A BOOTSTRAP TECHNIQUE FOR BUILDING DOMAIN-DEPENDENT LANGUAGE MODELS

Ganesh N. Ramaswamy *Harry Printz* *Ponani S. Gopalakrishnan*

IBM Thomas J. Watson Research Center
Yorktown Heights, New York

ABSTRACT

In this paper, we propose a new bootstrap technique to build domain-dependent language models. We assume that a seed corpus consisting of a small amount of data relevant to the new domain is available, which is used to build a reference language model. We also assume the availability of an external corpus, consisting of a large amount of data from various sources, which need not be directly relevant to the domain of interest. We use the reference language model and a suitable metric, such as the perplexity measure, to select sentences from the external corpus that are relevant to the domain. Once we have a sufficient number of new sentences, we can rebuild the reference language model. We then continue to select additional sentences from the external corpus, and this process continues to iterate until some satisfactory termination point is achieved. We also describe several methods to further enhance the bootstrap technique, such as combining it with mixture modeling and class-based modeling. The performance of the proposed approach was evaluated through a set of experiments, and the results are discussed. Analysis of the convergence properties of the approach and the conditions that need to be satisfied by the external corpus and the seed corpus are highlighted, but detailed work on these issues is deferred for the future.

1. INTRODUCTION

Several recent experiments have shown that domain-dependent language models offer the best hope for reducing the word error rate for domain-restricted speech recognition tasks. But building a language model usually requires a large amount of training data, and since large, domain-restricted corpora are not easy to find, it can be difficult to apply this useful result. Solutions to this problem proposed in the literature use some form of class-based modeling [2], [4], [5], [8], or some form of mixture modeling and task adaptation [3], [6], [7], [10].

In this paper, we propose a new approach to solve the problem, by using a bootstrap technique to iteratively build a series of domain-dependent language models. The main ingredient of the approach is to iteratively select sentences from a collection of out-of-domain data, by using an initial language model built from a small amount of data relevant to the domain. The domain-dependent language model is updated at the end of each iteration, and the updated model is used to select additional sentences for the subsequent iterations. In the experiments conducted to evaluate the approach, we successfully added a significant amount of new data to the initial collection, and we were able to reduce the language model perplexity and speech recognition error rates. The results of the experiments are discussed in the paper, along with other enhancements, such as combining the approach with class-based modeling and mixture modeling, which can further strengthen the language model building process.

The remainder of the paper is divided into four sections. Section 2 describes the proposed bootstrap technique in greater detail, along with variations to the technique. Experiments and results are discussed in Section 3. Factors that need to be considered in successfully using the proposed approach are discussed in Section 4. The final section concludes the paper with a summary.

2. THE BOOTSTRAP TECHNIQUE

In this section, we describe the bootstrap technique for building domain dependent language models in greater detail. The first step is to collect a small number of sentences that are highly relevant to the domain. If necessary, these sentences may be generated by hand. This body of data will be called the *seed corpus*. The minimum number of sentences that the seed corpus should contain will depend on the task, and is an important parameter that will determine the quality of the final model. We will revisit this issue in Section 4.

The seed corpus is used to build a small initial language model, called the *reference model*. The proposed technique is independent of which type of language model is used. In the experiments described in Section 3, we used the popular trigram language models, along with the class-based and mixture modeling enhancements [2], [7], [9]. A portion of the seed corpus is reserved for validation purposes.

The next step is to construct the *external corpus*, by collecting non-specific data from various internet sites and other data sources (including language modeling data from other domains). We discuss the properties that the external corpus should satisfy in Section 4.

A suitable metric has to be defined that can be used to evaluate the relevance of the candidate sentences from the external corpus to the domain. Our technique can be used in conjuction with any reasonable measure of the relevance of a candidate sentence to a domain-specific language model. In the experiments described in Section 3, we use the *perplexity* measure [9].

Once we have chosen a suitable metric, we can start evaluating the candidate sentences from the external corpus. In the simplest version of the approach, a sentence is selected for inclusion if the perplexity is below a threshold, and discarded otherwise. Variations to this step will be discussed later. The threshold for inclusion is an important parameter of the technique, and we will revisit this issue in Section 4.

Once we have a sufficient number of new sentences, we can rebuild the reference language model, and recalculate the threshold for inclusion. The number of new sentences to be added at each iteration is another important parameter of the technique.

It may be desirable to limit the number of new sentences at each iteration to be no more than a certain percentage of the number of sentences already in the (updated) seed corpus. Hence, we may scan the entire external corpus and select the predetermined number of top ranking sentences, provided that they also satisfy the threshold criteria. However, this may be an computationally expensive procedure, and the alternate method of evaluating the sentences from the external corpus in sequence and terminating the current iteration when the predetermined number of sentence satisfying the threshold (but not necessarily top ranking) may be used when the size of the external corpus is very large.

At each iteration, the quality of the updated language model is evaluated using the independent test data. Both perplexity and recognition accuracy should be measured to make an accurate evaluation. The iterations continue until a satisfactory termination point is achieved, or when the performance of the language model starts to deteriorate.

2.1. Enhancements to the Bootstrap Technique

The proposed method of iteratively constructing language models can be further enhanced using a mixture of two or more language models. One model is built using the sentences that are most relevant to the domain, according to the reference model; other models are built using less relevant sentences. Each individual model will have a different threshold for sentence inclusion. In this case, a candidate sentence can be added to the most relevant language model if it has a high enough score, or to one of the less relevant models if the score is low, or it can be discarded altogether. The resulting spectrum of models is mixed for robustness.

Another technique that can be used in conjunction with the proposed approach is class-based language modeling. If the reference language model is built using classes, we can use an automatic classer to identify the classes present in a candidate sentence, and then compute the perplexity of the sentence on the class-based reference language model. For example, if we have classes for names, and the candidate sentence contains a name that does not appear in the reference language model, then the candidate sentence can still be selected.

3. PERFORMANCE EVALUATION

In this section we describe some of the experiments that were done to evaluate the proposed approach. The experiments were done within the context of a task involving a spoken natural language user interface to an email application. This is a highly specialized task and the general data sources typically used to build language models were not very useful for for this domain. For a task like this, it would normally be necessary to carry-out rather labor-intensive "Wizard-of-Oz" type of data collection process. Hence, this task was a particularly suitable test of our approach.

We constructed the seed corpus for this task using about 500 sentences, which were generated by hand and through a limited data collection process. These sentences contained on the average of about 6 words per sentence, for a total of 3000 words in the seed corpus. An additional set of 300 sentences (acquired in the same fashion), was put aside for testing.

The external corpus constructed for the experiments consisted of about 150,000 sentences drawn from various different sources, including the popular Broadcast News and Switchboard corpora, as well as some IBM internal data sources. The size of the external corpus was intentionally smaller than what would be typically used for building language models, to make experiments manageable.

Table 1: **Iterative Language Model Building.** The table below contains statistics corresponding to the language models built at each iteration. The first column (iteration 0) corresponds to the initial reference model built using just the seed corpus.

Iteration Number	0	1	2	3
Total Sentences	500	806	1469	1736
Perplexity on Seed Corpus	28	24	22	23
Minimum Score	4	4	3	3
Maximum Score	6147	4975	7829	7028
Mean Score	93	66	60	54
Median Score	20	16	17	19
Standard Deviation	386	319	362	278
80th Percentile Score	59	37	36	34
90th Percentile Score	150	66	60	57
95th Percentile Score	311	209	108	125
98th Percentile Score	636	449	398	420

The initial reference language model was built using the seed corpus. The language model was a standard trigram model, along with bigram and unigram models used for smoothing [9]. The models were also class-based [2], with simple classes such as names, numbers, months, etc. The vocabulary used for building the language model consisted of the top 5000 words in English, along with all the words in the seed corpus and the test set (to avoid out-of-vocabulary errors in recognition experiments). The statistics corresponding to the reference model are given in the column corresponding to iteration 0 in Table 1. The perplexity of the reference model computed on the seed corpus (which was used to build the reference model) was 28. The individual sentence based perplexity scores ranged from a minimum of 4 to a maximum of 6147, with a mean of 93 and a median of 20. Other statistics can be found in Table 1.

As noted in the previous section, we used the sentence-based perplexity score as the metric for measuring the relevance of a candidate sentence to a new domain. Taking a conservative approach, we decided to set the threshold for inclusion as the 80th percentile score from the previous iteration. Therefore, in selecting sentences for the first iteration, we used a threshold of 59, which is the 80th percentile score of the seed corpus (iteration 0). During the first iteration, we used this procedure to add 306 additional sentences from the external corpus and rebuilt the language model. The statistics for the language model built at iteration 1 are shown in Table 1. Using the 80th percentile score of 37 from iteration 1 as the threshold, we added another 663 sentences during iteration 2, and rebuilt the model. Similarly, during the third (and final) iteration, we added another 267 sentences, for a total of 1736 sentences. The decision to end the iterations at this point was somewhat arbitrary, although the declining number of sentences added seems to suggest that we probably have extracted most of the relevant sentences from this small external corpus.

Table 2 shows the results of the first set of experiments that were conducted to establish a baseline, on the test set described

Table 2: **Baseline Experiments.** The table below shows the perplexities and recognition error rates for several baseline experiments. Both the perplexity computations and the recognition tests were done on an independent test set.

Model	Perplexity	Error Rate
Seed Corpus Only	183	24.7%
External Corpus Only	1554	38.9%
Seed+External (BF)	509	25.6%
Seed+External (3:1 Mix)	188	24.2%

Table 3: **Bootstrap Experiments.** The table below shows the perplexities and recognition error rates for experiments corresponding to several iterations of the bootstrap technique. The perplexity computations and the recognition tests were done on the same test set as in Table 2.

Iteration	New Sentences	Perplexity	Error Rate
1	306	164	23.9%
2	663	187	22.5%
3	267	177	21.9%
Mixture	-	149	19.2%

Table 4: **Examples of Sentences.** The table below shows examples of sentences from the seed corpus, and from the *most relevant*, *less relevant* and *not relevant* groups corresponding to the "mixture" experiment of Table 3.

Seed Corpus	show me the next email open the inbox delete this note scroll down reply to this message
Most Relevant	put the most recent at the bottom can you alphabetize by subject what's the full name of this person trash that continue to the next page
Less Relevant	I'd like to see the planet begin with a lower case letter open the proposal testing one two three what's new
Not Relevant	but why is his head missing that was a big excitement could her toothpaste have been the culprit I hope to bridge the gap she weighed less than two pounds

earlier. We used a speaker independent continuous speech recognition system, with the vocabulary and all the parameters of the recognition engine fixed for all the experiments (although the system used here is similar to the system described in [1], we used a version that was tuned for speed and not for performance). The detailed description of the recognition engine is omitted here since it is not relevant to the specific focus of evaluating the proposed technique.

With the initial reference language model, the error rate was 24.7% and the perplexity (also computed using the same held-out test set) was 183. We also built a language model using just the external corpus, and this model resulted in a much higher perplexity and error rate, as shown in Table 2. When we pooled together all the data from the seed and external corpora, in a brute force (BF) manner, and built a single language model, we obtained more moderate results, as shown in the row labeled "Seed+External (BF)". We then conducted a series of experiments where we mixed the two language models (one built using just the seed corpus and the other built using just the external corpus) at the probability level, with varying mixture weights. The lowest recognition error rate was obtained when the models were mixed at 3:1 ratio, and the results for this case are also shown in Table 2. Since the external corpus does provide some smoothing, it was not surprising that the results for the mixture case were better than when we used the seed corpus alone.

The results from the main set of experiments to evaluate the bootstrap technique are shown in Table 3. For the first experiment, we used the language model from iteration 1 of Table 1, and the error rate now dropped to 23.9 % and the perplexity dropped to 164, both of which were better than the baseline results of Table 2. For iterations 2 and 3, the error rate continued to drop, although the perplexity measures increased slightly compared to iteration 1 (while still remaining below those from the baseline experiments of Table 2). The best results, both from perplexity and error rate points of view, were obtained when we used the mixture modeling enhancement to the bootstrap technique, described earlier in Section 2.1. We took all the sentences from iteration 3 (including the seed corpus), and partitioned the set into two groups, on the basis of the perplexity scores. The "most relevant" group consisted of 725 sentences, and the "less relevant" group consisted of 1011 sentences, and we constructed two language models. We also collected all the sentences from the external corpus that were not selected in iteration 3, and constructed another language model. We mixed these three language models (at the probability level) using a mixture ratio of 6:3:1, with the higher weight corresponding to the most relevant model. With this enhancement, the error rate dropped to 19.2%, which is a 22% relative reduction in error rate compared to the 24.7% error rate from the baseline experiment. Examples of sentences for the different groups are shown in Table 4.

4. DISCUSSION

In this section, we discuss several issues that need to be considered to effectively use our technique. The most important of these is to understand the conditions under which the iterative scheme will converge to a language model that is better than the initial reference model. A rigorous analysis of the conditions for convergence is beyond the scope of this paper, but we highlight the issues that would affect the convergence.

The size and coverage of the seed corpus is perhaps the most important consideration. If the size and coverage of the seed corpus are sufficiently high, the resulting language models would not only perform well, it would also be possible to add more sentences at each iteration, thus reducing the number of iterations required. A larger seed corpus would also provide

robustness against adding non-relevant sentences, particularly when the external corpus contains a significant amount of non-relevant sentences.

The lack of coverage in the seed corpus may be partially compensated by the external corpus if the external corpus contains a significant amount of highly relevant data. If this is the case, then the selected sentences from the external corpus would extend the coverage in subsequent iterations. Such an extension of coverage would provide robustness to the scheme, but more importantly, the quality of the resulting language models themselves depend very heavily on the external corpus. A good seed corpus can ensure that we do not add non-relevant sentences that corrupt the language models, but to improve the performance of the models with each iteration, it is necessary that the external corpus contains enough relevant sentences.

Another parameter that controls the convergence of the scheme is the number of sentences that are added in each iteration. In Table 3, one possible explanation for the degradation in the perplexity measure for iteration 2 may be related to the large number of sentences added at this iteration (we were following a blind rule of adding all sentences satisfying the threshold). A more conservative approach of adding only a few sentences at each iteration would help to ensure robustness, at the expense of increased computation. If the size and coverage of the seed corpus appear to be insufficient, and if the external corpus is unable to compensate for the insufficiency, it may be worthwhile to limit the number of new sentences to be added at each iteration to a conservative value.

Determining the termination point for the iterative scheme is usually straightforward. The logical termination point is when there is evidence that most of the relevant sentences from the external corpus have been selected. We can make that decision by looking at the number of new sentences that qualify at each iteration, and a significant drop in this number may flag termination. Alternatively, we can simply decide based on the performance of the resulting language models.

We noted in Section 2.1 that the bootstrap technique can be enhanced by using class-based modeling. Although we used some simple classes in the experiments of Section 3, we did not make full use of class-based modeling. For example, sentences such as "show me the next email", are common in the email domain that we used in the experiments of Section 3. Then, sentences such as "show me the next flight", which can be found in the air travel domain, can be added to the email domain language model, if we were to put "email" and "flight" in the same class. This would be a non-trivial use of class-based modeling.

The mixture modeling enhancement suggested in Section 2.1 was used successfully in the final experiment in Section 3. However, instead of clustering the selected sentences in the end, one could establish several "buckets" at the beginning of the iterative process, each with its own threshold for sentence inclusion, and the selected sentences could be directly placed in the appropriate bucket.

In the experiments of Section 3, we used the perplexity scores to measure the degree of relevance of a candidate sentence to the domain. The choice to use the perplexity measure was somewhat arbitrary and it would be useful to experiment with other reasonable measures. Another interesting experiment would be to evaluate new data at the paragraph level, or using some other reasonable unit, instead of the sentence level evaluation used in Section 3.

5. CONCLUSION

In this paper, we presented a new bootstrap technique to iteratively build domain-dependent language models from a limited amount of training data. We illustrated the performance of the proposed technique through a set of experiments. Various issues related to the convergence and effectiveness of the proposed approach were also discussed. Although the proposed technique may not be substitute for extensive in-domain data collection, it appears to be a very effective way to use external data sources to build domain-dependent language models.

REFERENCES

[1] Bahl, L. R., et. al., "Performance of the IBM Large Vocabulary Continuous Speech Recognition System on the ARPA Wall Street Journal Task," IEEE International Conference on Acoustics Speech and Signal Processing, Detroit, pp. 41-44, May 1995.

[2] Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai, J. C., and Mercer, R. L., "Class-Based N-Gram Models of Natural Language," *Computational Linguistics*, Vol. 18, No. 4, pp.467-479, 1992.

[3] Crespo, C., Tapias, D., Escalada, G., and Alvarez, J., "Language Model Adaptation for Conversational Speech Recognition using Automatically Tagged Pseudo-Morphological Classes," *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 823-826, Munich, April 1997.

[4] Frahat, A., Isabelle, J.-F., O'Shaugnessy, D., "Clustering Words for Statistical Language Models Based on Contextual Word Similarity," *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 180-183, Atlanta, May 1996.

[5] Issar, S., "Estimation of Language Models for New Spoken Language Applications," *International Conference on Spoken Language Processing*, Vol. 2, pp. 869-872, Philadelphia, October 1996.

[6] Iyer, R., Ostendorf, M., and Gish, H., "Using Out-of-Domain Data to Improve In-Domain Language Models," IEEE Signal Processing Letters, Vol. 4, No. 8, pp. 221-223, August 1997. Proceedings of the IEEE ICASSP, Detroit, pp. 41-44, May 1995.

[7] Jelinek, F., and Mercer, R. L., "Interpolated Estimation of Markov Source Parameters from Sparse Data," *Workshop on Pattern Recognition in Practice*, pp. 381-397, Amsterdam, 1980.

[8] Jelinek, F., "Self-Organized Language Modeling for Speech Recognition," *Readings in Speech Recognition*, A. Waibel and K. F. Lee (ed.), Morgan Kaufman Publishers, 1990.

[9] Jelinek, F., *Statistical Methods for Speech Recognition*, The MIT Press, 1997.

[10] Masataki, H., Sagisaka, Y., Hisaki, K., and Kawahart, T., "Task Adaptation Using MAP Estimation in N-Gram Language Modeling," *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vol. 2, pp. 783-786, Munich, April 1997.