluation data set.

The evaluation is done with the tuned parameters of the sublanguage component and the weights of the eight scores decided by the training optimization. Then the evaluation is conducted using 300 sentences, 20 articles, (the ARPA 1995 CSR "eval test" distributed by NIST) disjoint from the dev test and training corpus. The evaluation of the sublanguage method has to be done by comparing the word error rate (WER) of the system with sublanguage scores to that of the SRI system without sublanguage scores.

Inevitably, this evaluation is affected by the performance of the base system. In particular, the number of errors for the base system and the minimum number of errors obtainable by choosing the N-best hypotheses with minimum error, are important. (We will call the latter kinds of error "MNE" for "minimal N-best errors".) The difference of these numbers indicates the possible improvement we can achieve by rescoring the hypotheses using additional components.

We can't expect our sublanguage model to fix all of the 375 word errors (non-MNE). For one thing, there are a lot of word errors unrelated to the article topic, for example function word replacement ("a" replaced by "the"), or deletion or insertion of topic unrelated words (missing "over"). Also, the word errors in the first sentence of each article are not within our means to fix. [1]

	Num. of error	WER
SRI system	1522	25.37 %
MNE	1147	19.12 %
Possible Improvement	375	6.25 %

Figure 2: Word Error of the base system and MNE

6 Result

The absolute improvement using the sublanguage component over SRI's system is 0.65%, from 25.37% to 24.72%, as shown in Table 3. That is, the number of word errors is reduced from 1522 to 1483. This means that 10.40% of the possible improvement was achieved (39 out of 375). The

System	WER	Num. of Error	Improve excl. MNE
SRI	25.37 %	1522	
SRI+SL	24.72 %	1483	10.40 %

Figure 3: Word Error Rate

absolute improvement looks tiny, however, the relative improvement excluding MNE, 10.40 %, is quite impressive, because there are several types of error which can not be corrected by the sublanguage model, as was explained before.

The following is an example of the actual output of the system. (This is a relatively badly recognized example.)

```
==== Example ====
in recent weeks hyundai corporation and
fujitsu limited announced plans for
memory chip plants in oregon at projected
costs of over one billion dollars each

in recent weeks CONTINENTAL VERSION
SUGGESTS ONLY limited announced plans for
MEMBERSHIP FINANCING FOR IT HAD projected
COST of one DAY each

in recent weeks CONTINENTAL VERSION
SUGGESTS ONLY limited announced plans for
memory chip plants in WORTHINGTON PROJECT
COST of one MILLION each
```

[1] Note that, in our experiment, a few errors in initial sentences were corrected, because of the weight optimization based on the eight scores which includes all of the SRI's scores. But it is very minor and these improvements are offset by a similar number of dis-improvements caused by the same reason.

The first sentence is the correct transcription, the second one is SRI's best scored hypothesis, and the third one is the hypothesis with the highest combined score of SRI and our models. This sentence is the 15th in an article on memory chip production. As you can see, a mistake in SRI's hypothesis, `membership` instead of `memory` and `chip`, was replaced by the correct words. However, other parts of the sentence, like `hyundai corporation and fujitsu`, were not amended. We found that this particular error is one of the MNE, for which there is no correct candidate in the N-best hypotheses. Another error, `million` or `day` instead of `billion`, is not a MNE. There exist some hypotheses which have `billion` at the right spot, (the 47th candidate is the top candidate which has the word). Our sublanguage model works to replace word `day` by `million`, but this was not the correct word.

7 Discussion

Although the actual improvement in word error rate is relatively small, partially because of factors we could not control, of which the problem of MNE is the most important, the results suggest that the sublanguage technique may be useful in improving the speech recognition system. One of the methods for increasing the possibility of improvement is to make N (of N-best) larger, thus including more correct hypotheses in the N-best. We tried this, because SRI actually provided us with 2000 N-best hypotheses. However, parameter optimization showed us that 100 is the optimal number for this parameter. This result can be explained by the following statistic. Table 4 describes the number of MNE as a function of N for the training data set and evaluation data set. Also in parentheses, the number of possible improvements for each case is shown. According to

N	MNE (evaluation)	MNE (training)
1	1522	1258
50	1163 (359)	991 (267)
100	1147 (375)	960 (298)
200	1134 (388)	947 (311)
500	1116 (406)	935 (323)
1000	1109 (413)	930 (328)
2000	1107 (415)	929 (329)

Figure 4: N and Word Error

the table, the number of MNE decreases rapidly for N up to 100; however, after that point, the number decreases only slightly. For example, in the evaluation data set, increasing N from 500 to 2000 introduces only 9 new possible word error improvements. We believe this small number gives our component greater opportunity to include errors rather than improvements.

Improvements will no doubt be possible through better adjustment of the parameter settings. There are parameters involved in the similarity calculation, the size of the sublanguage set, the ratio threshold, etc. To date, we have tuned them by manual optimization using a relatively small number of trials and a very small training set (the 20 articles for which we have N-best transcriptions). We will need to use automatic optimization methods and a substantially larger training set. Since we do not have a much larger set of articles with speech data, one possibility is to optimize the system in terms of perplexity using a much larger text corpus for training, and apply the optimized parameters to the speech recognition system. With regard to the size of sublanguage set, a constant size may not be optimal. Sekine (Sekine, 1994) reported on an experiment which selects the size automatically by seeking the minimum ratio of the document set perplexity to the estimated perplexity of randomly selected document sets of that size. This approach can be applicable to our system.

We may also need to reconsider the strategy for incorporating the sublanguage component into the speech recognition system. For example, it might be worthwhile to reconsider how to mix our score with SRI's language model score. SRI provides language model scores for each hypothesis, not for words. However, we can imagine that, if their language score can be computed with high confidence for a particular word, then our model should have relatively little weight. On the other hand, if the language model has low confidence, sublanguage should have strong weight. In other words, the combination of the scores should not be done by linear combination at the sentence level, but should be done at the word level.

Also there are several things we need to re-evaluate regarding our sublanguage model. One of them is the threshold method we adopt here, which introduces undesirable discontinuities into our language model. The method for retrieving similar articles may also need to be modified. We used a simple technique which is common in information retrieval research. However, the purpose of our system is slightly different from that of information retrieval systems. So, one future direction is to look for a more suitable retrieval method for our purpose.

In closing, we wish to mention that the sublanguage technique we have described is a general approach to enhancing a statistical language model, and is therefore applicable to tasks besides speech recognition, such as optical character recognition and machine translation. For example, if a machine translation system uses a statistical model for target language word choice, our

approach could improve word choice by selecting more topic related words.

8 Acknowledgment

The work reported here was supported by the Advanced Research Projects Agency under contract DABT63-93-C-0058 from the Department of the Army. We would like to thank the collaboration partners at SRI, in particular Mr.Ananth Sankar and Mr.Victor S. Abrash. Also we thank for useful discussions and suggestions Prof. Grishman and Slava Katz.

References

Satoshi Sekine, John Sterling and Ralph Grishman 1995 NYU/BBN 1994 CSR evaluation In *Proceedings of the ARPA Spoken Language Systems Technology Workshop*

R Kuhn. 1988 Speech Recognition and the Frequency of Recently Used Words: A Modified Markov Model for Natural Language In *Proceedings of 12th International Conference on Computational Linguistic*

F Jelinek, B Merialdo, S Roukos, and M Strauss. 1991 A Dynamic Language Model for Speech Recognition In *Proceedings of Speech and Natural Language DARPA Workshop*

J Kupiec. 1989 Probabilistic Models of Short and Long Distance Word Dependencies in Running Text In *Proceedings of Speech and Natural Language DARPA Workshop*

Ronald Rosenfeld and Xuedong Huang. 1992 Improvements in Stochastic Language Modeling In *Proceedings of DARPA Speech and Natural Language Workshop*

Satoshi Sekine 1994 A New Direction for Sublanguage NLP In *Proceedings of International conference on New Methods in Language Processing*

K Sparck-Jones. 1973 Index Term Weighting In *Information Storage and Retrieval, Vol.9, p619-633*

Vassilios Digalakis, Mitch Weintraub, Ananth Sankar, Horacio Franco, Leonardo Neumeyer, and Hy Murveit 1995 Continuous Speech Dictation on ARPA's North Business News Domain In *Proceedings of the ARPA Spoken Language Systems Technology Workshop, p88-93*

David S. Pallett et.al. to appear 1995 Benchmark Test for the ARPA Spoken Language Program In *Proceedings of the ARPA Spoken Language Systems Technology Workshop*