Effective topic-tree based language model adaptation

Javier Dieguez-Tirado, Carmen Garcia-Mateo and Antonio Cardenal-Lopez

Dpto. Teoría de la Señal y Comunicaciones
ETSI Telecomunicación – University of Vigo
VIGO (SPAIN)
jdieguez,carmen,cardenal@gts.tsc.uvigo.es

Abstract

We work on adaptation schemes for language modeling well suited for limited resources scenarios. In order to take advantage of available out-of-domain corpora, language model adaptation using topic mixtures was investigated. This technique has not given good practical results in the past. In this paper, we have performed several modifications to an existing tree-based approach. The tree was obtained from the background corpus by means of partitional clustering. All the nodes were exploited in the adapted model, and non-erroneous in-domain transcriptions were used as the adaptation corpus. The modified technique yielded a 14% perplexity improvement in a bilingual BN task, outperforming several non-hierarchical approaches. A strategy for an early application of the language model allowed to translate this perplexity improvement into a 4% WER reduction.

1. Introduction

N-gram models have proven to be a very successful statistical language modeling scheme for use in practical speech recognition systems. Since its adoption in the early 80s, most of the attempts to improve upon the performance of a standard trigram language model (LM) have resulted in small gains, at the expense of a considerably higher implementation cost. However, the main limitations of n-gram models are clear: (i) n-grams are trained on a fixed set of data, but most of the times they are applied in a changing environment, and (ii) n-gram models are unable to capture long range dependencies between words, often referred to as the *topic* of the conversation. Language model adaptation [1] addresses these limitations, and has been accepted as a promising way of improvement.

The use of *topic mixtures* has been a popular method for language model adaptation in the last decade [2, 3, 4]. This technique relies on splitting a background text corpus into a set of topic homogeneous subsets, building a language model for each topic, and calculating the weighted linear combination that optimizes the perplexity over a particular piece of text (or recognition hypothesis). While this method seems to be able to keep track of long range dependencies, and to react to changes in the application domain, it has not provided particularly good results: perplexity improvements have been small, and have rarely translated into word error rate (WER) reductions. One of the major limitations of these methods, which may be responsible for their disappointing performance, is that the component language models become undertrained as the topic granularity is increased [1]. Some proposed solutions are to add a general language model component, constructed from the whole text corpus; or to use topic trees [5, 6]. Topic trees are sets of hierarchically organized language models, ranging from specific but sparse to general but well-trained which can be combined to obtain the final LM. The advantage of this approach is that the compromise between specificity and sparseness may be easily optimized, but there are several implementation-related issues, (topic clustering, selection of training data, etc), which have limited its effectiveness.

To overcome these problems, several modifications of the basic algorithm presented in [5] are proposed in this paper. As described in Section 2, these modifications comprise the use of partitional clustering to obtain the tree, the employment of the whole tree for building the adapted model and the application of the computed LM in the recognition first pass instead of in the N-Best stage. With all these improvements a 14% perplexity reduction, and a 4% WER improvement was achieved over the baseline method in the context of a bilingual broadcast news transcription system. A detailed description of the experiments is presented in Section 3.

2. LM adaptation using a topic tree

The method presented in [5] has been modified, in order to avoid potential sources of problems which may have been affecting its performance:

- The topic tree has been constructed using unsupervised automatic clustering rather than based on manual keywords, as it has been shown that automatic clustering may lead to lower perplexities, e.g. [3].
- Instead of selecting just a subset of the tree to construct the interpolated language model, we have included the whole tree. This avoided imperfections derived from the selection process. The weights were trained using pre-collected in-domain data rather than first-pass transcriptions, avoiding the use of erroneous material.
- The perplexity improvements would be more easily translated into WER reductions if an early application of the adapted LM was performed. Thus, the adapted LM was integrated into the search pass of the recognizer, instead of waiting until an N-best phase.

2.1. Construction of a topic tree

A topic tree has been constructed from a background corpus using automatic document clustering. The attention was put into two main aspects: (i) definition of a document distance measure and (ii) clustering algorithm.

Some typical distance measures used for document clustering include unigram perplexity [4], and cosine similarity using a tf-idf vectorial representation [3]. We have chosen tf-idf for its computational advantages, and for preserving a consistent quality across the clustering process. With the purpose of retaining

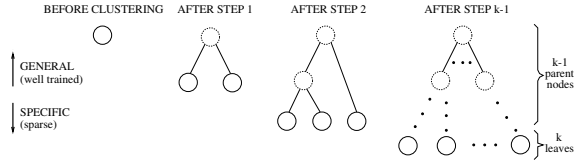

Figure 1: *Partitional clustering with the $k-1$ bisections method*

some stylistic content in our vectorial representation, we decided against the use of stemming and stopping. The use of a different feature for each word has meant we have had to cope with a vector dimensionality equaling the vocabulary size.

K-means [4] and agglomerative clustering [2, 6] were considered as potential clustering algorithms. Agglomerative clustering seems to be an adequate approach if a topic tree is pursued, and has been used as such by [6]. However, recent research seems to indicate that partitional clustering may have computational and performance advantages over agglomerative clustering [7]. The cited work suggests the use of the $k-1$ bisections method with I_2 criterion function, so we have employed it in this paper.

Using this method, the construction of a topic tree is straightforward. The collection of M document vectors using a vocabulary of N terms is arranged into a matrix of MxN elements, and $\text{sim}(v,u)$ is defined as the cosine similarity of vectors v and u. First, the matrix is split into two groups S_1 and S_2 so that the 2-way clustering solution ($m=2$) optimizes the I_2 criterion function:

$$max \sum_{i=1}^{m} \sqrt{\sum_{v,u \in S_i} \text{sim}(v,u)} \quad (1)$$

At each further step, the cluster that would optimize the overall criterion function is selected and is further bisected. After $k-1$ steps, k clusters are obtained. The tree is formed by using the final k clusters as leaves and the intermediate $k-1$ bisected clusters as parent nodes, giving a total of $2 \cdot k - 1$ nodes. This process is illustrated in Figure 1.

2.2. Language model adaptation using EM

After a topic tree was obtained from the background text corpus, component trigram models were trained for each node. No vocabulary or n-gram pruning was performed at this stage. Finally, an adapted language model was obtained by combining a set I of component language models:

$$P(w|h) = \sum_{i \in I} \lambda_i P_i(w|h) \quad I \subset \{1, \cdots, 2k-1\} \quad (2)$$

where the probability P is given for a word w with history h. The interpolation weights $\{\lambda_i\}$ were trained by minimizing the perplexity on a given adaptation corpus, using the EM algorithm. Two main decisions were adopted: (i) no restrictions were put on the number of components of the language model mixture and (ii) the adaptation corpus was taken from non-erroneous a-priori transcriptions from the target domain.

By including all language model components into the mixture, there is no need for an a-priori selection of the most relevant components. Thus, the deployment of an imperfect classifier, as done in [5], is avoided. There were also no restrictions placed on the number of nodes included per branch. The purpose is to let the EM algorithm identify for itself the best components to minimize the target perplexity. However, this increase in the number of parameters to be trained also imposes that the adaptation corpus must be sufficiently large. For this reason, the usual approach of employing the first-pass decoder transcriptions for *dynamic* adaptation cannot be applied. A possible solution to alleviate this problem was explored in [8]. In this paper, for simplicity reasons, we have restricted ourselves to *static* adaptation, by training a single universal domain-adapted language model.

In order to provide a large enough amount of adaptation data, most of the available task-related in-domain data was used (training set). Overfitting effects were catered for by using a small amount of held-out in-domain text (validation set), and optimizing the number of clusters k on this set instead of on the training set. The use of a large adaptation corpus entails some implementation difficulties, as the CPU and memory requirements of the EM algorithm are proportional to both the number of clusters and the number of samples. Convergence time was reduced in the following way: the set of N samples was partitioned into 2^K blocks of $N/2^K$ samples each. Optimization was performed in K steps, using the first 2^i blocks of samples in each step. Each step was initialized using the weights resulting from the previous step. This resulted in faster steps requiring a smaller number of iterations. If an additional speed or storage gain is needed, the process may be aborted before step K if the perplexity is observed to vary less than a certain threshold.

2.3. Applying the language model in an early stage

One of the main drawbacks of most language model adaptation approaches is the deferral of the application of the full adapted model until an N-best rescoring pass. The search pass is usually performed using a general unadapted, lightweight language model. This often causes some of the better paths to be lost, impeding their recovery via best path rescoring. It also limits the words that can be recognized to those appearing in the original unadapted lexicon. We believe this is one of the main reasons why some topic-mixtures methods have been unable to translate perplexity improvements into a lower WER (eg. [4]).

Thus, a strategy that is able to apply the adapted language model directly in the first search pass was developed. For this purpose, we have taken advantage of a useful property of interpolated language models, that allows them to be converted to a single conventional n-gram model. After static merging, the definitive adapted lexicon was obtained by selecting the desired number of words with highest unigram probabilities. The model was pruned using entropy-based pruning, to limit the model it to a reasonable size. This pruned, standalone trigram model was applied efficiently in the search pass, by means of a state-of-the-art lookahead mechanism developed for our recognizer. This mechanism is based in a three layer architecture that exploits redundancy in the lookahead calculations [9].

3. Experimental results

3.1. Experimental Framework

The proposed language model adaptation strategy was tested within the Transcrigal framework, a bilingual broadcast news (BN) transcription system for the Spanish (ES) and Galician (GA) languages [8]. The Transcrigal-DB in-domain database has recently been updated to 31 news shows. Each show has an approximate length of 60 minutes and is of a bilingual nature: 89% corresponds to planned and spontaneous GA speaker turns,

ID subset	# shows	# words	ppl	% oov
train	26	267758	258.9	0.34
validation	2	20216	238.2	0.23
test	3	31573	268.2	0.36

Table 1: *Baseline perplexities (in-domain corpus)*

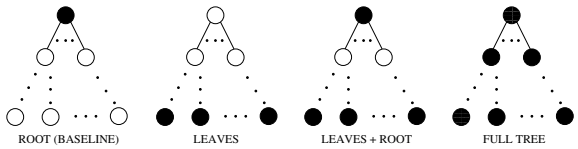

Figure 2: *Partial and full trees*

while the remaining 11% consists of ES spontaneous utterances. The 31 shows were partitioned into training, validation and test data as 26, 2 and 3 shows, respectively.

The background corpus used for building the topic tree was extracted from the bilingual journal "El Correo Gallego", collected between December 2000 and January 2005. This corpus contains 105M words grouped into 303K documents, and 639K word types. Distribution among ES and GA is approximately 60% and 40% respectively. Baseline perplexities are measured against the general, unpruned background LM with the full vocabulary and are presented in Table 1.

Clustering was performed with the aid of the CLUTO toolkit [7], and component language models were trained and merged using the SRILM toolkit [10], with Katz smoothing. For recognition experiments, LMs were pruned with a threshold $2,5 \cdot 10^{-8}$ and the vocabulary size was fixed at 20K words. Pruning parameters of the ASR system were tuned for 3xRT execution in current Pentium 2,40–3,06 GHz servers. Acoustic models were adapted to male, female and anchor speakers using the procedure described in [8].

3.2. Overall performance, overfitting and fragmentation

A set of trees with different topic granularities was obtained from the background corpus by letting k vary between 1 and 512. We were unable to go beyond 512 because of memory constraints in the clustering process. The use of a full tree was compared to other typical topic mixtures approaches by selecting fours sets of indices I (Eq. 2) for each tree (Fig. 2).

Training and validation set perplexities are depicted in Figure 3 for all these schemes, using the 26 BN training shows as adaptation data. The number of clusters k that minimizes the perplexity of each set is marked with a square. It can be seen that the k that optimizes the validation text is always lower than the best k for the training set, due to overtraining effects.

The "leaves" scheme is seen to be severely affected by fragmentation, as its performance clearly falls below the baseline level when $k > 16$. Smoothing with the general model ("leaves+root") helps to overcome this problem, but there is also a critical point where performance begins to deteriorate. The full tree is observed to follow closely the performance of the previous scheme up to this critical point, but is able to deliver further improvements as the number of topics is increased. These results suggest that higher perplexity reductions might be achieved by increasing the number of clusters.

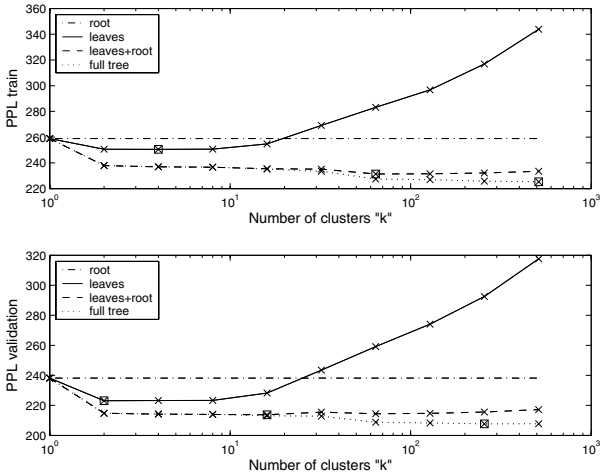

Figure 3: *Overfitting and fragmentation effects*

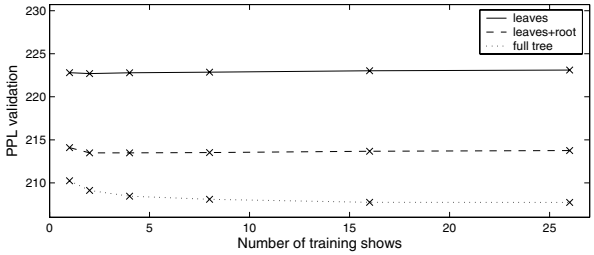

Figure 4: *Effect of adaptation corpus size*

3.3. Adaptation corpus size

The presented technique for training the interpolation weights (Sec. 2.2) involves incorporating the adaptation corpus into the optimization process progressively. We also used this method to monitor the effect of the adaptation corpus size on the perplexity improvements. The adaptation corpus was made to grow exponentially, from 1 to 26 shows taken from the in-domain training set. For each corpus size, the number of clusters was optimized by using the validation set. The resulting perplexities are presented in Figure 4. It can be seen that full tree method works best for all adaptation sizes, and shows higher improvements when the adaptation corpus is extended.

3.4. Most significant nodes and active branches

The introduction of every single node of the tree in the EM algorithm optimization process, allows to determine the most relevant components according to the resulting weights. An experiment was performed to determine if the most important nodes belong to different branches, or are concentrated in a few, distinct branches. For this purpose, the full tree with $k = 256$ was taken, and its 511 components were sorted in decreasing weight order. The upper half of Figure 5 shows the probability mass of the b best components: $\sum_{i=1}^{b} \lambda_i$. The lower half shows the number of branches exceeding a certain number of active nodes, when only the b best nodes are selected. The results indicate the existence of some branches with a considerable number of active nodes, even for small values of b. Thus, the decision not to place a restriction on the maximum number of nodes per branch is justified.

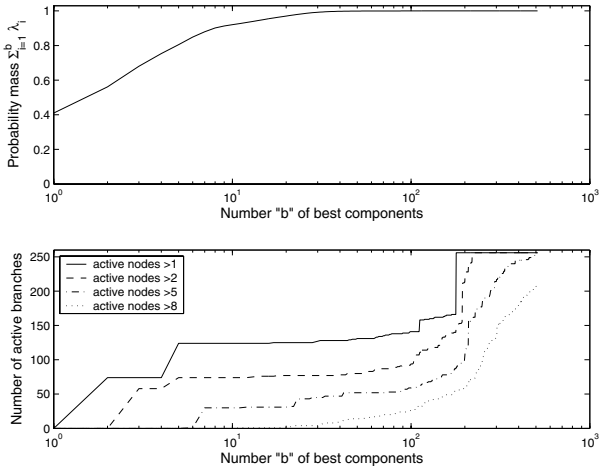

Figure 5: *Post-selection of the best LM components*

method	ppl$_{full}$	% oov$_{20k}$	% WER
root (baseline)	268.2	6.62	40.53
leaves	252.0	6.12	39.59
leaves+root	241.0	6.01	39.37
full tree	**230.4**	**5.82**	**38.76**

Table 2: *Test set results*

3.5. Test set results

Both language model and recognition experiments were performed on the test set. Table 2 presents the results. Perplexity values are given using the full 639K-word vocabulary with a 0.36% OOV rate (Tab. 1), so that the different perplexities are comparable. It can be seen that perplexity improvements obtained on the validation set are also translated to the test set. The OOV rate after pruning the adapted language model to 20K words is also given. This also shows an advantage of the full-tree method in selecting an adapted lexicon. After pruning the language model and restricting the vocabulary, recognition experiments were conducted. The obtained WER rates are well correlated with the perplexity improvements. Note that similar WER values are being obtained in state-of-the-art BN systems with a comparable amount of resources [11].

The proposed adaptation approach may be easily combined with other well-known strategies, allowing further WER reductions. We have performed these improvements: (i) the adapted language model was combined with the in-domain LM reducing the WER to 34.87%; (ii) dynamic adaptation was performed as in [8], further improving the WER to 32.31%. The full tree method also proved itself superior in these 3-way combinations.

4. Conclusions and Further Work

In this paper, several modifications were proposed to an existing topic-tree based language model adaptation approach. A topic tree was constructed from a background corpus using partitional clustering, the full tree was exploited in a mixture model framework, and the interpolation weights were trained using non erroneous transcriptions. Experiments were conducted within a bilingual BN framework, yielding a 14% improvement in perplexity and a 4,3% reduction in word error rate over the baseline performance. The tree-based approach was shown to outperform other non-hierarchical approaches. The early application of the language model in the decoding process showed itself beneficial towards improving the correlation between WER and perplexity. The presented strategy may be easily combined with other techniques to yield further improvements. Our future work involves extending the language model mixtures to include components derived from documents obtained using information retrieval, from the same background corpus.

5. Acknowledgements

This project has been partially supported by Spanish MEC under the project TIC2002-02208, and Xunta de Galicia under the projects PGIDT03PXIC32201PN.

6. References

[1] J. Bellegarda, "Statistical language model adaptation: review and perspectives," *Speech Communication*, vol. 42, no. 1, pp. 93–108, January 2004.

[2] R. M. Iyer and M. Ostendorf, "Modeling long distance dependence in language: Topic mixtures versus dynamic cache models," *IEEE Trans. Speech and Audio Processing*, vol. 7, no. 1, pp. 30–39, January 1999.

[3] Y. Gotoh and S. Renals, "Topic-based mixture language modelling," *J. Natural Language Engineering*, vol. 5, pp. 355–375, 1999.

[4] P. R. Clarkson, "Adaptation of statistical language models for automatic speech recognition," Ph.D. dissertation, University of Cambridge, 1999.

[5] K. Seymore and R. Rosenfeld, "Using story topics for language model adaptation," in *Proc. Eurospeech*, vol. 4, Rhodes, Greece, September 1997, pp. 1987–1990.

[6] R. Florian and D. Yarowsky, "Dynamic nonlocal language modeling via hierarchical topic-based adaptation," in *Proceedings of the 37th Annual Meeting of the ACL*, College Park, Maryland, June 1999, pp. 167–174.

[7] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proc. International Conference on Information and Knowledge Management*, McLean, VA, November 2002, pp. 515–524.

[8] J. Dieguez-Tirado, C. Garcia-Mateo, A. Cardenal-Lopez, and L. Docio-Fernandez, "Adaptation strategies for the acoustic and language models in bilingual speech transcription," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 1, Philadelphia, PA, March 2005, pp. 833–836.

[9] A. Cardenal-Lopez, F. J. Dieguez-Tirado, and C. Garcia-Mateo, "Fast LM look-ahead for large vocabulary continuous speech recognition using perfect hashing," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 1, Orlando, FL, May 2002, pp. 705–708.

[10] A. Stolcke, "SRILM – an extensible language modeling toolkit," in *Proc. Int. Conf. Spoken Language Processing*, vol. 2, Denver, CO, September 2002, pp. 901–904.

[11] L. Lamel, J.-L. Gauvain, G. Adda, M. Adda-Decker, L. Canseco, L. Chen, O. Galibert, A. Messaoudi, and H. Schwenk, "Speech transcription in multiple languages," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, vol. 3, Montreal, Canada, May 2004, pp. 757–760.