Statistical Shallow Semantic Parsing despite Little Training Data

Rahul Bhagat
Information Sciences Institute
University of Southern California
Marina del Rey, CA, 90292, USA
rahul@isi.edu

Anton Leuski
Institute for Creative Technologies
University of Southern California
Marina del Rey, CA, 90292, USA
leuski@ict.usc.edu

Eduard Hovy
Information Sciences Institute
University of Southern California
Marina del Rey, CA, 90292, USA
hovy@isi.edu

1 Introduction and Related Work

Natural language understanding is an essential module in any dialogue system. To obtain satisfactory performance levels, a dialogue system needs a semantic parser/natural language understanding system (NLU) that produces accurate and detailed dialogue oriented semantic output. Recently, a number of semantic parsers trained using either the FrameNet (Baker et al., 1998) or the PropBank (Kingsbury et al., 2002) have been reported. Despite their reasonable performances on general tasks, these parsers do not work so well in specific domains. Also, where these general purpose parsers tend to provide case-frame structures, that include the standard core case roles (Agent, Patient, Instrument, etc.), dialogue oriented domains tend to require additional information about addressees, modality, speech acts, etc. Where general-purpose resources such as PropBank and Framenet provide invaluable training data for general case, it tends to be a problem to obtain enough training data in a specific dialogue oriented domain.

We in this paper propose and compare a number of approaches for building a statistically trained domain specific parser/NLU for a dialogue system. Our NLU is a part of Mission Rehearsal Exercise (MRE) project (Swartout et al., 2001). MRE is a large system that is being built to train experts, in which a trainee interacts with a Virtual Human using voice input. The purpose of our NLU is to convert the sentence strings produced by the speech recognizer into internal shallow semantic frames composed of slot-value pairs, for the dialogue module.

2 Parsing Methods

2.1 Voting Model

We use a simple conditional probability model $P(f \mid W)$ for parsing. The model represents the probability of producing slot-value pair f as an output given that we have seen a particular word or n-gram W as input. Our two-stage procedure for generating a frame for a given input sentence is: (1) Find a set of all slot-value that correspond with each word/ngram (2) Select the top portion of these candidates to form the final frame (Bhagat et al., 2005; Feng and Hovy, 2003).

2.2 Maximum Entropy

Our next approach is the Maximum Entropy (Berger et al., 1996) classification approach. Here, we cast our problem as a problem of ranking using a classifier where each slot-value pair in the training data is considered a class and feature set consists of the unigrams, bigrams and trigrams in the sentences (Bhagat et al., 2005).

2.3 Support Vector Machines

We use another commonly used classifier, Support Vector Machine (Burges, 1998), to perform the same task (Bhagat et al., 2005). Approach is similar to Section 2.2.

2.4 Language Model

As a fourth approach to the problem, we use the Statistical Language Model (Ponte and Croft, 1997). We estimate the language model for the slot-value pairs, then we construct our target interpretation as

Method	*Precison*	*Recall*	*F-score*
$Voting$	0.82	0.78	0.80
ME	0.77	0.80	0.78
SVM	0.79	0.72	0.75
$LM1$	0.80	0.84	0.82
$LM2$	0.82	0.84	0.83

Table 1: Performance of different systems on test data.

a set of the most likely slot-value pairs. We use unigram-based and trigram-based language models (Bhagat et al., 2005).

3 Experiments and Results

We train all our systems on a training set of 477 sentence-frame pairs. The systems are then tested on an unseen test set of 50 sentences. For the test sentences, the system generated frames are compared against the manually built gold standard frames, and Precision, Recall and F-scores are calculated for each frame.

Table 1 shows the average Precision, Recall and F-scores of the different systems for the 50 test sentences: Voting based (Voting), Maximum Entropy based (ME), Support Vector Machine based (SVM), Language Model based with unigrams (LM1) and Language Model based with trigrams (LM2). The F-scores show that the LM2 system performs the best though the system scores in general for all the systems are very close. To test the statistical significance of these scores, we conduct a two-tailed paired Student's t test (Manning and Schtze, 1999) on the F-scores of these systems for the 50 test cases. The test shows that there is no statistically significant difference in their performances.

4 Conclusions

This work illustrates that one can achieve fair success in building a statistical NLU engine for a restricted domain using relatively little training data and surprisingly using a rather simple voting model. The consistently good results obtained from all the systems on the task clearly indicate the feasibility of using using only word/ngram level features for parsing.

5 Future Work

Having successfully met the initial challenge of building a statistical NLU with limited training data, we have identified multiple avenues for further exploration. Firstly, we wish to build an hybrid system that will combine the strengths of all the systems to produce a much more accurate system. Secondly, we wish to see the effect that ASR output has on each of the systems. We want to test the robustness of systems against an increase in the ASR word error rate. Thirdly, we want to build a multi-clause utterance chunker to integrate with our systems. We have identified that complex multi-clause utterances have consistently hurt the system performances. To handle this, we are making efforts along with our colleagues in the speech community to build a real-time speech utterance-chunker. We are eager to discover any performance benefits. Finally, since we already have a corpus containing sentence and their corresponding semantic-frames, we want to explore the possibility of building a Statistical Generator using the same corpus that would take a frame as input and produce a sentence as output. This would take us a step closer to the idea of building a Reversible System that can act as a parser when used in one direction and as a generator when used in the other.

References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of COLING/ACL*, page 8690, Montreal, Canada.

Adam L. Berger, Stephen Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Rahul Bhagat, Anton Leuski, and Eduard Hovy. 2005. Statistical shallow semantic parsing despite little training data. Technical report available at `http://www.isi.edu/~rahul`.

Christopher J. C. Burges. 1998. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

Donghui Feng and Eduard Hovy. 2003. Semantics-oriented language understanding with automatic adaptability. In *Proceedings of Natural Language Processing and Knowledge Engineering*.

Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding semantic annotation to the penn treebank. In *Proceedings of HLT Conference*.

Christopher D. Manning and Hinrich Schtze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA.

Jay M. Ponte and W. Bruce Croft. 1997. Text segmentation by topic. In *Proceedings of the First European Conference on Research and Advanced Technology for Digital Libraries*, pages 120–129.

W. Swartout, R. Hill, J. Gratch, W. Johnson, C. Kyriakakis, C. LaBore, R. Lindheim, S. Marsella, D. Miraglia, B. Moore, J. Morie, J. Rickel, M. Thiebaux, L. Tuch, R. Whitney, and J. Douglas. 2001. Toward the holodeck: Integrating graphics, sound, character and story. In *Proceedings of Autonomous Agents*.