INTERSPEECH
2009 BRIGHTON

Improving Speech Understanding Accuracy with Limited Training Data Using Multiple Language Models and Multiple Understanding Models

Masaki Katsumaru[1], *Mikio Nakano*[2], *Kazunori Komatani*[1],
Kotaro Funakoshi[2], *Tetsuya Ogata*[1], *Hiroshi G. Okuno*[1]

[1]Graduate School of Informatics, Kyoto University, Japan
[2]Honda Research Institute Japan Co., Ltd., Japan
{katumaru,komatani,ogata,okuno}@kuis.kyoto-u.ac.jp
{nakano,funakoshi}@jp.honda-ri.com

Abstract

We aim to improve a speech understanding module with a small amount of training data. A speech understanding module uses a language model (LM) and a language understanding model (LUM). A lot of training data are needed to improve the models. Such data collection is, however, difficult in an actual process of development. We therefore design and develop a new framework that uses multiple LMs and LUMs to improve speech understanding accuracy under various amounts of training data. Even if the amount of available training data is small, each LM and each LUM can deal well with different types of utterances and more utterances are understood by using multiple LM and LUM. As one implementation of the framework, we develop a method for selecting the most appropriate speech understanding result from several candidates. The selection is based on probabilities of correctness calculated by logistic regressions. We evaluate our framework with various amounts of training data.

Index Terms: speech understanding, multiple language models and language understanding models, limited training data

1. Introduction

The speech understanding module in a spoken dialogue system is essential in making a voice user interface (VUI) effective in spoken dialogue systems. The most important issue with speech understanding is how to get high performance at an earlier stage in the development of spoken dialogue systems, that is, *rapid prototyping*. Rapid prototyping is important for both achieving stable service at an earlier stage of system deployment and collecting reliable user responses and data at an earlier stage. The quality of speech understanding usually depends on the amount and quality of training data obtained mainly by such collection. The time and effort spent for such collection are not negligible. Therefore, a rapid-prototyping approach is greatly demanded. Speech understanding consists of an automatic speech recognition (ASR) component and a language understanding (LU) component. The ASR component uses an acoustic model and a language model (LM), while the LU component uses a language understanding model (LUM). To develop a speech understanding component, system developers have to prepare an LM and an LUM. Since the performance of speech understanding greatly depends on the quality of the LM and LUM [1], the selection of the LM and LUM and their combination is critical in delivering spoken dialogue systems in real-world applications. For statistical models, much training data is needed. For finite-state grammar-based models, the grammar should be carefully written by the system developers. Since the amount of training data depends on the time and effort of human participants, domain-dependent training data are particularly difficult to obtain. In other words, there is no universal combination of LM and LUM that works for any application at any stage of development.

Our idea is that, every LM and every LUM performs best for a particular domain or some particular utterances. The best combination of an LM and LUM depends on the utterance. Our objective is to develop a way to combine an LM and LUM so as to obtain high speech understanding accuracy even with a limited amount of training data because system developers can easily collect a small amount of training data, such as a few hundreds of utterances by a few participants.

Conventional studies of speech understanding have led to the development of many types of LMs, such as finite-state grammars and N-grams, and many types of LUMs, such as finite-state transducers (FSTs), weighted FSTs (WFSTs), and keyphrase extractors (Extractors). The optimal combination of an LM and LUM depends on the amount of available training data and the types of utterances to be handled. Conventional studies have identified the LMs and LUMs that give the best performance by using fixed training and fixed test data such as the Air Travel Information System (ATIS) corpus. In system development, the optimal combination of an LM and LUM is not clear because the amount of available training data varies. Therefore, system developers determine the types of LM and LUM to use by trial and error. So far there have been several attempts to improve ASR and speech understanding using multiple speech recognizers and multiple language understanding modules. The ROVER [2] improves ASR accuracy by integrating the outputs of multiple ASRs with different acoustic and language models. It differs from our approach in that it does not deal with speech understanding, and it is based on the assumption that each ASR is well-developed and achieves high accuracy for a variety of speech inputs. Eckert *et al.* [3] used multiple LMs to deal with both in-grammar and out-of-grammar utterances but did not mention language understanding. Hahn *et al.* [4] used multiple LUMs but only a single LM. Fukubayashi *et al.* [5] investigated rapid prototyping of language understanding components and estimated the weights of WFST for an LUM with a small amount of training data. We use their estimation method for WFST in our language understanding component. We previously developed a speech understanding framework called “Multiple Language models and Multiple Understanding models (MLMU)” in which multiple LMs

Copyright © 2009 ISCA

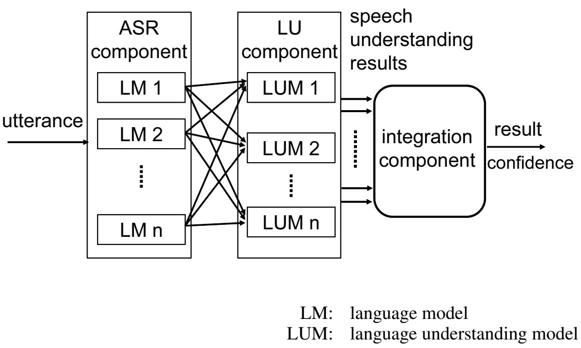

Figure 1: *Overview of speech understanding with MLMU*

and LUMs are used [1]. Using multiple speech understanding models increases the number of utterances that the system can correctly deal with compared with using a single model. The best speech understanding result is selected from the multiple results generated by arbitrary combinations of LMs and LUMs. We demonstrated that speech understanding accuracy was improved when a large amount of training data was available. We have now analyzed the detailed behaviors of our MLMU and determined that the accuracy obtained by MLMU varies with the amount of available training data. The results are useful for estimating the amount of training data that should be collected in advance to deploy a stable spoken dialogue system.

2. Speech understanding framework MLMU

MLMU is a framework by which system developers can use multiple speech understanding methods by preparing multiple LMs and multiple LUMs. Figure 1 illustrates the flow of speech understanding with MLMU. System developers list the available LMs and LUMs for each system domain, and the system understands utterances by using these models. The framework selects one understanding result from the multiple results or calculates a confidence score for the result by using the generated multiple understanding results.

MLMU improves speech understanding for the following reason. The performance of each speech understanding (a combination of an LM and LUM) might not be very high when either training data for the statistical model or available expertise and effort for writing grammar are insufficient. In such cases, some utterances might not be covered by the system's finite-state grammar LM, and probability estimation in the statistical models may be poor. Using multiple speech understanding models should solve this problem because each model has different specialities. For example, finite-state grammar LMs and FST-based LUMs achieve high accuracy in recognizing and understanding in-grammar utterances, whereas out-of-grammar utterances are better recognized and understood by N-gram models and LUMs based on WFST or keyphrase extractors. Therefore, it is more likely that the understanding results of MLMU will include the correct results than when a single understanding model is used.

U1: **On the 9th of June** (*in-grammar utterance*)

FSG + FST	ASR:	on June 9th
	LU:	**month: 6, day: 9**
N-gram + WFST	ASR:	on Noon in June
	LU:	month: 6

Result of FSG + FST was correct; "9th" was misrecognized by N-gram-based ASR.

U2: **On the 9th and 10th** (*out-of-grammar utterance because of existence of the underlined part*)

FSG + FST	ASR:	from Saturday on 9th.
	LU:	day: 9, day-of-week: Sat.
N-gram + WFST	ASR:	on the 9th, 10th
	LU:	**start-day: 9, end-day: 10**

Result of N-gram + WFST was correct; underlined part of utterance was misrecognized by FSG-based ASR.

Figure 2: *Example utterances understood by multiple LMs and LUMs.*

3. Implementation

3.1. Available language models and language understanding models

We implemented MLMU as a library of RIME-TK [6], which is a toolkit for building multi-domain spoken dialogue systems. With the current implementation, developers can use the following LMs:

1. LM based on finite-state grammar (FSG)
2. Domain-dependent statistical N-gram model (N-gram)

and the following LUMs:

1. Finite-state transducer (FST)
2. Weighted FST (WFST)
3. Keyphrase extractor (Extractor).

That is, six speech understanding methods are available. System developers can also use multiple finite-state-grammar-based LMs or N-gram-based LMs and multiple FSTs or WFSTs. They can specify a set of combination for each domain after preparing the LMs and LUMs. Grammar models work well when sufficient time is available for writing grammar, and statistical models work well when a corpus for training models is available.

Figure 2 shows two example utterances that were understood by two speech understanding methods, one based on the combination of an FSG-based LM and an FST-based LUM (FSG + FST) and the other based on the combination of an N-gram-based LM and a WFST-based LUM (N-gram + WFST). The first utterance was understood by using FSG + FST, and the second was understood by using N-gram + WFST. By using multiple understanding methods, we can obtain correct speech understanding results for both utterances if we can select the correct result.

3.2. Selecting understanding result on basis of ASR and LU features

We also implemented a mechanism for selecting the best understanding result. Figure 3 illustrates the mechanism. It selects the result with the highest estimated probability of correctness. Probabilities are estimated for each understanding result by using logistic regression, which uses several ASR and

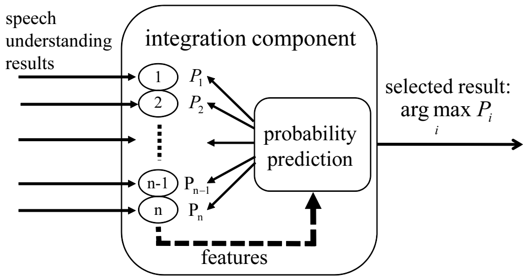

Figure 3: *Implementation of integration component*

LU features. We denote each speech understanding result as *i* (*i* = 1,. . . ,6) and define P_i as the probability that speech understanding result *i* is correct. A result is selected on the basis of $\mathrm{argmax}_i\ P_i$. We constructed logistic regression models for each P_i. The regression function is

$$P_i = \frac{1}{1 + exp(-(a_{i1}F_{i1} + \ldots + a_{im}F_{im} + b_i))}. \quad (1)$$

The coefficients $a_{i1}, \ldots, a_{im}$ and b_i were fitted using training data. The independent variables $F_{i1}, F_{i2}, ..., F_{im}$ are the features shown in Table 1. In the table, *n* denotes the number of understanding results, that is, *n* = 6 in our experiment.

Features F_{i1} to F_{i3} represent the characteristics of the ASR results. The acoustic scores were normalized by the utterance duration. F_{i1} is the acoustic score of understanding result *i* and represents the likelihood of ASR. F_{i2} shows the difference between the acoustic score of *i* and acoustic score by ASR using a domain-independent LM for utterance verification. F_{i2} verifies the correctness of the ASR result. F_{i3} is utterance duration in seconds. Features F_{i4} to F_{i9} represent the characteristics of the LU results. Features F_{i4} to F_{i6} are defined on the basis of the concept-based confidence scores [7]. calculated using posterior probabilities based on the N-best LU results. Features F_{i7} to F_{i9} are defined on the basis of the number of concepts contained in the LU result.

4. Experimental evaluation

In an experiment to evaluate on framework we used the two LMs and three LUMs introduced in Section 3.1. We used a concept error rate (CER) to represent the speech understanding accuracy, which is calculated as follows:

$$CER = \frac{\#\ \text{concept errors}}{\#\ \text{concepts in utterances}}. \quad (2)$$

Concept errors consist of insertion, deletion, and substitution errors. We investigated CERs when the amount of training data to estimate the coefficients of regression functions varied.

4.1. Preparing LMs and LUMs

The FSG rules were written in sentence units by a system developer. A domain-dependent statistical N-gram model (N = 3) was trained using 10,000 sentences randomly generated from the grammar. The vocabulary sizes of the grammar LM and the domain-dependent statistical LM were both 278. We also used a domain-independent statistical N-gram model to obtain acoustic scores for utterance verification; it was trained using Web text [8] and had a vocabulary size of 60,250.

Table 1: *Features of speech understanding result i*

F_{i1} :	acoustic score of ASR
F_{i2} :	difference between F_{i1} and acoustic score of ASR for utterance verification
F_{i3} :	utterance duration [sec.]
F_{i4} :	average confidence scores for concepts in *i*
F_{i5} :	average of F_{i4} ($\frac{1}{n}\sum_i^n F_{i4}$)
F_{i6} :	proportion of F_{i4} (F_{i4} / $\sum_i^n F_{i5}$)
F_{i7} :	average # concepts ($\frac{1}{n}\sum_i^n$ #concept$_\mathrm{i}$)
F_{i8} :	max. # concepts (max (#concept$_\mathrm{i}$))
F_{i9} :	min. # concepts (min (#concept$_\mathrm{i}$))

The grammar used in the FST-based LM was the same as the FSG used for ASR. Two developers wrote the grammar but did not use real user utterances to make its coverage wide. The WFST-based LU was based on a method that can estimate WFST parameters with a small amount of data [5]. Its parameters were estimated by using 105 utterances of only one user. A keyphrase extractor extracted as many concepts as possible from the ASR results on the basis of grammar while ignoring words that did not match the grammar. It is noteworthy that only the WFST-based LUM required a small amount of training data, and the other LMs and LUMs required no training data in this experiment.

4.2. Target data for evaluation

We used 3091 utterances in the rent-a-car reservation domain spoken by 22 participants [9]. We used Julius (ver. 4.0.2) as the speech recognizer and a 3000-state phonetic tied-mixture triphone model as the acoustic model[1]. ASR accuracies in mora accuracy when using the FSG and the N-gram model were 71.9% and 75.5% respectively. We manually annotated whether the understanding result of each utterance was correct or not and used the annotations as training data to fit the coefficients $a_{i1}, \ldots, a_{im}$ and b_i.

4.3. Experimental results

We fitted the coefficients of the regression functions and selected understanding results with a 4-fold cross-validation. We used only one participant's utterances to estimate the coefficients for simulating the case when only a small amount of training data is available. The participant whose utterances were used as training data was different for each fold, and the average number of utterances for each fold was 154.

Table 2 lists the CERs for combinations of a single LM and LUM and for our method. Of all combinations of a single LM and LUM, the best accuracy was obtained with (5) (N-gram + WFST). The accuracy of (1) (FSG + FST) was higher than (2) that of (FSG + WFST) because the amount of data for estimating the weights of WFST was only 105 utterances, and the weights were not estimated properly. The accuracy of (4) (N-gram + FSG) was the worst because the FST did not accept some ASR results generated by the N-gram model. The CER with our method was 3.1 points better than with (5). The number of times each understanding result was selected with our method is shown in Table 3. The table shows that all combinations of a single LM and LUM were selected with our method. This indicates that using multiple LMs and LUMs is effective. Table 2 shows the accuracy of "oracle selection", in which we

[1] http://julius.sourceforge.jp/

Table 2: *CERs for each speech understanding methods*

speech understanding method (LM + LUM)	CER [%]
(1) FSG + FST	26.8
(2) FSG + WFST	27.9
(3) FSG + Extractor	27.0
(4) N-gram + FST	35.5
(5) N-gram + WFST	25.1
(6) N-gram + Extractor	26.1
selection from (1) through (6) (our method)	22.0
oracle selection	13.5

Table 3: *Number of times each speech understanding result was selected with our method*

(1)	(2)	(3)	(4)	(5)	(6)	total
388	392	798	534	430	549	3091

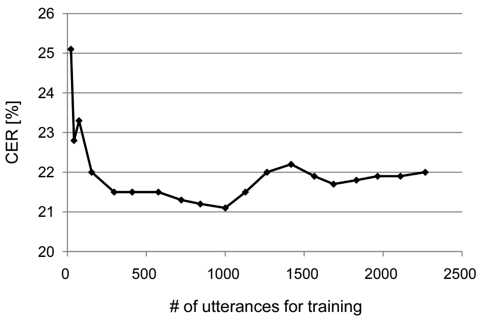

Figure 4: *Change in CER with various amounts of training data*

Table 4: *CER increase when a feature was removed*

Removed feature	F_{i8}	F_{i2}	F_{i7}	F_{i1}	F_{i3}	F_{i5}
CER increase [%]	1.1	0.5	0.5	0.2	0.2	0.2

manually selected the best speech understanding result from the six results. The CER with oracle selection was 13.5%, which is much better than all combinations of a single LM and LUM. This shows that using multiple LMs and multiple LUMs can potentially improve speech understanding accuracy.

Figure 4 shows CERs calculated after we changed the amount of training data with which we estimated the coefficients of the regression functions. The accuracy when 200 utterances were used was almost the same as when more training utterances were used. This indicates that our framework enables high speech understanding accuracy to be obtained when only a small amount of training data is available. However, the CER with our method was still much higher than with the oracle method even when the amount of training data was increased. We thus need to improve the selection method and features.

To identify which features play an important role in the selection, we calculated the CER after each feature was removed one by one. The top six features that increased the CER when they were removed are listed in Table 4. The table shows the number of concepts in speech understanding results and acoustic scores are effective to select a reliable result from several candidates.

5. Conclusion

We have experimentally demonstrated that high speech understanding accuracy can be obtained even with a small amount of training data by using multiple language models (LMs) and language understanding models (LUMs). The concept error rate (CER) with our method was lower than any combination of a single LM and LUM with 154 utterances as training data. This means that our framework should be effective in real system development when a large amount of training data is not available. Furthermore, "oracle selection" showed the potentiality of this framework for greatly improving speech understanding accuracy.

We will conduct more experimental evaluations for other domains to prove the effectiveness of our framework. In this work, we selected one speech understanding result to integrate several speech understanding results obtained with two LMs and three LUMs. There are other LMs and LUMs described in [10], and methods for integrating them, such as voting [4]. We will investigate these models and methods. Finding a way to calculate confidence scores for speech understanding results for efficient dialogue management is also planned.

6. References

[1] M. Katsumaru, M. Nakano, K. Komatani, K. Funakoshi, T. Ogata, and H. G. Okuno, "A speech understanding framework that uses multiple language models and multiple understanding models," in *Proc. NAACL-HLT*, 2009, pp. 133–136.

[2] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)," in *Proc. ASRU*, 1997, pp. 347–354.

[3] W. Eckert, F. Gallwitz, and H. Niemann, "Combining stochastic and linguistic language models for recognition of spontaneous speech," in *Proc. ICASSP*, 1996, pp. 423–426.

[4] S. Hahn, P. Lehnen, and H. Ney, "System combination for spoken language understanding," in *Proc. Interspeech*, 2008, pp. 236–239.

[5] Y. Fukubayashi, K. Komatani, M. Nakano, K. Funakoshi, H. Tsujino, T. Ogata, and H. G. Okuno, "Rapid prototyping of robust language understanding modules for spoken dialogue systems," in *Proc. IJCNLP*, 2008, pp. 210–216.

[6] M. Nakano, K. Funakoshi, Y. Hasegawa, and H. Tsujino, "A framework for building conversational agents based on a multi-expert model," in *Proc. SIGdial*, 2008, pp. 88–91.

[7] K. Komatani and T. Kawahara, "Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output," in *Proc. COLING*, vol. 1, 2000, pp. 467–473.

[8] T. Kawahara, A. Lee, K. Takeda, K. Itou, and K. Shikano, "Recent progress of open-source LVCSR Engine Julius and Japanese model repository," in *Proc. ICSLP*, 2004, pp. 3069–3072.

[9] M. Nakano, Y. Nagano, K. Funakoshi, T. Ito, K. Araki, Y. Hasegawa, and H. Tsujino, "Analysis of user reactions to turn-taking failures in spoken dialogue systems," in *Proc. SIGdial*, 2007, pp. 120–123.

[10] R. D. Mori, F. Bechet, D. Hakkani-Tur, M. McTear, G. Riccardi, and G. Tur, "Spoken language understanding," *Signal Processing Magazine, IEEE*, vol. 25, pp. 50–58, 2008.