

Combination of Finite State Automata and Neural Network for Spoken Language Understanding

Chai Wutiwivatchai, Sadaoki Furui

Department of Computer Science
Tokyo Institute of Technology, Japan
{chai, furui}@furui.cs.titech.ac.jp

Abstract

This paper proposes a novel approach for spoken language understanding based on a combination of weighted finite state automata and an artificial neural network. The former machine acts as a robust parser, which extracts some semantic information called subframes from an input sentence, then the latter machine interprets a concept of the sentence by considering the existence of subframes and their scores obtained from the automata. With a large number of concepts handled in our mixed-initiative dialogue system, the proposed system achieves a considerable concept interpretation result on either a typed-in test set or a spoken test set. A high subframe recall rate also verifies an applicability of the proposed system.

1. Introduction

A pioneering mixed-initiative spoken dialogue system with Thai language interaction has been constructed in a domain of hotel reservation [1]. Lack of resources for the new language, especially annotated corpora, has caused difficulty in the invention. Consequently except for the speech recognition engine, most parts of our first system were constructed based on handcrafted rules. The preliminary evaluation showed that a considerable size of errors came from high out-of-vocabulary (OOV) and out-of-concept (OOC) rates in speech recognition and language understanding respectively. Reducing the OOV can be easily achieved by adding recognizer lexicon entries. However, extension of concepts that the system can handle needs expansion of linguistic knowledge and a complicated annotated corpus.

Instead of improving the handcrafted rule-based understanding system implemented in the first version, we have recently tried to create a new system that can be automatically trained by a given corpus. Many research projects have split the spoken language understanding task into two consecutive subsystems namely speech recognition and understanding. Since the speech recognizer is not the main focus of this article, its task is equivalent to finding the most likely meaning \tilde{M} by searching over the possible word strings resulted from the recognizer.

$$\tilde{M} = \arg \max_M P(M|W) \quad (1)$$

In our system, the meaning M is represented by the combination of a concept C and a set of semantic slots S . We observe that the concept C can be interpreted from the set S , if a proper space of S is defined. Therefore,

$$\begin{aligned} P(M|W) &= P(C, S|W) \\ &= P(C|S, W)P(S|W) \\ &= P(C|S)P(S|W) \end{aligned} \quad (2)$$

From the Eqs. 1 and 2, we get

$$\tilde{M} = \arg \max_{C, S} P(C|S)P(S|W) \quad (3)$$

Searching for the set of semantic slots S that maximize the probability $P(S|W)$ is the function of a *semantic parser*, whereas maximization of the probability $P(C|S)$ is performed by a *semantic or concept interpreter*. Many understanding systems interpret the sentence concept together with parsing its semantic details using a set of rules, parse trees, or decoding networks [2, 4]. Some researchers proposed to classify the sentence concept prior to the semantic parsing [5].

In Thai language, there is no definite or indefinite article, no verb conjugation, no noun declension, no object pronoun, and past and future tenses are often indicated only by context or with the words "already" or "will" tacked on. This causes a lot of words insignificant to the main meaning of the sentence. To deal with such a language, it is necessary to find an efficient semantic parser that can handle a large number of insignificant words. The parser must also be able to deal with under-specified grammar sentences happened commonly in speaking style, as well as incomplete sentences corrupted by recognition errors. We need to construct an effective concept interpreter that is suitably connected to the parser.

In this paper, a novel technique for a spoken language understanding system is proposed. Instead of using a full syntactic parser, a word/phrase spotting technique, which has been proven to be efficient for natural language understanding [6] is applied to an input sentence. This produces a set of semantic slots, defined in this paper as subframes, and likelihood scores. With the likelihood scores as input features for an artificial neural network, a final sentence concept is extracted. It can be noted that the number of concepts to be classified must be large in order to achieve a highly mixed-initiative dialogue system. Next section describes more details about system architecture as well as the way we implemented. The understanding system can be trained using a corpus annotated by a non-expert linguist. The procedure to prepare the corpus is explained in Section 3. Although the system is trained by a sentence set collected via keyboard, it works well when evaluated on either keyboard-based or speech recognition-based sentences as shown in Section 4. Finally Section 5 gives a conclusion.

2. System Architecture

Our spoken language understanding system consists of two components, a semantic parser which in our work is called *subframe extraction* module, and a *concept interpretation* module. Figure 1 illustrates the overall system architecture. The idea and implementation detail of each subsystem are described below.

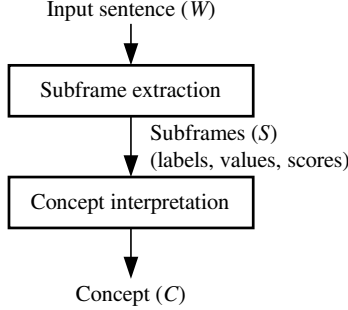


Figure 1: Overall system architecture.

2.1. Subframe extraction

As described in the introduction, an idea behind the subframe extraction module is a robust parser that can extract a set of semantic slots called *subframes*. A subframe consists of a label and an optional value. Figure 2 gives some examples of sentences and apparent subframes. A subframe of “facility” must have a value such as “pool”, whereas a subframe of “yesnoq” means yes/no question and needs no additional value. The subframes should be defined to achieve the following properties.

- A subframe has a unique semantic meaning.
- Order of subframes in a sentence is not important.
- Each type of subframe occurs only once in a sentence.
- The semantic meaning of a subframe can be interpreted from a sequence of words/fillers arbitrarily placed in the sentence (the sequences can overlap or cross each other).

“from the sixth two nights to the eighth of July”		
Subframe label	Subframe value	Corresponding word sequence (x = filler)
fromdate	July-6	from the sixth x of July
todate	July-8	x to the eighth of July
numnight	2	x two nights x
“there is a pool, right?”		
Subframe label	Subframe value	Corresponding word sequence (x = filler)
reqprovide	-	there is x right
facility	pool	x pool x
yesnoq	-	x right

Figure 2: Examples of sentences and their corresponding subframes.

The way we implement is similar to that of a phrase-spotting engine except the definition of units extracted or spotted. A subframe needs to be constructed by one or more phrases to complete its meaning. Unlike a phrase, a word

sequence representing a subframe can be split (separated by filler words).

In the first version of our understanding system, we have created the subframe extraction module based on a set of finite state automata (FSA). Each FSA acts as an acceptor of a subframe. In the current version, we replace the handcrafted FSA by a weighted FSA (WFSA) that can be trained by a corpus. The corpus contains a set of sentences, subframes (labels and values) occurring in each sentence, and word sequences corresponding to the subframes. In the training process, a WFSA is constructed for a subframe using the steps shown in Figure 3.

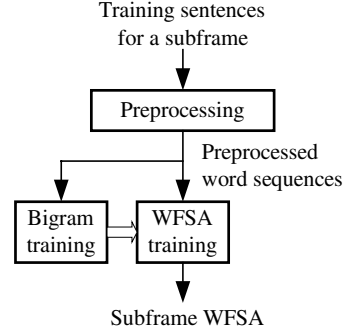


Figure 3: Subframe WFSA training procedure.

In the preprocessing step, words that are not crucial to the current subframe are replaced by filler symbols, connected fillers are reduced to a single filler, and fillers appeared at beginning and end of sentence are removed. Let’s C denote a preprocessed set. For each subframe S , C_S is a subset of C containing sentences in which S is located. In the main training procedure, a WFSA with words/fillers as input labels and negative logarithm of bigram probabilities as weights is constructed from each sentence in C_S and added to a grand WFSA by a union operation. Then the grand WFSA is determinized and minimized. This process is repeated for every sentence in C_S , resulting a final WFSA for the subframe.

In the parsing step, an input sentence is parsed to each of the subframe WFSA. Among some candidate word sequences accepted by a WFSA, one with the lowest score is selected. The score is defined by an average of cumulative weights over the number of words (N) contained in the word sequence as shown in Eq. 4. By this criterion, the longer word sequence with higher average bigram probability over the sequence is likely to be chosen.

$$Score = \frac{\sum_{i=1}^N Weight_i}{N} = \frac{\sum_{i=1}^N (-\log P_{bi})_i}{N} \quad (4)$$

A more robust parser can be achieved by introducing a word class which represents words with a common semantic feature. The use of word class allows us to easily extend the system to cover new words by adding word class entities.

2.2. Concept interpretation

The concept interpretation task can be viewed as a pattern classification problem. A similar idea has been reported in [5], where a classifier is used to determine the topic of a user

by considering the words contained in a sentence. Instead of semantically parsing after the topic classification, we first extract subframes and then classify the sentence concept using the subframe scores.

Many pattern classification techniques, including the Bayes classifier, n-gram model, and support vector machine (SVM), have been conducted for similar tasks [5]. Gorin et al. [7] has proposed a maximum a posteriori (MAP) based method for topic classification, which has been applied to their call routing system. The algorithm is to maximize a posterior distribution,

$$p_i = \max_k P(C_k | S_i) \quad (5)$$

where C_k is the k^{th} class or topic, and S_i is the i^{th} fragment which is a phrase-like sequence of words in a sentence. Then the decision rule selects a class of the fragment with maximum p_i . This technique, denoted in this paper as PMAP, fits to our problem when a subframe replaces the fragment and a concept is a class to be identified. $P(C_k | S_i)$ can be computed simply as the concurrence count of C_k and S_i , divided by the total frequency of S_i .

Compared to the systems constructed for the Air Travel Information (ATIS) domain [3, 5] and the call routing system proposed by AT&T [7], a much larger number of user concepts needs to be recognized in our task which achieves highly mixed-initiative and natural conversational dialogue. In our work, we investigate the use of an artificial neural network (ANN) as a concept classifier and compare it with the use of SVM and PMAP. For the ANN and SVM engines, an input feature, which is a subframe score, is modified from the one given by the subframe extraction module by linear normalization within a range of 0 to 1.

3. Corpus Annotation

Annotated corpora have been built in two steps. The first step is to assign a concept of the sentence and subframes occurred in the sentence. Next, for the sentences containing a focused subframe, words that are meaningful to the subframe are marked.

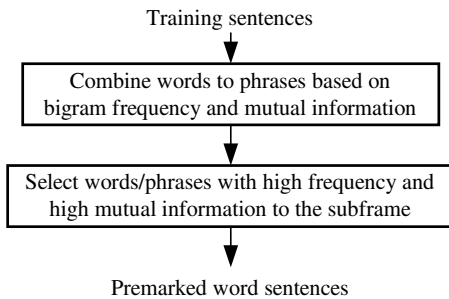


Figure 4: An approach for marking the words that are meaningful to a subframe.

To reduce an effort of manual annotation, a semi-automatic approach has been applied to the corpus annotation. The language understanding system developed in the first version is used to roughly tag the sentence concept and subframes, which are later manually corrected. To mark the words corresponding to a subframe, a strategy shown in Figure 4 has been introduced.

At first, two consecutive words that achieve the highest bigram count, with their mutual information greater than a threshold, are merged to a new word. The process is iterated until the highest bigram count becomes lower than a threshold. Next within each subframe, the word, whose frequency and mutual information to the subframe exceeds a threshold, is marked. Finally the annotation is manually checked and edited using a specific GUI program. This process helps us to gain approximately two hundred annotated sentences per hour by only one non-expert staff.

4. Experiments

We have collected two sets of corpus. The first one was from our specific website which simulated some conversational dialogues in Thai language. Users answered the questions displayed on the screen by typing. The corpus we used in our experiment was collected during November – December 2002. After getting rid of some garbage sentences (out-of-domain, confusing, etc.), it consisted of 6,083 sentences from 150 users. We then annotated all sentences using the procedure explained in the previous section. Each sentence belonged to one of 42 concepts and contained a part of 118 subframes. The first corpus was separated into a training set (TR) of 5,073 sentences and a test set (TS1) of 1,010 sentences. The second corpus containing 403 utterances was collected for the evaluation of the dialogue system version 1 [1]. Two test sets were derived from the second corpus, one was the recognized word sequence set (TS2) with 30.3% WER obtained by the first version speech recognizer, and the other (TS3) was its exact transcription. Some details of each set (except TS2, which is a recognition version of TS3) are shown in Table 1.

Using the training set TR, 115 subframe WFSAs were constructed using the AT&T FSM library tool [8]. Each sentence in the same TR set is parsed with the subframe WFA, which produced a set of accepted subframes and their scores. It is worth noting that an out-of-subframe rate for test sets might not be precisely computed, since we have permitted to annotate only some subframes prominently seen in the sentence.

The existences of subframes and the normalized scores together with the target concept of each sentence in the TR set were used to train a concept interpretator. In the experiment, a simple multilayer perceptron ANN with back-propagation learning algorithm was compared with the PMAP classifier and the SVM. The ANN created by the SNNS toolkit [9] consisted of an input layer with 115 input nodes, each belonged to one subframe, a hidden layer of 100 nodes, and an output layer of 42 nodes corresponding to 42 possible concepts. SVM is normally a binary classifier. To construct a multi-class classifier, many techniques have been proposed and compared [10]. Although some techniques called One-vs-One and DAGSVM have been proven to be the most efficient for multi-class SVM classification, an amount of binary SVMs needed to be constructed, $42 \times 41/2$ models, was not practical. Hence, we tested only a technique called One-vs-Rest, where 42 SVMs were created for 42 concepts.

An important parameter that highly affects the concept interpretation performance is an out-of-concept (OOC) rate. The OOC rate of TS3 set calculated in the system version 1 was 14.9% [1]. Reduction of OOC to only 0.5% as shown in Table 1 indicates that the new developed system has almost

overcome the problem. Table 2 then reports the concept interpretation results using various classifiers for each test set. The results when using the training set for evaluation are also reported in order to show the training capability.

Table 2 clearly shows that the ANN outperforms the other methods in every case. Although the use of SVM has been proven to be very efficient for a similar task [5], there are some reasons to explain why it cannot achieve the best. First, compared to the number of target concepts defined in [5], a much larger number of concepts is required in our task to achieve a mixed-initiative dialogue scheme. Although some techniques for multi-class SVM such as One-vs-One and DAGSVM may produce a better classification performance, they can be hardly implemented as described previously. PMAP has a disadvantage that it has no discriminative ability compared to the ANN. Without discriminative ability, some subframes that are not indicative to a concept but often appear in common, can easily deteriorate the decision made by PMAP.

Table 1: Characteristics of corpora used in our experiment.

Characteristics	TR	TS1	TS3
# Utterances (# Concepts)	5073	1010	403
# Words/utterance	7.5	6.5	6.8
# Subframes	9874	1685	503
OOO rate (%)	-	0.0	0.5

Table 2: Concept interpretation results for various concept classifiers.

Classifier	Interpretation accuracy (%)			
	TR	TS1	TS2	TS3
Rule-based (version 1)	-	-	54.2	60.6
ANN	97.4	87.7	70.0	82.6
SVM	63.2	58.7	59.3	59.3
PMAP	71.0	63.5	65.3	71.1

Although the system has been trained using a corpus collected by typing, it works well with the test sets either transcribed or recognized from the real spoken conversation. Compared to the concept interpretation result evaluated in the system version 1, the new ANN-based version achieves 36.3% and 29.1% improvement when evaluated by the transcribed test set and the recognized test set respectively. Not only a very large improvement has been obtained, but the current system can also be adapted by a new corpus collected in the near future.

Table 3: Subframe precision/recall rate of TS1.

Precision (%)	Recall (%)
66.0	93.4

Another important issue is how well the system detects the subframes contained in a sentence. Some subframes are critical to the dialogue system, whereas some are not. Moreover, a subframe critical to a concept may not impact another concept. Hence to evaluate the subframe detection performance, we must first define which subframes are critical

to each concept. Then precision/recall rate is computed by counting only the impacting subframes. Table 3 shows the subframe precision/recall rate for the test set TS1. The very high recall rate verifies that our understanding system is promising. Since the annotation of subframes was not strictly precise, subframes that were not prominent to the concept were likely to be overlooked in annotation, and it might have lowered the precision rate.

5. Conclusions

A novel approach of spoken language understanding has been proposed. A robust parser based on the phrase spotting technique was conducted in the subframe extraction module. We replaced the handcrafted FSA parser implemented in the system version 1 by the WFSA, which could be trained by a corpus. Parsing an input sentence to the WFSA gave a set of accepted subframes and their scores, which were used by the concept interpretation module. Compared with several successful classifiers, a simple ANN classifier achieved the best performance. Although the proposed system was trained by a corpus collected by typing, it worked well with spoken utterances. By the combination of WFSA and ANN, the system can be easily retrained by a larger spoken corpus.

As the next step, we are planing to optimize our spoken language understanding system, so that it can be connected smoothly to the existing dialogue manager. After improving the dialogue manager and the text generator, a full evaluation of our Thai language dialogue system version 2 will be performed.

6. References

- [1] WutiwWATCHAI, C. and FURUI, S., "Pioneering a Thai Language Spoken Dialogue System", in *Spring Meeting of Acoustic Society of Japan*, 2003.
- [2] Bonneau-Maynard, H. and Lefevre, F., "Investigating Stochastic Speech Understanding", *Proc. ASRU*, 2001.
- [3] Minker, W., Bennacef, S., and Gauvain, J. L., "A Stochastic Case Frame Approach for Natural Language Understanding", *Proc. ICSLP*, 1996.
- [4] Miller, S., Bobrow, R., Ingria, R., and Schwartz, R., "Hidden Understanding Models of Natural Language", *Proc. ACL*, p.25-32, 1994.
- [5] Wang, Y. Y., Aceo, A., Chelba, C., Frey, B., and Wong, L., "Combination Of Statistical And Rule-based Approaches For Spoken Language Understanding", *Proc. ICSLP*, p.609-612, 2002.
- [6] Komatani, K., Tanaka, K., Kashima, H., and Kawahara, T., "Domain-Independent Spoken Dialogue Platform Using Key-Phrase Spotting Based On Combined Language Model", *Proc. Eurospeech*, 2001.
- [7] Gorin, A. L., Riccardi, G., and Wright, J. H., "How May I Help You", *Speech Communication*, 23, p.113-127, 1997.
- [8] AT&T FSM Library, <http://www.research.att.com/sw/tools/fsm/>
- [9] Stuttgart Neural Network Simulator, <http://www-ra.informatik.uni-tuebingen.de/SNNS/>
- [10] Hsu, C. W. and Lin, C. J., "A Comparison of Methods of Multi-class Support Vector Machines", *Tech. Report, National Taiwan University*, 2001.