HIERARCHICAL STATISTICAL LANGUAGE MODELS: EXPERIMENTS ON IN-DOMAIN ADAPTATION

Lucian Galescu, James Allen

University of Rochester, USA
{galescu,james}@cs.rochester.edu

ABSTRACT

We introduce a hierarchical statistical language model, represented as a collection of local models plus a general sentence model. We provide an example that mixes a trigram general model and a PFSA local model for the class of decimal numbers, described in terms of sub-word units (graphemes). This model practically extends the vocabulary of the overall model to an infinite size, but still has better performance compared to a word-based model.

Using in-domain language model adaptation experiments, we show that local models can encode enough linguistic information, if well trained, that they may be ported to new language models without re-estimation.

1. INTRODUCTION

Language models are an important component of large-vocabulary speech recognizers. N-gram models are currently the most widely used, despite their obvious shortcomings. It seems, however, that these models have reached their limits, and the many variations proposed in recent years have not provided much improvement. Language is structured, mostly in a hierarchical way, and without being able to take advantage of structural constraints, further progress in statistical language modelling is hard to achieve. In this paper we propose a model that is inherently structural, thus allowing for incorporating some linguistic knowledge, a better model of context, and for integrating sub-language models.

The model we propose is based on a "building block" strategy and is similar in many ways to other models found in recent literature ([1, 7, 8, 13, 20]). It assumes that language models can be derived for certain classes of constituents (in a broad sense) independently of how the rest of the linguistic data is modelled. Each constituent model can incorporate other constituent models as linguistic units. For example, a model for currency phrases may use a sub-model for numbers. The topmost model effectively connects all the various blocks (a block can be as simple as a single word). Intuitively, a hierarchical statistical language model (HSLM) is a collection of sub-models organized in a hierarchy. Each sub-model can generate strings of symbols from an alphabet of its own, or can insert the output of other sub-models lower in the hierarchy. As we want the overall model to be probabilistic, the only requirement is that each sub-model returns a probability for every word or word sequence they might cover. As usual, other restrictions may be imposed for reasons of efficiency and availability of training data for estimating the models' parameters. However, due to the modularity of the model, these restrictions can be made separately for each sub-model, and need not be global, as in the more conventional models. For the same reason, the basic units of the models need not be all words. We used a model employing sub-word units, which would generate an infinite vocabulary. In contrast, the approaches similar to ours and, in fact, most of the language models currently in use have only finite coverage.

The hierarchical framework facilitates the introduction of more linguistic (structural) information than n-gram models do, and allows for incremental adaptation of the various sub-models without re-training the whole model. In this paper we present the results of an in-domain language model adaptation experiment on the WSJ corpus. The goal is to prove that local models may indeed capture linguistic generalizations that can be transferred during adaptation. The HSLM model used for exemplification has only two layers: a trigram top layer, and a probabilistic finites-state automaton (PFSA) local model for the sublanguage of numbers (which is infinite). During adaptation, the trigram component is adapted by interpolation, while the sublanguage model is kept unchanged. We show that this adapted model compares favorably to a word-based language model adapted by interpolation.

2. THE MODEL

The model proposed here, which we call a *hierarchical hybrid statistical language model*, can be thought of as a generalization of class-based and phrase-based n-gram models.

Traditional class-based models assume a simple (uniform or unigram) distribution of words in each class. The modelling is done at the level of the word; sometimes, phrases are also accounted for, usually by lexicalizing them – this, in fact, provides a rudimentary linguistic model, as pointed out in [18]. We believe it would be beneficial to allow for richer structure inside the class model, and also for more sophisticated probability distributions inside the classes. For example, each class can be modelled with a PFSA, or even a probabilistic context-free grammar (PCFG) if it is small enough not to raise questions of efficiency ([8, 20]). Note also that PCFGs may be approximated by PFSAs ([14]). Or else, they could be n-gram models, themselves. We also made use of regular expressions, which are immediately convertible to FSAs, which could then be stochasticized. We call the model *hybrid* because we don't require all components of a model to be based on the same architecture, but rather that each have the most appropriate structure given the sublanguage that it tries to model and the amount of training data available for it. For practical reasons, though, we ask that each of them return a probability for every word or word sequence they might cover, so that the overall model remains probabilistic. Of course, hybrid models need special decoding mechanisms, but we think the approach is feasible, if care is taken that each sub-model lends itself to efficient decoding.

An input utterance $\mathbf{W}=w_1..w_n$ can be segmented into a sequence $\mathbf{T}=t_1..t_m$, where each t_i is a tag that stands for a subsequence $\bar{u}_{t_i}$ of words in **W**; in the trivial case that the subsequence is composed of one word only, we identify the tag with word. Thus, in the hierarchical model, the likelihood of **W** under the segmentation **T** is

$$P(\mathbf{W},\mathbf{T}) = \prod_{i=1}^{n} [P(t_i \mid t_1..t_{i-1}) P(\bar{u}_{t_i} \mid t_i)], \quad (1)$$

The likelihood of a word sequence given its tag is given by the local probability models $P_{t_i}(\cdot) = P(\cdot \mid t_i)$ and can be expressed as in (1) in terms of models lower in the hierarchy. This lends the overall model a *hierarchical structure*.

In general there might be several segmentations at each level corresponding to one word sequence, in which case the total probability should be obtained by summing over all segmentations. We restrict ourselves here to a model in which there is only one segmentation.

The modelling doesn't have to be at the level of the word. There may be constructs for which better modelling can be done at sub-word level. We exemplify in this paper with a model for numbers (for text), but the same principle can be applied to other entities. One direction we would like to pursue is modelling of proper names at sub-word level (syllable). General-purpose language models based on sub-word units have not proved as good as the word-based ones, so we would like to retain the rich statistical information present in the frequent words, and for the infrequent ones to include components based on smaller units. This would have the effect of increasing the coverage of the overall language model with just a small size increase, and alleviating the data sparseness problem for the overall model. For the local model it would be possible to gather training data from different sources, thus making for more reliable estimation.

The architecture of the local models could be, in the most obvious cases, designed by humans ([1, 5, 7, 10-12, 15, 21]). These models would be linguistically sound, and would allow for more intuitive parameterization. [13] and [20] use automatic clustering of phrases based on the syntactic and/or semantic categories assigned to them by a parser. Distributional clustering of class phrases is done in [16]; their models, although not presented as such, can be represented as two- or three-layer hierarchical models.

The construction of the model proceeds by identifying in the training data, using either a tokenizer or a parser, the words and phrases that belong to a certain class, and replace their occurrences with class-specific tags. The top-level model is trained in the usual way on the tokenized corpus. The data collected for each class is used for training the local model for that class.

3. IN-DOMAIN ADAPTATION

We use a simple in-domain adaptation technique, based on linear interpolation. The adapted model is a mixture of two components, a general model trained on a large background corpus, and a domain-specific model trained on a small adaptation corpus. For a more detailed description of this and other adaptation techniques, we refer the reader to [3].

This technique has proved useful in accounting for variations in topic and/or style. It can also be used for cross-domain adaptation, where the background corpus and the adaptation corpus are more dissimilar, but there results are not so good, especially at the level of trigrams. We restrict ourselves to an easier problem, as the goal here is not in improving on the adaptation techniques, but on showing that local models can be portable and that other things being equal, the adapted hierarchical model performs better than a conventional adapted model.

To this end, we train on the background corpus a simple hierarchical model composed of a trigram general model and one local PFSA model based on sub-word units. We then adapt as explained above the general component using a small adaptation corpus. The local model is transferred without change in architecture or parameters.

4. EXPERIMENTAL RESULTS

We performed experiments on newspaper text (WSJ). For training the background model we selected the first 200k sentences (about 4.5M words) from the WSJ87 corpus. For adaptation and testing we selected the WSJ89 corpus. We expect it to be fairly similar to the background corpus in style, but many topics would be different, and, quite likely, there would be some vocabulary differences.

The first half of the WSJ89 corpus, about 30k sentences (746k words), was used for adaptation. A third of the remaining data was held out for development and the rest, 20k sentences (about 500k words) was used for testing, and was not seen beforehand.

It is known that the adaptation by linear interpolation technique helps most when the adaptation data is very small, and less so when there is more adaptation available. We tested the performance of our models both using the full adaptation corpus and a smaller subset (about 40% of it).

4.1. The Models and the Adaptation Procedure

The baseline model is a word-based trigram model, with Witten-Bell discounting and 0-1 cutoffs, built with the CMU-Cambridge SLM Toolkit ([2]).

The background model is a word trigram with a vocabulary comprising the most frequent 20,000 words. This vocabulary is expanded with 6,004 more words that appear in the 20,000 most frequent words in the adaptation corpus. This basic vocabulary was fixed for all the word-based models; for the top-level components of the hierarchical models, the vocabulary included all non-number words in the basic vocabulary, plus a tag for the class of numbers.

From the adaptation data a trigram was estimated, and then interpolated with the background model to give the baseline adapted model. The interpolation weights were optimized on the held-out data, using the Expectation Maximization algorithm ([4]).

	APP					*PP*				
	bkg	**adapt1**	**interp1**	**adapt2**	**interp2**	**bkg**	**adapt1**	**interp1**	**adapt2**	**interp2**
baseline	259.11	376.31	209.30	255.42	183.57	186.90	271.44	150.97	184.24	132.41
HSLM	240.49	344.96	197.66	240.04	174.68	183.38	263.04	150.72	183.04	133.20
reduction	7.19%	8.33%	5.56%	6.02%	4.84%	-	-	-	-	-

Table 1: *APP* and *PP* results for the word-based (**baseline**) and the hierarchical model (**HSLM**) and relative *APP* reductions achieved.

Our HSLM models have only two layers, a trigram top layer, and a PFSA model for the sublanguage of numbers (which is infinite). During adaptation, the trigram component is adapted by interpolation with a trigram model estimated on the tokenized adaptation data, while the sublanguage model is kept unchanged. The adaptation conditions for the trigram component were the same as for the word-based model.

The local model describes decimal numbers at the graphemic level, i.e., the basic units are characters. We identified decimal numbers using the following regular expression[1]:

```
((0-9)+(,[0-9][0-9][0-9])*)(.[0-9]*)?
```

for which we built an equivalent 7-state deterministic FSA (Figure 1). This automaton was turned into a PFSA; the probability model has six parameters, and was trained on all the numbers found in the training data, including those not in the basic vocabulary.

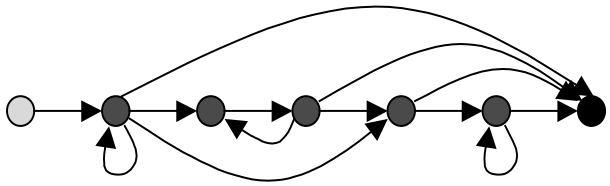

Figure 1: FSA model for the class of numbers. We omit the arc labels, but note that the arcs are labeled with multiple symbols.

The model assigns to all numbers an estimate of what the probability of a random number should be; the estimate is normalized on the training data. Thus, this model takes into account the fact that for a finite amount of test data only a certain proportion of words will be numbers. Note that the probability depends only on the fact that the word seen in the test data is a number, and not on which number it is; in particular, numbers not encountered in the training data will receive the same probability as the ones encountered. For a comparison between this probability model and a more conventional one, see [5].

4.2. Results

As the hierarchical model can recognize more numbers than those in the basic vocabulary, we cannot use *perplexity* (*PP*) to compare the performance of the two models ([17]). Although we computed *PP* values for all models, we only give them for the curiosity of the reader. The actual comparisons are made in terms of *adjusted perplexity* (*APP*), a measure introduced in [19], which adjusts the value of *PP* by a quantity dependent on the number of unknown words in the test set, and the number of their occurrences. We compute the *APP* value on the full test set, and thus we can compare two models with different vocabularies. We again refer the reader to [5] for more details on our evaluation procedure.

The main results are depicted in Table 1. The significance of the headings is as follows:

- **bkg** are the models trained on the background corpus only;
- **adapt** are the models trained on the adaptation corpora only. We marked with **1** the models trained on the smaller adaptation corpus, and with **2** the ones trained on the full adaptation corpus;
- **interp** are the adapted models, resulting from the interpolation of the appropriate **bkg** and **adapt** models, as described above.

All HSLM models performed better than the word-based models. The HSLM-adapted model brought significant *APP* improvements, and, of course, better coverage compared to the baseline (the OOV rate is reduced with about 1.5% relative). As expected, for all measures, the benefit is larger when there is less adaptation data. Although the reductions may not seem very large, given that numbers account for less than 2% of the test data, they are quite impressive.

It is interesting to see that the largest reduction is observed on the models trained on the small adaptation corpora only. This suggests that with a few good local models, a hierarchical model could be a reasonable initial language model for a new domain, when there is little training data and background corpora similar to the target domain, since it requires less training data to achieve the same performance as a word-based model. It might be interesting to test this hypothesis in conjunction to other techniques for building initial models (E.g., [6]).

These results show that the local model is indeed able to capture generalizations about the data that can be ported effectively to new models. Also, part of the improvements are due to the better context modelling, since words following the numbers are going to be predicted based on the class tag, and not the number itself.

5. CONCLUSION

We introduced a hierarchical statistical language model that generalizes over most of the previous variations of n-gram

[1] Compare to the ones given in [9].

models. We hinted at some of the advantages we can expect, and provided an example that mixes two different models, a trigram general model and a PFSA local model for the class of decimal numbers, described in terms of sub-word units (graphemes). This model practically extends the vocabulary of the overall model to an infinite size, but still has better performance compared to a word-based model.

We experimented with in-domain language model adaptation and showed that hierarchical models compare favorably to word-based language models. This suggests that local models can encode linguistic information that may be ported to new language models without re-estimation.

Other simple sub-models can be developed and integrated easily. As more structural information is introduced in the model, we expect additional improvements. Spoken language recognition experiments are scheduled for the near future.

6. ACKNOWLEDGEMENTS

This work was supported by the following grants: ONR research grant no. N00014-95-1-1088, DARPA research grant no. F30602-98-2-0133, US. Air Force/Rome Labs research contract no. F30602-95-1-0025, and NSF research grant no. IRI-9623665.

7. REFERENCES

1. Brugnara, F., and M. Federico. “Dynamic language models for interactive speech applications”. In Proc. EUROSPEECH, pp. 2751–2754, 1997.

2. Clarkson, P., and R. Rosenfeld, "Statistical Language modelling using the CMU-Cambridge Toolkit," Proc. EUROSPEECH, pp. 2707–2710, 1997.

3. DeMori, R., and M. Federico, "Language Model Adaptation". In *Computational Models of Speech Pattern Processing*, Keith Pointing (ed.), NATO ASI Series, Springer Verlag, 1999.

4. Dempster, A.P., N.M. Laird, and D.B. Rubin. "Maximum Likelihood from Incomplete Data via the EM Algorithm". *Journal of the Royal statistical Society*, Series B, 39(1): 1-38, 1977.

5. Galescu, L., and J. Allen. “Evaluating Hierarchical Hybrid Statistical Language Models”. In these Proc., ICSLP, 2000.

6. Galescu, L., E.K. Ringger, and J.F. Allen. "Rapid Language Model Development for New Task Domains". In Proc. LREC, 1998.

7. Giachin, E.P. “Automatic training of stochastic finite-state language models for speech understanding”. In Proc. ICASSP, pp. 173–176, 1992.

8. Gillett, J. and W. Ward. “A Language Model Combining Trigrams and Stochastic Context-free Grammars”. In Proc. ICSLP, pp. 2319–2322, 1998.

9. Grefenstette, G., and P. Tapanainen. “What is a word, what is a sentence? Problems of tokenization”. In Proc. 3rd Int. Conf. Comp. Lexicography, pp. 79--87, 1994.

10. Guyon, I., and F. Pereira. “Design of a Linguistic Postprocessor using Variable Memory Length Markov Models”. In Proc. 3rd Int. Conf. Document Anal. and Recognition, pp. 454–457, 1995.

11. Meteer, M., and J.R. Rohlicek. “Statistical Language Modeling Combining N-gram and Context-free Grammars”. In Proc. ICASSP, vol. II, pp. 37–40, 1993.

12. Moore, R.C., D. Appelt, J. Dowding, J.M. Gawron, and D. Moran. “Combining Linguistic and Statistical Knowledge Sources in Natural-Language Processing for ATIS”. In Proc. of the Spoken Language Systems Technology Workshop, pp. 261–264. Morgan Kaufmann, 1995.

13. Nasr, A., Y. Estéve, F. Béchet, T. Spriet, and R. de Mori. “A Language Model Combining N-grams and Stochastic Finite State Automata”. In Proc. EUROSPEECH, pp. 2175–2178, 1999.

14. Pereira, F.C.N., and R.N. Wright. 1997. “Finite-state approximation of phrase-structure grammars”. In Emmanuel Roche and Yves Schabes, editors, *Finite-State Language Processing*. MIT Press, Cambridge, pp 149–173.

15. G. Riccardi, R. Pieraccini, and E. Bocchieri. “Stochastic Automata for Language Modeling”. *Computer Speech & Language*, 10(4):265–293, 1996.

16. Ries, K., F.D. Buø, and A. Waibel. “Class Phrase Models for Language Modeling”. In Proc. ICSLP, pp. 398–401, 1996.

17. Roucos, S. “Measuring perplexity of language models used in speech recognizers”. Technical report, BBN Laboratories, 1987.

18. Seneff, S. “The Use of Linguistic Hierarchies in Speech Understanding”. In Proc. ICSLP, pp. 3321–3330, 1998.

19. Ueberla, J. “Analyzing and Improving Statistical Language Models for Speech Recognition”. PhD thesis, Simon Fraser University, Vancouver, Canada, 1994.

20. Wang, Y.-Y., M. Mahajan, and X. Huang. “A Unified Context-Free Grammar and N-Gram Model for Spoken Language Processing”. In Proc. ICASSP, 2000.

21. Ward, W., and S. Issar. “A Class Based Language Model For Speech Recognition”. In Proc. ICASSP, pp. 416–419, 1996.