Data Driven Methods for Utterance Semantic Tagging

Ding Liu, Anthea Cheung, Anna Margolis, Patrick Redmond, Jun-won Suh, Chao Wang

Nuance Communications Inc., Burlington, MA

{ding.liu,anthea.cheung,anna.margolis,patrick.redmond,jun-won.suh,chao.wang}@nuance.com

Abstract

The proliferation of mobile devices, along with advances in speech and natural language processing technologies, have given birth to a new wave of personal assistance applications that enable users to quickly and more naturally perform many tasks through voice on their smart devices. This paper focuses on a natural language understanding (NLU) solution for one such application. We adopted a data-driven approach, aiming to take advantage of large volume of deployment data for continued learning and system improvement. In this paper, we compare two different statistical models---a hidden Markov model and a maximum entropy Markov model---for the task of semantic slot extraction, and we present empirical results on real user data.

Index Terms: natural language understanding, semantic tagging, max entropy model, hidden Markov model

1. Introduction

Spoken dialogue systems allow users to interact with computers and devices using their voice. Early commercial applications of dialogue technology were most common in IVR (interactive voice response) systems for call centers [1]. Recently, driven by the ubiquity of smart phones and by improving spoken language technology, we have seen growing use of voice controlled personal assistant applications. One such product, Siri [2] on the iPhone, is fast becoming a household name. Similar interfaces like Samsung S Voice [3], Dragon Mobile Assistant [4], and Google Now [5] are used by millions of users. This new generation of spoken dialogue systems can simultaneously support a broad range of domains such as voice dialing, sending an SMS, interacting with social media websites, finding/playing multimedia content, surfing the web, etc. While some of these domains are relatively simple, such as changing the phone settings or launching an application, others---like manipulating events in the calendar or making restaurant reservations---require sophisticated natural language understanding and dialogue technologies to ensure successful task completion.

In this paper, we describe a natural language understanding solution for a commercially deployed virtual assistant application on smart phones. The application allows users to access many of the phone's functionalities using their voice, for example, placing calls, sending messages, scheduling meetings, getting directions, publishing social status updates, playing music, searching the web, etc. We adopt a data-driven solution in our approach, aiming to take advantage of large volume of real deployment data for continued learning and system improvement. Usage data from the field are frequently incorporated in improving the statistical NLU models, which are then deployed to achieve improved accuracy on new incoming utterances.

Our NLU processing consists of two stages. In the first stage, a statistical model determines the "intent" of the incoming utterance, such as "alarm creation" or "event lookup." The second stage model will then extract all semantic slots relevant for that hypothesized intent. The semantic slots are predefined for each intent type based on the function requirements of the application. For example, an alarm creation utterance can have an "alarm time" semantic slot, while an event lookup utterance can have semantic slots for "event date", "event time", "event title", "invitee", etc. The rest of the paper focuses on the second stage component, i.e., semantic slot extraction. We first describe the two algorithms we used to perform this task, namely a hidden Markov model, and a maximum entropy Markov model. Section 3 reports evaluation experiments comparing these two methods. We conclude in section 4 and point out some directions for future work.

2. Semantic Slot Extraction

Our semantic slot extraction task is related to information extraction tasks such as semantic role labeling [6] and entity recognition [7], which have been active areas of research for more than a decade. As is common, we frame the task as a sequence tagging problem: given an utterance, we use a sequence model to assign a tag to each word, and then use simple rules to map the tags into semantic slots. For example, given the input utterance "wake me up at 7:00 a.m.", the tagger should assign the "time" tag to the words "7:00" and "a.m.". The words with the "time" tag will be collected into the time slot. All other words should be assigned tags which will not be mapped to any slot. (We call these "null" tags.)

Using $W_{1,N}$ to represent the input utterance of length N and $T_{1,N}$ (T_1, T_2, …, T_N) to represent a semantic tagging sequence candidate, we can formulate the sequence tagging problem as:

$$T_{1,N}^{\ best} = \arg\max_{T_{1,N} \in \Omega(N)} \Pr\left(T_{1,N} \mid W_{1,N}, M(D)\right) \quad (1)$$

Here, D represents the training data, M represent the statistical modeling method, and $\Omega(N)$ represents the set of semantic tag sequences of length N. We explored two different modeling frameworks for the sequence tagging problem: one is based on a Hidden Markov Model (HMM) [8], and the other is a Maximum Entropy Markov Model (MEMM) [9].

Unlike the HMM, which converts the probability of a tag sequence given an utterance into product of transition and emission probabilities, a Max Entropy Markov Model directly models the probability of a tag sequence as the product of the probability of each individual tag given its defined context:

$$\Pr\left(T_{1,N} \mid W_{1,N}\right) = \prod_{i=1}^{N} \Pr_{\max ent}\left(T_i \mid Context(T_i)\right) \quad (2)$$

The context of the model specifies the scope from which the features are extracted for predicting each tag. For example, if the context is defined as the current position plus the position before it, the resulting model is called a first order Maximum Entropy Markov Model:

$$\Pr(T_{1,N} \mid W_{1,N}) = \prod_{i=1}^{N} \Pr(T_i \mid T_{i-1}, W_{i-1}, W_i) \quad (3)$$

Table 1 shows the types of features used in our system[1]. The training of the Max Entropy model follows the SCGIS algorithm with a unified Gaussian variance for all the features.

For training, the semantic slot annotation of each word is converted into a sequence of IOB-2 tags [10]. For decoding, a Viterbi algorithm is used to find the best sequence of IOB-2 tags and then collected into the corresponding semantic slots.

Feature	Description
(lc) w_i	(lower cased) word at position i
(lc) w_{i-1} w_{i-2}	(lower cased) two words before i
(lc) w_{i+1} w_{i+2}	(lower cased) two words after i
prefix-3 w_i	prefixes of w_i up to length 3
dict w_i	the symbol of w_i in the dicts
tag_{i-1} tag_{i-2}	previous 2 tags before i

Table 1: *Features Used in the Maximum Entropy Tagger.*

3. Experiments

The raw training data in our experiments is collected from several sources, including a few previously deployed applications, the Amazon Mechanical Turk, and deployment data from the target application. The raw data is manually transcribed (if audio) and annotated to generate the final training data. Because data from different sources comes in different sizes, a weight is assigned to each data set to balance their proportion in the combined data set. The development and test data are exclusively from the target application.

We first test the accuracy of the semantic slot extraction without ASR errors. Table 2 summarizes the statistics of the training, development and test data. For English and Korean, sentences are segmented into words; for Chinese, they are segmented into characters.

Lang	#Sentence (K)	#Word/Char (K)	#Tag (K)
en-US	2556/990/29	21339/4416/170	11228/430/18
zh-CN	1654/127/14	23839/711/99	9661/66/6.5
ko-KR	2200/307/18	3816/1115/72	9949/168/7

Table 2: *Training/Development/Test Data.*

Table 3 shows the tagging performance of the second order HMM and the Max Entropy model described in Section 2, based on the micro F1-score of the semantic slot prediction computed across all intent types. For Max Entropy models, two recipes are used in the training. *MaxEnt* denotes the one trained with a fixed Gaussian variance of *200*, and using the original utterance counts. *Optimized Maxent* denotes the approach where two parameters are tuned for each Maxent model based on the dev data including the Gaussian variance and the utterance count chunking frequency *c*, used to decrease the impact of the count of the utterances:

$$Utterance_{count} = \max\left(1, \frac{Utterance_{count}}{c}\right) \quad (4)$$

[1] For different languages, some features may not apply. For example, there is no lower cased version of Chinese characters.

Max Entropy model significantly outperforms HMM for all the tested languages in our experiments. By tuning the Gaussian variance and the utterance count chunking frequency, further improvement can be achieved.

Data Set	HMM	MaxEnt	Optimized Maxent
en-US	92.5	94.3	94.7
zh-CN	92.6	94.8	95.7
ko-KR	91.0	92.9	93.5

Table 3: *Tagging F-Score of HMM and MEMM.*

We also evaluate the tagging performance on outputs from an automatic speech recognition system. We are interested in comparing the tagging performance on transcriptions and ASR hypotheses on the same data sets to investigate the impact of ASR accuracy on the end-to-end system performance. Table 4 summarizes the sizes of the test data set, as well as the ASR performance in terms of word and sentence error rates (WER and SER). Because there is some flexibility in defining "words" in Korean ASR vocabulary, we report character error rate to remove potential biases due to differences in ASR vocabulary tokens.

Language	# Sentence	# Tag	SER(%)	WER(%)
en-US	11808	7247	23.26	10.55
ko-KR	7901	4039	20.62*	9.54*

Table 4: *ASR Performance on Audio Test Data. (Note: SER=Sentence Error Rate, WER=Word Error Rate. In the case of ko-KR, performance measures are based on characters.)*

Table 5 compares the performance of HMM and Max Entropy models on both transcription and hypothesis test data.

	HMM		MaxEnt	
	Trans	Hyp	Trans	Hyp
English	0.928	0.715	0.977	0.749
Korean	0.928	0.751	0.963	0.779

Table 5: *Tagging F-Score of HMM and MEMM on Transcriptions and ASR Hypotheses.*

4. Conclusions and Future Work

We compared two different statistical models for the task of semantic slot extraction for spoken utterances from a mobile virtual assistance application, and showed that a Max Entropy model incorporating rich contextual features can significantly outperform an HMM model. We also showed that the tagging accuracy drops significantly when ASR outputs were used instead of the manual transcription, indicating that a more robust integration with ASR for semantic slot extraction is desired to improve NLU systems for spoken inputs.

In the future, we would like to make the NLU more robust against ASR errors by feeding k-best hypotheses or even a word lattice from an ASR decoder to the NLU system to generate more semantic interpretation candidates. The statistical semantic tagging system can also be trained directly with ASR hypotheses to better adapt to noises introduced by ASR system.

5. References

[1] Bernhard. Suhm, Josh. Bers, Dan. McCarthy, Barbara. Freeman, David. Getty, Katherine. Godfrey, and Pat. Peterson, "A comparative study of speech in the call center: natural language call routing vs. touch-tone menus", in Proceeding CHI '02 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Pages 283-290.

[2] http://www.apple.com/ios/siri/

[3] http://www.samsung.com/global/galaxys3/svoice.html

[4] http://www.dragonmobileapps.com

[5] http://www.google.com/landing/now/

[6] D. Gildea, and D. Jurafsy, "Automatic Labeling of Semantic Roles", Computational Linguistics, vol. 28(3), pp. 245-285, 2002.

[7] G. Zhou and J. Su, "Named Entity Recognition Using an HMM-Based Chunk Tagger", in Proc. Assoc. for Computational Linguistics, pp. 473-480, 2002.

[8] L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", in Proceedings of the IEEE, pp. 257-286, 1989.

[9] A. McCallum, D. Freitag, and F. Pereira, "Maximum Entropy Markov Models for Information Extraction and Segmentation", in Proc. 17th Int. Conf. on Machine Learning, vol. 951, pp. 591-598, 2000.

[10] Erik F. Tjong Kim Sang and Jorn Veenstra, "Representing text chunks", in Proceedings of EACL, pp. 173–179, 1999.