DISCRIMINATIVE LEARNING USING LINGUISTIC FEATURES TO RESCORE N-BEST SPEECH HYPOTHESES

Maria Georgescul, Manny Rayner, Pierrette Bouillon, Nikos Tsourakis

ISSCO/TIM, ETI, University of Geneva

ABSTRACT

We describe how we were able to improve the accuracy of a medium-vocabulary spoken dialog system by re-scoring the list of n-best recognition hypotheses using a combination of acoustic, syntactic, semantic and discourse information. The non-acoustic features are extracted from different intermediate processing results produced by the natural language processing module, and automatically filtered. We apply discriminative support vector learning designed for re-ranking, using both word error rate and semantic error rate as ranking target value, and evaluating using five-fold cross-validation; to show robustness of our method, confidence intervals for word and semantic error rates are computed via bootstrap sampling. The reduction in semantic error rate, from 19% to 11%, is statistically significant at 0.01 level.

***Index Terms*—** natural language, speech recognition

1. INTRODUCTION

The extremely resource-intensive nature of the speech recognition decoding process means that most recognizers are forced to use language models that restrict themselves to modeling the surface language in the domain. It is consequently tempting to add a post-processing phrase, which applies more powerful knowledge sources to reorder a list of possible speech hypotheses produced by the recognizer. This procedure is often called "N-best rescoring". Typically, the N-best rescoring phase adds syntactic information that was not included in the language model in order to improve acoustic and semantic confidence scores [1, 2]. For example, if the language model is bigram-based, rescoring can use trigrams. Alternately, if the model is trigram-based, rescoring can use grammar-based parsing [3-8].

In the present study, which uses a spoken interface to a calendar database, we take the process a step further. The recognizer's vocabulary is quite small (about 220 words), and the base language model, which has the form of a probabilistic context-free grammar, already encodes a great deal of syntactic and distributional information. It is consequently surprising to discover that N-best rescoring was still able to deliver a substantial improvement in performance. Features are extracted from several different levels in the speech understanding process, and combined using machine learning.

The rest of the paper is organized as follows. Section 2 describes the system and the data used. The features we exploit are described in Section 3, and the feature selection criteria in Section 4. Section 5 describes the experiments performed and the results obtained. The final section concludes.

2. SYSTEM AND DATA

The Calendar application used for the experiments is described in [9] and runs on top of the Nuance 8.5 platform. The user is able to use spoken queries to access the content of a small meeting database, and ask about dates, times, locations, attendance etc. The interpretation of questions may depend on preceding context, in particular to determine referents for pronouns ("it", "he") and definite descriptions ("the meeting"), or to resolve ellipsis ("Is there a meeting on Monday?" … "Tuesday?"). The language model is built using the Regulus toolkit [10], which constructs a PCFG grammar, in Nuance format, starting from a linguistically motivated feature grammar for English. The Nuance-format grammar contains a total of 1724 CFG productions.

For our experiments, we used data collected in a small user study with five subjects, who were given simple information-retrieval problems to solve together with a few representative examples of system queries. The user utterances were recorded as SPHERE-headed wavform files, manually transcribed, and automatically evaluated to determine whether or not they were within the coverage of the grammar. This produced 527 transcribed utterances, of which 480 were in-coverage. Since grammar-based language models behave very differently on in-coverage and out-of-coverage data, we only used the in-coverage portion for our experiments. Each in-coverage utterance was tagged as "semantically correct" or "semantically incorrect". An utterance U was tagged as "semantically correct" if either a) the recognition result for U was the same as the transcription, or b) a non-trivial semantic representation could be produced for U, and this representation was the same as the one which would have been produced from the transcription.

We were surprised, when testing early versions of the system, to discover that speech understanding performance was very poor compared to that of apparently similar Regulus applications; 1-best Word Error Rate (WER) on in-coverage data was around 11% and Semantic Error Rate (SemER) around 19%. By way of comparison, the English recognizer from the MedSLT application [11], with twice as large a vocabulary, had WER around 6% and SemER around 10%. Closer examination suggested to us that the problems arose primarily from the nature of the domain. In particular, articles and other short words are unusually important in Calendar utterances. For example, "When is **the** meeting in Geneva?" asks for the start time for a specific, contextually determined meeting in Geneva, while "When is **a** meeting in Geneva?" asks for start-times for all such meetings; similarly, "next week" refers to a 7 day period starting on Sunday, while "**the** next week" refers to a 7 day period starting now. Another important problem is the easy confusability of sentence-initial "is" and "was", which means that tense is often misrecognised. Tense is semantically important; for example "Is there a meeting on Monday?" refers to next Monday, while "**Was** there a meeting on Monday?" refers to last Monday.

On the positive side, we discovered that the difference between 1-best and 5-best WER was unusually large. In other Regulus application we have examined, 5-best WER has typically been about 80% of 1-best WER; for Calendar, 5-best WER was as low as 45% of 1-best WER. This suggested to us that N-best rescoring might well have more to offer than usual.

3. FEATURES

On the basis of *ad hoc* testing with the live system, we decided to use four types of feature, each of which was taken from different levels of processing. First, we include the information derived directly from the recognizer. Specifically, we use two features: the Nuance confidence score (an integer between 1 and 100), and the rank in the N-best list. Although these two features are obviously correlated, the rank gives important extra information; empirically, the first hypothesis is much more likely to be correct than the second, even if the difference in confidence scores is minimal.

The second feature group encodes syntactic information, and is taken from the parsed representation of the hypothesis. For example, as noted above, it was clear that confusions between definite and indefinite articles were a major problem; we consequently defined features encoding a few common types of context, for example existential constructions ("Is there…") where syntactic evidence made one of these more plausible. Another feature marked imperatives where the main verb was not "show" or something similar. Other features in this group classify the surface-semantic representation as one of a small number of types, for example "WH-question" or "Elliptical phrase". Here, the main intuition was that elliptical phrases are less frequent than full questions.

The third group of features is defined on the deep semantic representation of the query, and reflects the intuition that some queries are *logically* implausible. In particular, one feature attempts to identify underconstrained queries; we defined a query to be underconstrained if it made no reference to dates, times, locations or people. A second feature attempted to identify queries, normally resulting from misrecognitions, where the tense is inconsistent with other temporal constraints. A typical example would be "What meetings **were** there next week?".

The final group of features is based on the system response that would be produced for each hypothesis. This time, the intuition is that some types of response are inherently more plausible than others. For example, other things being equal, an error or clarification response is less plausible than a simple answer; a large set of individuals is less plausible than a small set (usually this means the query is underconstrained); and a negative answer is less plausible than a positive one. We separated the space of possible responses into six types, *"say yes"*, *"say no"*, *"say nothing found"*, *"say list of n referents"*, *"say clarification question"* and *"other"*, and defined one binary-valued feature for each type.

Concretely, we produced 5-best recognition hypothesis lists for each utterance, and passed each hypothesis through all levels of processing up to and including production of a possible response. (It is of course relevant here that main dialogue processing is side-effect free). For each hypothesis, features were extracted from the vector of intermediate processing results.

4. FEATURE SELECTION

In order to find the linguistic features that discriminate most effectively between semantically correct and incorrect classes, we used the following feature selection criterion.

We start with a dataset consisting *m* vectors $x_i = (x_{i1}, x_{i2}, ..., x_{in})$, where *n* is the total number of linguistic features measured and x_{ij} corresponds to the *j*-th linguistic feature measured for the *i*-th hypothesis. Each hypothesis is labeled with $y_i \in \{0,1\}$, i.e. semantically correct vs. semantically incorrect recognition utterances.

For each feature values given by the vector $(x_{1j}, x_{2j}, ..., x_{nj})$, we compute the mean μ_j^+ and standard deviation σ_j^+ for those features x_{ij} which correspond to hypothesis that are semantically correct. Similarly, we compute the mean μ_j^- and standard deviation σ_j^- for those x_{ij} which correspond to hypothesis that are semantically incorrect. Then, we associate to each feature the following score:

$$\mathrm{score}(x_j) = \left| \frac{\mu_j^+ - \mu_j^-}{\sigma_j^+ + \sigma_j^-} \right| .$$

This score gives highest values to those linguistic features whose values differ to the greatest extent on average in the respective (hypothesis) classes. Also high score values will be given to linguistic features that have small deviations scores in different classes.

The scores associated with each linguistic feature were used in order to discard all features having score values smaller than a given threshold. The features with the highest scores are given in Table 1. Features with scores lower than the threshold value of 0.05 were discarded.

Feature	**Score**
underconstrained query	0.45
nbest rank	0.35
confidence ASR	0.27
response type – say nothing found	0.24
response type – say list of referents	0.15
response type – no response	0.14
response type – say no	0.14
dialogue move – on date	0.14
non indefinite existential	0.13
response type – other	0.11
lf context available	0.11
elliptical utterance	0.10
dialogue move – tense information	0.09
inconsistent tense	0.07
resolution	0.06
dialogue move – utterance type	0.05
non show imperative	0.05
indefinite meeting and meeting referent	0.05

Table 1: Features and associated scores

5. EXPERIMENTAL SETTINGS AND RESULTS

We applied support vector machine (SVM) learning designed for reranking [12]. The experiments were carried out in two versions, respectively using WER and SemER as the ranking target values.

We performed SVM learning experiments in two phases. In a first phase we used the set of features obtained via the filter feature selection method described in Section 4. However, there is a bias difference between the evaluation score described in Section 4 and SVM ranking. That is, the experimental error rates obtained after reranking with these features proved to be rather close to the baseline error rates. Therefore, in a second phase, we applied a backward sequential selection method for the set of features selected via the filtering method. In this manner, features such as *"resolution"*, *"inconsistent tense"* and the features derived from the categorical features *"response type"* and *"dialogue move"* were discarded.

Algorithm	**WER**	**WER Confidence Interval**	**SemER**	**SemER Confidence Interval**
Baseline (1-best)	11.17		18.85	
SVM, linear kernel	9.24	(7.57, 11.22)	15.16	(11.19, 19.95)
SVM, polynomial kernel (kp = 3)	9.07	(7.41, 11.01)	14.94	(10.98, 19.75)
SVM RBF kernel (kp=1)	8.97	(7.33, 10.79)	14.50	(10.55, 19.34)

Table 2: Percentage error rates and confidence intervals (at a significance level of 0.01) obtained when SVM learning was performed using WER as the re-ranking target value. "Baseline" = result for 1-best recognition.

Algorithm	**WER**	**WER Confidence Interval**	**SemER**	**SemER Confidence Interval**
SVM, linear kernel	9.99	(8.20, 12.03)	11.64	(7.98, 16.09)
SVM, polynomial kernel (kp = 2)	9.80	(8.11, 11.76)	11.42	(7.94, 15.89)
SVM, RBF kernel (kp = 1)	9.74	(8.09, 11.70)	11.42	(7.94, 15.89)

Table 3: Percentage error rates and confidence intervals (at a significance level of 0.01) obtained when SVM learning was performed using SemER as the re-ranking target value.

The final evaluation of our reranking procedure was performed using 5-fold cross-validation. During SVM learning, we used three simple kernels: linear, polynomial and radial basis function (RBF). In Table 2 and Table 3, we provide the average error rates of SVM reranking on testing data, using the best kernel parameter values determined during the model tuning phase. The kernel parameter is labeled as *kp* in the tables. For the evaluation, we measured both WER and SemER.

The experimental results show that the polynomial and radial basis kernels performed nearly equally well compared to the linear kernel as regarding both average WER and SemER. This shows that a more complex kernel is not required for our dataset.

The SemER obtained via SVM reranking procedure using SemER as the re-ranking target value is statistically significantly different from the baseline results at a significance level of 0.01. In order to construct confidence intervals, we apply the bootstrap sampling with replacement technique. In the third and fifth columns of Table 2 and 3, we provide the studentized bootstrap confidence intervals [13] for the average error rate. The 99% confidence interval is computed via nonparametric approximation based on 10000 bootstrap replicates.

In order to integrate the SVM learning module with our dialogue system, we implemented a server which provides reranking for unseen n-best hypothesis and the associated linguistic features (using the best model obtained during the training phase) via TCP/IP.

6. CONCLUSIONS

We have shown how various features automatically computable from the speech recognition and spoken dialogue understanding process could successfully be used to carry out N-best rescoring in a medium-vocabulary spoken dialogue application that used an elaborate grammar-based language model. Our best-performing features reduced SemER from 18.9% to 11.4% (40% relative), and WER from 11.2% to 9.1% (23% relative). Bootstrap sampling showed that the reduction in SemER was statistically significant at 1% level. We found it interesting that optimization of SemER gave much stronger results than optimization of WER.

As mentioned in Section 2, N-best rescoring appears to be unusually suitable in this domain. We are currently developing criteria for identifying other applications where grammar-based N-best rescoring might work well, and evaluating whether suitable generalizations of our methods are useful there.

7. REFERENCES

[1] M. Boros, W. Eckert, F. Gallwitz, G. Görz, G. Hanrieder, and N. H., "Towards Understanding Spontaneous Speech: Word Accuracy vs. Concept Accuracy," in *Proceedings of the Fourth International Conference on Spoken Language Processing (ICSLP)*, Philadelphia, PA, USA, 1996, pp. 1009 - 1012.

[2] A. Chotimongkol and A. I. Rudnicky, "N-best Speech Hypotheses Reordering Using Linear Regression," in *Proceedings of the Seventh European Conference on Speech Communication and Technology (EuroSpeech)*, Aalborg, Denmark, 2001, pp. 1829-1832.

[3] E. Brill, R. Florian, J. C. Henderson, and L. Mangu, "Beyond N-Grams: Can Linguistic Sophistication Improve Language Modeling?," in *Proceedings of the International Conference On Computational Linguistics (COLING)*, Montreal, Canada, 1998, pp. 186 - 190.

[4] M. Gabsdil and O. Lemon, "Combining Acoustic and Pragmatic Features to Predict Recognition Performance in Spoken Dialogue Systems," in *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL)*, Barcelona, Spain, 2004, pp. 343 - 350.

[5] R. Jonson, "Dialogue Context-Based Re-ranking of ASR Hypotheses," in *Proceedings of the Spoken Language Technology Workshop*, Aruba, 2006, pp. 174 - 177.

[6] W. P. McNeilly, J. G. Kahny, D. L. Hillardz, and M. Ostendorfyz, "Parse Structure and Segmentation for Improving Speech Recognition," in *Proceedings of the IEEE/ACL Workshop on Spoken Language Technology*, Aruba, 2006.

[7] M. Rayner, D. Carter, V. Digalakis, and P. Price, "Combining Knowledge Sources to Reorder N-best Speech Hypothesis Lists," in *Proceedings of the Human Language Technology Conference (HLT)*, Plainsboro, NJ, 1994, pp. 217 - 221.

[8] M. Walker, J. Wright, and I. Langkilde, "Using Natural Language Processing and Discourse Features to Identify Understanding Errors in a Spoken Dialogue System," in *Proceedings of the 17th International Conference on Machine Learning (ICML)*, 2000.

[9] N. Tsourakis, M. Georgescul, P. Bouillon, and M. Rayner, "Building Mobile Spoken Dialogue Applications Using Regulus," in *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco, 2008.

[10] M. Rayner, B. A. Hockey, and P. Bouillon, *Putting Linguistics into Speech Recognition*: Center for the Study of Language and Information /SRI, 2006.

[11] M. Rayner, P. Bouillon, N. Chatzichrisafis, M. Santaholma, M. Starlander, B. A. Hockey, Y. Nakao, H. Isahara, and K. Kanzaki, "MedSLT: A Limited-Domain Unidirectional Grammar-Based Medical Speech Translator," in *Proceedings of the First International Workshop on Medical Speech Translation*, New York, USA, 2006.

[12] T. Joachims, "Optimizing Search Engines Using Clickthrough Data," in *Proceedings of the 8th ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, Edmonton, Alberta, Canada, 2002.

[13] A. C. Davison and D. V. Hinkley, *Bootstrap Methods and their Application*: Cambridge University Press, 1997.