THE USE OF A LINGUISTICALLY MOTIVATED LANGUAGE MODEL IN CONVERSATIONAL SPEECH RECOGNITION

Wen Wang[1,2] *Andreas Stolcke*[1] *Mary P. Harper*[2]

[1] Speech Technology and Research Laboratory, SRI International, Menlo Park, CA 94025
[2] Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285
{wwang,stolcke}@speech.sri.com, harper@ecn.purdue.edu

ABSTRACT

Structured language models have recently been shown to give significant improvements in large-vocabulary recognition relative to traditional word N-gram models, but typically imply a heavy computational burden and have not been applied to large training sets or complex recognition systems. In previous work, we developed a linguistically motivated and computationally efficient almost-parsing language model using a data structure derived from Constraint Dependency Grammar parses that tightly integrates knowledge of words, lexical features, and syntactic constraints. In this paper we show that such a model can be used effectively and efficiently in all stages of a complex, multi-pass conversational telephone speech recognition system. Compared to a state-of-the-art 4-gram interpolated word- and class-based language model, we obtained a 6.2% relative word error reduction (a 1.6% absolute reduction) on a recent NIST evaluation set.

1. INTRODUCTION

Structured language models (LMs) have recently been shown to give significant improvements in large-vocabulary recognition relative to traditional word N-gram models, but often imply a heavy computational burden at training and/or test time, and have therefore not been used in state-of-the-art research systems. Such systems are typically trained on hundreds of hours of speech and hundreds of millions of words of text and transcripts. In [1, 2], we developed an almost-parsing language model (LM) within the Constraint Dependency Grammar framework that is computationally efficient because it does not require expectation maximization (EM) in training and takes the form of a simple class-based N-gram model in testing. We have evaluated the almost-parsing LM on a variety of large-vocabulary continuous speech recognition (CSR) tasks in an N-best rescoring framework, and found that it reduced recognition error rates significantly, achieving performance competitive with state-of-the-art parser-based LMs [3, 4]. In this paper, we investigate the performance of the almost-parsing LM in conversational telephone speech (CTS) recognition, in the context of a complex, multi-pass recognition system. Section 2 provides a brief description of SRI's English CTS system and the acoustic and language model training data. Section 3 reviews the SuperARV almost-parsing LM and Section 4 describes how we integrated the LM into the recognition system. Results are reported in Section 5, followed by conclusions.

This research was supported in part by Purdue Research Foundation, National Science Foundation under Grant No. BCS-9980054, and DARPA under Contract No. MDA972-02-C-0038. Distribution is unlimited. The Hub5 experiments were conducted by the first author at SRI International as a part of her doctoral dissertation. Part of this work was carried out while the third author was on leave at National Science Foundation. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the view of DARPA or the National Science Foundation.

2. THE SRI ENGLISH CTS SYSTEM

2.1. Brief system description

The SRI 2003 CTS system performs an almost-parallel decoding of the speech data using two sets of acoustic models, one based on HLDA-normalized MFCC cepstral features, the other based on PLP features normalized using a traditional LDA followed by an MLLT diagonalizing transformation [5]. Both MFCC and PLP subsystems use 3rd-order differentials in the original feature vectors. At various points in the processing, the two systems exchange information via cross-adaptation and finally N-best ROVER system combination [5]. Language models of increasing orders are used for initial decoding and lattice generation, lattice expansion and rescoring, and finally N-best rescoring. Details on the processing stages are presented in Section 5.

2.2. Training data

Acoustic models were trained on LDC's Switchboard-1 corpus, the Credit Card corpus, the CallHome English corpus, transcribed Switchboard Cellular data released by LDC, and the Switchboard-2 data transcribed by CTRAN and released by BBN, for a total of 418 hours of speech.

The class-conditioned mixture LMs used in the baseline system and almost-parsing LMs were trained on the acoustic training transcripts for all the above sources, as well as the 1996 Broadcast News Hub-4 LM training corpus (130M words). An additional 191M words of LM training data were retrieved from the Web through the Google search engine by searching for conversational N-grams extracted from the CTS transcripts [6]. Furthermore, 102M words of data relevant to the topics of the Switchboard-2 and LDC Fisher data collections was selected from Google newsgroups, in an attempt to better match the unseen CTS test data drawn from those collections [5].

3. THE SUPERARV LANGUAGE MODEL

The SuperARV LM [1] is a highly lexicalized probabilistic LM based on Constraint Dependency Grammars (CDGs). It tightly integrates multiple knowledge sources, for example, word identity, lexical features that have synergy with syntactic analyses (e.g., gapp, mood), and syntactic and semantic constraints at both the *knowledge representation level* and *model level*.

The first type of integration was achieved by introducing a linguistic structure, called a super abstract role value (*SuperARV*),

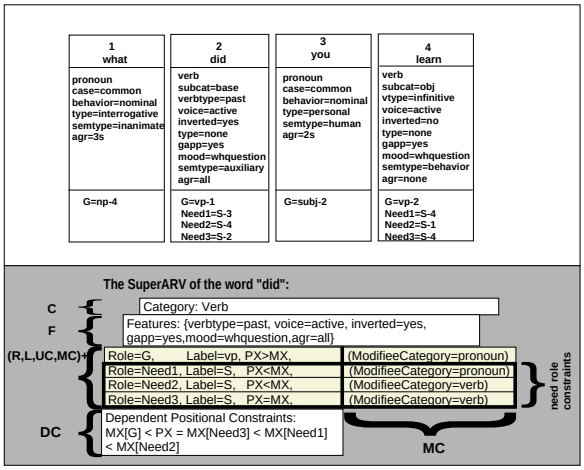

Fig. 1. The SuperARV for the word *did* given the CDG parse for the sentence *what did you learn*. Note: *G* represents the governor role; the *Need1*, *Need2*, and *Need3* roles are used to ensure that the requirements of the word are met. PX and MX represent the position of a word and its modifiee, respectively.

to encode multiple knowledge sources in a uniform representation that is much more fine-grained than part-of-speech (POS). A SuperARV is an abstraction of the joint assignment of dependencies for a word, which provides a mechanism for lexicalizing CDG parse rules. The gray box of Figure 1 presents an example of a SuperARV for the word *did*, which is derived from the dependency parse of the sentence *What did you learn* depicted in the white box of Figure 1. Each word in the parse has a lexical category, a set of feature values, and a governor role (denoted G) which is assigned a role value, comprised of a label, as well as a modifiee, which indicates the position of the word's governor or head. For example, the role value assigned to the governor role of *did* is `vp-1`, where its label `vp` indicates its grammatical function and its modifiee `1` is the position of its head *what*. The words in the parse can also have need roles (denoted *Need1*, *Need2*, and *Need3*), which are used to ensure the grammatical requirements (e.g., subcategorization) of a word are met. Note that the verb *did* needs a subject (*Need1*) and a base form verb (*Need2*), but since the word takes no other complements, the modifiee of the role value assigned to *Need3* is set equal to its own position.

A SuperARV is formally defined as a four-tuple for a word, $\langle C, F, (R, L, UC, MC)+, DC\rangle$, where C is the lexical category of the word, $F = \{Fname_1 = Fvalue_1, \ldots, Fname_f = Fvalue_f\}$ is a feature vector ($Fname_i$ is the name of a feature and $Fvalue_i$ is its corresponding value), (R, L, UC, MC)+ is a list of one or more four-tuples, each representing an abstraction of a role value assignment, where R is a role variable (e.g., governor), L is a functionality label (e.g., np), UC represents the relative position relation of a word and its dependent (i.e., modifiee), MC is the lexical category of the modifiee for this dependency relation, and DC represents the relative ordering of the positions of a word and all of its modifiees. Notice that the SuperARV structure for *did* provides an explicit way to combine information about its lexical features with one consistent set of dependency links for the word that can be directly derived from its parse assignments. A SuperARV can be thought of as providing admissibility constraints on syntactic and lexical environments in which a word may be used. Once SuperARVs are assigned to a word sequence, a parse for the sentence can be produced by the constrained operation of deciding dependencies to link the SuperARVs together.

The model-level integration was accomplished by jointly estimating the probabilities of a sequence of words w_1^N and their SuperARV membership t_1^N:

$$\begin{aligned} Pr(w_1^N t_1^N) &= \prod_{i=1}^{N} Pr(w_i t_i | w_1^{i-1} t_1^{i-1}) \\ &= \prod_{i=1}^{N} Pr(t_i | w_1^{i-1} t_1^{i-1}) \cdot Pr(w_i | w_1^{i-1} t_1^{i}) \end{aligned}$$

We use this to enable the joint prediction of words and their SuperARVs so that word identity information is tightly integrated at the model level. Note that SuperARVs serve as hidden events for constraining word prediction and the SuperARV LM is fundamentally a class-based LM using SuperARVs as classes. Since a large number of classes would seriously reduce the quality of a class-based LM, the number of SuperARVs needs to be controlled. Encouragingly, we found that the number of SuperARVs scales up quite well as the training set size increases. On our language modeling tasks, a moderate-sized corpus with 25,168 words produces 328 SuperARVs, compared to 791 for the 37M word LM training set of the WSJ CSR task, 1,612 for the 300M word Hub4 Broadcast News LM training set, and 622 for the 300+M word CTS LM training data.

Since the parameter space for the SuperARV LM is larger than a word-based LM, in [1] we evaluated several smoothing algorithms and how to interpolate with or backoff to lower-order n-gram probability estimations. For each smoothing algorithm investigated, we used a combination of heuristics and mutual information values to globally determine the lower-order n-grams to include in the interpolation, as well as their ordering. For a trigram SuperARV LM on the DARPA WSJ CSR task, the modified Kneser-Ney smoothing algorithm [7] showed the best performance. A detailed description of the best order of interpolation appears in [1].

The SuperARV LM must be trained on a corpus of CDG parses. However, since there is no CDG treebank (except for the DARPA Naval Resource Management task), we have developed a methodology to automatically transform context-free grammar (CFG) constituent bracketing into CDG annotations [1]. In addition to generating dependency structures by headword percolation [3], our transformer utilizes rule-based methods to determine lexical features and need role values for the words in a parse. Although these procedures are effective, they cannot guarantee that the CDG annotations generated are completely correct. In [2], the impact of errorful training data on the SuperARV LM was investigated on the Hub4 Broadcast News CSR task. Two state-of-the-art parsers, Collins' lexicalized probabilistic CFG parser [8] and Charniak's maximum-entropy inspired parser [9], were chosen based on accuracy, robustness, and mutual consistency [2] to generate CFG parses. The resulting CFG treebank was then transformed to CDG parses for training the SuperARV LM. We found that the SuperARV LM is effective even when trained on inconsistent and errorful training data. Encouraged by these results, we use similar methods to investigate the performance of the SuperARV LM within the SRI Hub5 CTS recognition system.

4. APPLYING SUPERARV LM IN SRI CTS SYSTEM

4.1. SRI CTS baseline LMs and SuperARV LMs

The SRI CTS system, similarly to other multiple-pass speech recognition systems, employs computationally inexpensive decoding steps first to generate intermediate results that constrain the search space, followed by rescoring with more sophisticated acoustic and language models. In our original CTS system, a bigram class-conditioned mixture LM [6] is used during the first pass of acoustic decoding, and a trigram class-conditioned mixture LM is used for lattice expansion. For rescoring N-best hypotheses, a 4-gram class-conditioned mixture LM was employed as well as a standard 400-class class-based LM built from the CTS sources alone, using classes induced from bigram mutual information statistics [10]. The 4-gram class-conditioned mixture LM was interpolated with the 400-class class-based LM during N-best rescoring steps, using fixed weights of $(0.8, 0.2)$. For the 2-gram, 3-gram, and 4-gram class-conditioned mixture LMs (all in ARPA format), separate LMs were built for each data source; all source-specific LMs, with the exception of the web-topic LM, were then combined by class-conditioned interpolation [6], and the resulting mixture LM was interpolated with the topic-related LM using fixed weighting of $(0.8, 0.2)$. The interpolation of word N-grams was static and resulted in a single combined backoff N-gram model, as described in [11]. Note these configurations of bigram, trigram, and 4-gram LMs were applied in the baseline SRI CTS system.

To train a SuperARV LM, similarly to our procedure on the Hub-4 Broadcast News CSR task, we parsed the sentences in the LM training data using Collins' and Charniak's parsers to generate their CFG parse trees and then transformed the trees to CDG parses. Just as for the 4-gram class-conditioned mixture LM, a separate SuperARV 4-gram LM was trained for each available source and the resulting source-specific LMs were then combined into a single model, with the weights obtained by minimizing the perplexity on a held-out development test set.

4.2. SuperARV LM integration into recognition system

To enable efficient integration of the SuperARV LM into the recognition system we generated an ARPA-style backoff LM based on the SuperARV word probability estimates. Note that the SuperARV language model, as a class-based LM, is theoretically able to estimate probabilities for any word sequence; however, to keep the generated word LM to a reasonable size, N-gram pruning similar to [12] is applied. The pruning threshold is tuned on a development set to achieve a satisfactory balance between LM size and perplexity (in future work, we plan to investigate pruning methods that more directly optimize recognition performance). The word 4-gram SuperARV LM thus obtained was used in the N-best rescoring stages of our system. Note this procedure of "dumping" a SuperARV LM into an ARPA-style backoff LM is for the purpose of efficiently integrating it into the SRI CTS system using the SRILM toolkit. This efficiency compromise may have reduced the effectiveness of the SuperARV LM. The ARPA-format SuperARV 4-gram was also interpolated with the 400-class class-based LM during N-best rescoring steps, using fixed weights of $(0.8, 0.2)$.

As an expedient to leverage the SuperARV LM in the earlier stages of the recognition system, we used the "LM rescoring" feature of the SRILM toolkit [11]. We replaced bigram and trigram probability estimates in the baseline system (for the initial decoding and lattice expansion stages) with SuperARV LM probability estimates (backing off as needed for lack of full 4-gram word contexts). Following this replacement, backoff weights are recomputed to normalize the LMS. We call the updated bigram and trigram models "SuperARV conditioned" LMs.

Table 1. The perplexity values on the development set from the 4-gram class-conditioned mixture LM, the SuperARV 4-gram LM, and their interpolations with the 400-class class-based LM, respectively.

LMs	Perplexity
4-gram class-conditioned mixture LM	64.34
The same 4-gram interpolated with the 400-class LM	62.45
SuperARV 4-gram LM	53.74
SuperARV 4-gram interpolated with the 400-class LM	53.74

5. RESULTS AND DISCUSSION

5.1. Perplexity

Table 1 shows the perplexity on the development set for the 4-gram class-conditioned mixture LM and the SuperARV 4-gram LM. As can be seen, the SuperARV 4-gram LM achieves a relative perplexity reduction of 16.5% compared to the 4-gram class-conditioned mixture LM. However, its interpolation with the 400-class LM does not yield a further perplexity reduction. This may be due to the fact that the SuperARV LM, which is itself a class-based LM, already captures much of the knowledge concerning similarities among word distributions.

5.2. Recognition error rates

The SRI CTS system includes the following processing stages:

Step 1: Waveform segmentation using a speech/nonspeech HMM, gender identification, and the estimation of vocal tract length as well as MFCC and PLP feature computation and normalization;

Step 2 First-pass N-best decoding (we used N=2000) after phone-loop MLLR adaptation of MFCC within-word MMIE-trained triphone models, N-best decoding using a conditioned bigram LM and the adapted models, and N-best rescoring using the interpolated word and 400-class 4-gram LMs, phone-in-word duration and pause models [13], and pronunciation probabilities. The confusion-network based score combination and hypothesis selection are then performed. The best hypotheses are generated using N-best ROVER;

Step 3: MLLR adaptation of acoustic models used in Step 2 using the 1-best hypotheses generated in Step 2, lattice generation using the adapted model and a conditioned bigram LM, and lattice expansion using a conditioned trigram LM;

Step 4: N-best decoding from lattices after phone-loop adaptation of PLP within-word triphone models;

Step 5: N-best decoding from lattices after hypothesis adaptation of MFCC SAT MLE crossword model;

Step 6: N-best decoding from lattices after hypothesis adaptation of PLP SAT MLE crossword model;

Step 7: N-best decoding from lattices after hypothesis adaptation of MFCC non-SAT MMIE-trained non-crossword model;

Table 2. WER (%) and absolute reductions on the RT-02 tuning set and test set from the baseline system and the new run using all updated LMs (i.e., the SuperARV-conditioned bigram and trigram and the SuperARV 4-gram LM). The values in parentheses for each step are the absolute WER reductions from using the updated LMs over the baseline.

	WER (%)(absolute reduction)			
	Tuning Set		Test Set	
Step	Baseline	SARV	Baseline	SARV
2	31.4	29.9 (-1.5)	32.5	31.4 (-1.1)
4	30.1	29.1 (-1.0)	31.2	30.1 (-1.1)
5	26.4	25.3 (-1.1)	27.9	26.4 (-1.5)
6	26.3	24.9 (-1.4)	27.7	26.1 (-1.6)
7	26.4	25.0 (-1.4)	27.9	26.2 (-1.7)
8	25.9	24.4 (-1.5)	27.5	25.5 (-2.0)
9	25.4	24.4 (-1.0)	26.9	25.4 (-1.5)
10	25.0	23.6 (-1.4)	26.3	24.6 (-1.7)

Step 8: N-best decoding from lattices after hypothesis adaptation of MFCC SAT MMIE-trained crossword model;

Step 9: N-best decoding from lattices after hypothesis adaptation of PLP SAT MMIE-trained crossword model;

Step 10: 3-way N-best ROVER combination of the rescored N-best lists from Steps 7, 8, and 9;

Step 11: Forced alignment on the hypotheses from Step 10 to generate the word times and estimate confidences.

Note that from Step 4 on, for all steps involving N-best decoding, N-best rescoring was performed as in Step 2 and the best hypotheses were generated using N-best ROVER [14].

Table 2 compares the word error rates (WERs) for the SRI DARPA RT-03 Spring CTS system (which used the baseline LMs) with a modified system that uses the SuperARV LMs. The testset is the DARPA RT-02 CTS evaluation data. A tuning set of 48 speakers was used to optimize the knowledge source combination weights for both the old and the new LMs, and the remaining 72 speakers served as an unbiased test set; the table reports the results on both subsets separately. The evaluation system is otherwise unmodified.

The use of SuperARV LMs gives a significant reduction in word error rate between 1.0% and 2.0% absolute, reducing the final WER on the complete RT-02 set (i.e., tuning and test sets) by 1.6% (from 25.8% to 24.2%). The improvement on the complete RT-02 set is 1.3% in Step 2 (from 32.1% to 30.8%), where it is unaffected by adaptation and cross-adaptation of systems. Note that later stages benefit from both the improved LM and the better quality of adaptation hypotheses in earlier stages. Not shown in the table is the fact that the SuperARV conditioned bigram, prior to the first N-best rescoring in Step 2, yielded a 0.7% absolute lower WER. This is a rather surprising result given the limited context used for word prediction in that model. If we apply the LMs used in the baseline system during each pass, and apply the SuperARV LM only in the last pass to rescore the N-best lists from Steps 7, 8, and 9 and conduct the 3-way N-best ROVER combination, we achieved only a 0.7% absolute WER reduction. This suggests that it is important to apply the SuperARV LM as early in decoding as possible and to continue to use that knowledge during each pass.

We showed how the SuperARV approach to structured language modeling, an almost-parsing class-based LM based on Constraint Dependency Grammar parses, scales extremely well to complex, large-vocabulary recognition systems. By converting the SuperARV LM into the word N-gram ARPA LM format, no additional computational effort is incurred at recognition time, and the model can be used in all stages of a multi-pass CSR system, giving 6.2% relative WER reduction on a standard CTS recognition task.

The SuperARV framework was originally developed for read and planned speech, which tends to exhibit more standard grammatical structures. Our model does not yet include any provisions for dealing with the special features of conversational speech, such as incomplete sentences and disfluencies, and future work will be aimed at modeling these features.

6. REFERENCES

[1] W. Wang and M. Harper, "The SuperARV language model:Investigating the effectiveness of tightly integrating multiple knowledge sources", *in Proceedings of Conference of Empirical Methods in Natural Language Processing*, 2002.

[2] W. Wang, M. P. Harper, and A. Stolcke, "The robustness of an almost-parsing language model given errorful training data", *in Proc. ICASSP*, vol. 1, pp. 240–243, Hong Kong, Apr. 2003.

[3] C. Chelba and F. Jelinek, "Exploiting syntactic structure for language modeling", *in Proc. COLING-ACL*, vol. 1, pp. 225–231, Montreal, 1998.

[4] B. Roark, "Probabilistic top-down parsing and language modeling", *Computational Linguistics*, vol. 27, pp. 249–276, 2001.

[5] A. Stolcke, H. Franco, R. Gadde, M. Graciarena, K. Precoda, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, Y. Huang, B. Peskin, I. Bulyko, M. Ostendorf, and K. Kirchhoff, "Speech-to-text research at SRI-ICSI-UW", *in DARPA RT-03 Workshop*, Boston, May 2003, http://www.nist.gov/speech/tests/rt/rt2003/spring/presentations/sri+-rt03-stt.pdf.

[6] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures", in M. Hearst and M. Ostendorf, editors, *Proc. HLT-NAACL 2003*, vol. 2, pp. 7–9, Edmonton, Alberta, Canada, Mar. 2001. Association for Computational Linguistics.

[7] S. F. Chen and J. T. Goodman, "An empirical study of smoothing techniques for language modeling", Technical report, Harvard University, Computer Science Group, 1998.

[8] M. Collins, *Head-Driven Statistical Models for Natural Language Parsing*, PhD thesis, University of Pennsylvania, 1999.

[9] E. Charniak, "A Maximum-Entropy-Inspired Parser", *in Proceedings of the First Annual Meeting of the North American Association for Computational Linguistics*, 2000.

[10] P. F. Brown, V. J. D. Pietra, P. V. deSouza, J. C. Lai, and R. L. Mercer, "Class-based n-gram models of natural language", *Computational Linguistics*, vol. 18, pp. 467–479, 1992.

[11] A. Stolcke, "SRILM—an extensible language modeling toolkit", in J. H. L. Hansen and B. Pellom, editors, *Proc. ICSLP*, vol. 2, pp. 901–904, Denver, Sep. 2002.

[12] A. Stolcke, "Entropy-based pruning of backoff language models", *in Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pp. 270–274, Lansdowne, VA, Feb. 1998. Morgan Kaufmann.

[13] D. Vergyri, A. Stolcke, V. R. R. Gadde, L. Ferrer, and E. Shriberg, "Prosodic knowledge sources for automatic speech recognition", *in Proc. ICASSP*, vol. 1, pp. 208–211, Hong Kong, Apr. 2003.

[14] A. Stolcke, H. Bratt, J. Butzberger, H. Franco, V. R. Rao Gadde, M. Plauché, C. Richey, E. Shriberg, K. Sönmez, F. Weng, and J. Zheng, "The SRI March 2000 Hub-5 conversational speech transcription system", *in Proceedings NIST Speech Transcription Workshop*, College Park, MD, May 2000.