QASR: Question Answering Using Semantic Roles for Speech Interface

Svetlana Stenchikova *Dilek Hakkani-Tür* *Gokhan Tur*

Stony Brook ICSI SRI International

sveta@cs.sunysb.edu dilek@icsi.berkeley.edu gokhan@speech.sri.com

Abstract

In this paper, we evaluate a semantic role labeling approach to the extraction of answers in the open domain question answering task. We show that this technique especially improves the system performance when answers are communicated to the user by voice. Semantic role labeling identifies predicates and semantic argument phrases in a sentence. With this information we are able to analyze and extract structure from both questions and candidate sentences, which helps us identify more relevant and precise answers in a long list of candidate sentences. When searching for an answer to a question, we match the missing argument in the question to the semantic parses of the candidate answers. This technique significantly improves the accuracy of the question answering system and results in more concise and grammatical answers, which is essential for enabling voice interfaces to question answering systems. In this paper we apply our approach to factoid questions containing predicates; however, this technique can be also useful in answering more complex questions.

Index Terms: question answering, semantic roles

1. Introduction

Open domain question answering (QA) is the task of finding concise answers to natural language questions using the Web as a data set. For example, if one wants to find out *"Who first broke the sound barrier?"*, a question answering system simply returns the answer, *Yeager*. Question answering is different from information retrieval (search), which outputs pointers to documents with potential answers. One of the competitive advantages of question answering systems over search engines lies in their ability to provide a concise answer – particularly useful for less visually rich interfaces, such as speech-driven interfaces or hand-held devices. While users who have access to a computer may be able to efficiently find answers to their questions with a search engine by browsing through a large number of search results, a visually impaired user or a user without an access to a visual interface calling the system by phone may benefit from the additional processing of the data that a question answering system provides. However, in these cases the precision, conciseness, and grammaticality of the answer are important for comprehension.

In this work, we apply semantic role labeling to the QA task for factoid questions. Semantic role labeling aims to identify predicate/argument relations within a sentence.

To demonstrate the importance of predicate/argument extraction for the QA task consider the question *"Who created a comic strip Garfield?"* and a candidate sentence: *"Garfield is a popular comic strip created by Jim Davis featuring the cat Garfield ..."* Semantic Roles module identifies a predicate *created* and a direct object *a comic strip Garfield* in both the question and the candidate answer. In addition, it identifies *who* as an agent, which is the missing argument the system looks for in the question. In the candidate sentence *Jim Davis* is parsed as the agent. Without finding predicate/argument relations, we could extract the answer *Jim Davis* by creating an example-specific template. However, it is not feasible to create templates for each anticipated predicate/answer candidate pair because the number of predicates covered by open-domain question answering system is unlimited, as is the syntactic variation in candidate sentences. With the knowledge of the predicate/argument structure identified by the semantic role labeler, we can extract *Jim Davis* as the answer. We could approach the task by detecting named entities, however this approach would not be applicable to the questions where the answer is not a named entity. For example, an answer to *"What did Bell invent?"* is a non-named entity *the telephone* can not be extracted using named entity detection. Named entities approach would also be problematic for the candidate sentences that contain multiple matching named entities.

Our evaluation results show an improvement in answer accuracy compared to a baseline QA system. The results from a user study confirm our hypothesis that semantic role labeling approach produces more concise, more grammatical, and clearer answers.

In the following section we present current work on question answering and related applications of semantic role labeling. In Section 3, we describe QASR, a question answering system that uses semantic role labeling. In Section 4 we present automatic evaluations of this system. Section 5 focuses on the user evaluation. In Section 6 we describe our conclusions and ideas for future work.

2. Related Work

Many researchers currently work on Question Answering task participating in Text REtrieval Conference (TREC). TREC contains an annual competition on various text processing tasks, including a QA [1, 2, 3] task.

Narayanan and Harabagiu [4] use Framenet and PropBank on the AQUAINT corpus and show how sophisticated textual analysis (predicate/argument extraction) in combination with deep semantic representation and use of an inference model enhances QA systems. This work focuses on analysis of questions, decomposing a single complex query into a set of less complex queries using an ontology, morphological expansion, and an inference model. Our approach is different from that work in that we use semantic role labeling to find answer phrases as well as to analyze questions. Shapaqa's [5] grammatical relation extraction is similar to our approach. However, they use syntactic relations [6], which are an approximation to the semantic roles.

Katz and Lin [7] address semantic symmetry and ambiguous

Most of this research has been conducted when the authors were with AT&T Labs-Research

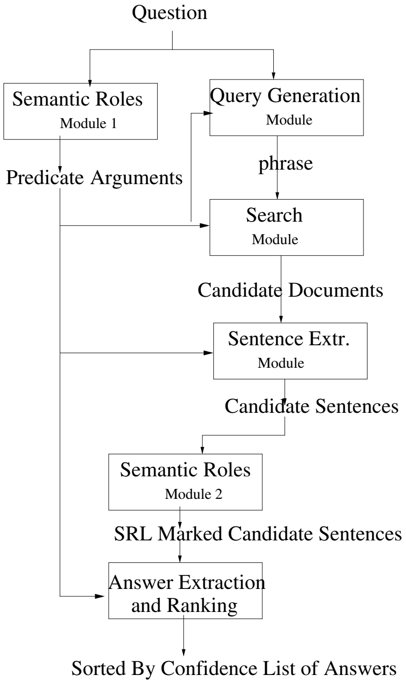

Figure 1: QASR System Architecture

modification by matching questions and answers on the level of syntactic relations. They note that ideally their system should use semantic relations.

Sun et. al. [8] successfully use semantic relations to match candidate answers. They define a similarity score and match all arguments in the question and candidate answer frame. In addition to confirming the advantages of semantic role labeling for answer extraction we also show user preference for the answers extracted using semantic role labeling technique.

In another recent effort using semantic relations for question answering Litkowsky [9] uses semantic relation triples that are automatically extracted from text. The relation triples correspond to the logical form and are incorporated it into the XML-based approach for Question Answering.

3. Approach

In our approach we use the Web as a data set, inspired by the performance of the systems described in [10], [11], [12]. QASR system adopts an architecture currently used by many QA systems where the main modules are: Query Generation, Search, Sentence Extraction, Answer Extraction and Ranking (see Figure 1). In addition to these components (also used on our baseline system), QASR uses a **Semantic Roles** module. The **Semantic Roles** module is applied first to the question and then to the candidate sentences, identifying predicate and arguments. *Assert* [13] program trained on PropBank [14] corpus is used for the **Semantic Roles** module. For the example in a sentence *"Nostradamus was born in 1503 in the south of France"*, Assert identifies *born* as a target predicate with three arguments: the object *"Nostradamus"*, a temporal argument *"in 1503"* and a locational argument *"in the south of France"*.

3.1. Search and Candidate Sentence Extraction

The **query generation** module creates a search engine query from the input natural language question and passes it to the **search** module for document retrieval. QASR system uses two methods for query generation: *exact phrase* and *conjunction of sub-phrases (inexact query)*. The exact phrase query is formed by removing the 'Wh' word, and converting the grammatical structure of the question to that of a statement. For example, given the question *When did Bell invent the telephone?*, the query *Bell invented the telephone* is generated. The *inexact query* is generated using the output of the **Semantic Roles** module applied to the question. For example, for the same question, *When did Bell invent the telephone?*, the **Semantic Roles** module identifies the predicate "invented", and the arguments "Bell" and "the telephone". Using this information, the inexact query *Bell* AND *invented* AND *the telephone* is generated by the **query generation** module. The method using exact phrase queries results in higher accuracy, but lower recall; whereas the one with the inexact query results in lower accuracy, but higher recall. Therefore, we use a cascaded approach to maximize the performance of the system; we first search for the answer using the exact phrase approach, and if no answer is returned, we switch to the inexact query approach (Figure 2).

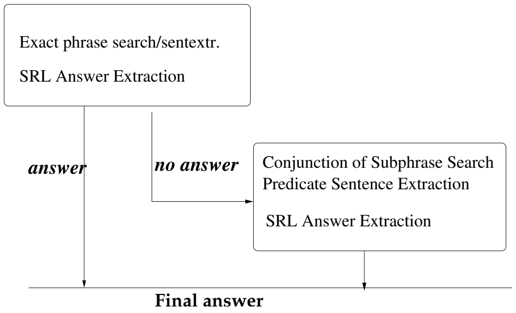
Exact phrase search/sentextr.
SRL Answer Extraction
answer
no answer
Conjunction of Subphrase Search
Predicate Sentence Extraction
SRL Answer Extraction
Final answer

Figure 2: 2-tier Cascaded Approach

Search is performed using the Google search engine. After the candidate documents are found, the **sentence extraction** module splits the returned HTML documents into sentences and extracts the sentences that contain phrases or sub-phrases of interest. All of the HTML candidate documents are sentence-split using a tool developed for the AnswerBus [15] system. We choose to use actual sentences from the returned documents in contrast to the snippets used by the AskMSR system [10]. Snippets are generally not complete sentences, which hurts the performance of a semantic role labeler. For *exact phrase* queries all sentences from extracted documents containing the searched phrase are chosen as candidate sentences. For *inexact queries*, sentences containing the searched predicate (identified by the *Semantic Roles* module) are selected as candidates.[1] We will further refer to these methods as *exact search/sentextr.* and *inexact search/sentextr*.

3.2. Answer Extraction

In the baseline system the answer is expected to appear in the candidate sentence on one side or the other of the search phrase, depending on the question type. For example, for the question *Who invented the silly paddy* the search phrase is *invented the silly paddy*. The answer is all words from the beginning of the sentence to the search phrase, because the question is a 'Who' question.[2] This simple baseline surprisingly achieves relatively good results in answer extraction by utilizing redundancy of the web.

In the SRL sentence extraction approach, the candidate sentences identified by the **Sentence Extraction** module are labeled

[1] We also considered using only sentences containing a search phrase, but this approach yield lower results.

[2] This method is only applicable to *exact* search/sentextr.

Search/SentExtr Type + Answer Extraction Method	accuracy	MRR
Exact+BASE	**19%**	**.24**
Exact+SRL	**24%**	**.29**
Inexact+SRL	16%	.23
CASCADE1: Ex+BASE;Ex+SRL	20%	.26
CASCADE2: Exact+SRL;Inex+SRL	**30%**	**.35**

Table 1: Evaluation of the QASR system performance.

by the **Semantic Roles** module. The labeled candidate sentences are then searched by the **Answer Extraction** module for the argument type of the 'Wh' word in the question. Because the performance of Assert on 'Wh' words is low, we use heuristics and question classification to determine the argument type of 'Wh' words. Heuristics are used to map 'Who', 'When' and 'Where' terms to semantic arguments; the question classifier described in [16] is used to map 'What' terms to semantic arguments. For the example question, *When did Bell invent the telephone?* the type of the searched argument is temporal. All candidate sentences contain a search predicate from the question, therefore the arguments of the predicate labeled by the semantic role labeler are the candidates in the answer extraction. Once the searched arguments are extracted from the candidate sentences, these answer candidates are ranked according to the frequencies of their occurrences in candidate sentences and provided to the user. Our method relies on the web redundancy by ranking more frequently occurring answers higher.

4. Experiments and Results

To evaluate QASR's performance, we use 190 questions containing a predicate other than "to be". [3] We decided to exclude the questions without a predicate because the semantic role labeler used in this task is based on PropBank, and the verb "to be" is not labeled as a predicate in PropBank. So we do not expect any improvement for these types of questions. The chosen question set comprises 38% or TREC-9 questions. Questions without a predicate are processed by the system using a baseline strategy. [4] For example, a sentence "Putin is the president of Russia" contains a predicate "is" and the semantic role program does not label the roles in this sentence. On the other hand, the sentence "Putin is visiting the US." contains a predicate "is visiting" and the arguments in this sentence are labeled by the semantic role labeler.

We evaluate QASR's overall performance using absolute accuracy and Mean Reciprocal Ranking (MRR). Absolute accuracy measures the percentage of questions where the first answer is correct. MRR assigns to each question a score equal to the inverse of the position of the first correct answer's index, or 0 if the correct answer is not in the top five answers supplied by the system. We used the evaluation script and correct answer patterns provided with the TREC-9 data to automatically evaluate QASR's performance. Table 1 presents the absolute accuracy and MRR values for QASR. We evaluate baseline answer extraction, SRL answer extraction on *exact* and *inexact* search/sentextr methods, and two versions of the cascaded approach.

System	Baseline	SRL on exact
Contains irrelevant information	26%	7%
Not grammatically correct	17%	2%
Average answer length (in words)	9.1	4.5

Table 2: Manual Evaluation of Correct Answers

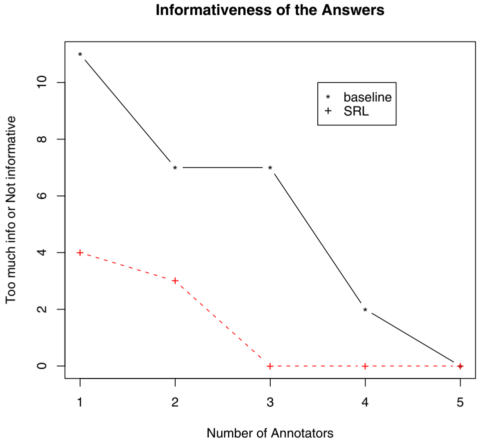

Figure 3: Number of answers marked as *too much information* or *not informative.*

The first cascaded approach uses the baseline system with *exact* search/sentextr and semantic role labeling with *exact* search/sentextr. The second cascaded approach combines SRL answer extraction on *exact* search/sentextr with SRL answer extraction on *inexact* search/sentextr resulting in the highest accuracy and MRR of .35 (increasing from .24 on baseline). [5] This improvement is due to the two factors: 1) the MRR of the SRL system on exact search candidates is higher than the MRR of the baseline system on the exact match candidates; 2) the SRL-based QA system has higher coverage because it also uses *inexact* search/sentextr.

Finally, we measured the quality of correct answers provided by QASR. Our measures are conciseness and grammaticality, which are manually labeled by an expert. The motivation behind this evaluation is that in a speech-enabled QA system irrelevant and ungrammatical answers may decrease the user's comprehension even if they are correct. Results of answer quality evaluation are presented in Table 2.

5. User Evaluation

In the experiments described above, an answer to a question is considered correct if it contains a correct answer as a substring. The evaluation does not penalize long and ungrammatical answers. In order to evaluate the quality of correct answers between the baseline and the SRL systems, we have also conducted a user study. Ten evaluators rank the answers from the baseline and the SRL systems, without knowing which system was used to generate the answer. We converted 18 correct answers to speech using AT&T text-to-speech engine. The answers used for user evaluation differ between the baseline and SRL systems, as examples in the Table 3. Each evaluator reads one question at a time, listens to the answer from one of the systems and rates the answer on the scale from 1 to 3 based on clarity of the answer's content (very clear, somewhat clear, or unclear), informativeness (too much information, suffi-

[3] Questions from TREC-9 QA track are used.

[4] Questions without predicates are not part of the evaluation; no change in performance is expected for these types of questions

[5] This is a statistically significant improvement according to Z-test, with 95% confidence interval.

Question	Baseline Answer	SRL Answer
Who painted Olympia?	Had Manet	Manet
Who wrote "An Ideal Husband"?	Oscar Wilde also	Oscar Wilde
When Babe Ruth was born?	in Baltimore , Maryland in 1895	in 1895
Who invented the radio?	As Marconi	Marconi
Where is Romania is located?	in south eastern Europe, bordering the Black Sea between Bulgaria and Ukraine	in south eastern Europe

Table 3: Example Answers to the Test Questions

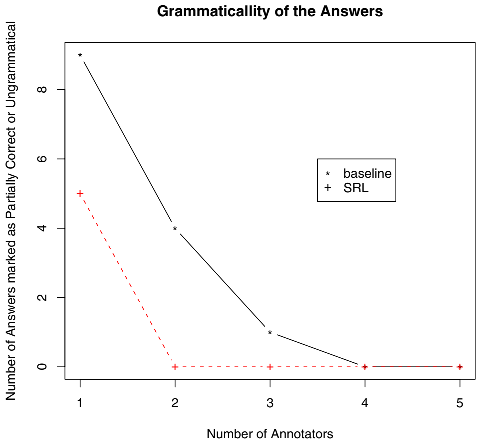

Figure 4: Number of answers marked as *partially correct* or *ungrammatical.*

cient information, or not informative), grammaticality (grammatically correct, partially correct, ungrammatical), and length (too long, sufficient, or too short) of the answer. Each evaluator marks all 18 questions, 9 from each of the systems. Answers from SRL and baseline systems alternate during the test, so that each baseline and each SRL answer is evaluated by 5 different annotators. variance between annotators. Figures 3 and 4 present results of the user ratings for the informativeness and grammaticality. Each of the graphs shows a number of answers marked unfavorable by at least 1, 2, 3, 4, or 5 annotators. We found that some annotators were too forgiving, marking every aspect of both systems positively. For all of the evaluated questions, the number of baseline answers marked unfavorably by at least one, two, or three annotators, is higher than the number of SRL answers marked unfavorably. These results confirm the user's preference for the answers produced by the SRL system.

6. Conclusions

We have presented an approach to automatic question answering that applies semantic role labeling to improve both query construction and answer extraction. Our approach produces significant performance improvements, and leads to more grammatical and concise answers, which is important for speech interfaces.

In the future, we plan to use a classifier-based approach to improve assignment of argument type to question terms. This will improve the accuracy of answer matching and increase the number of question types that QASR can handle.

7. References

[1] V. M. Vorhees, "Overview of trec 2000," Text Retrieval Conference (TREC 10)., 2001.

[2] V. M. Vorhees, "Overview of trec 2001," Text Retrieval Conference (TREC 2001)., 2002.

[3] V. M. Vorhees, "Overview of trec 2002," NIST Special Publication: SP 500-251 The Eleventh Text Retrieval Conference (TREC 2002)., 2003.

[4] S. Narayanan and S. Harabagiu, "Question answering based on semantic structures," in *Proceedings of the 20th International Conference on Computational Linguistics (COLING-2004)*, 2004.

[5] S. Buchholz and W. Daelemans, "Shapaqa: Shallow parsing for question answering on the www," in *Proceedings of RANLP*, 2001.

[6] W. Daelemans S. Buchholz, J. Veenstra, "Cascaded grammatical relation assignment," in *Proceedings of EMNLP/VLC-99*, 1999, pp. 239–246.

[7] B. Katz and J. Lin, "Selectively using relations to improve precision in question answering," in *Proceedings of the EACL 2003 Workshop on Natural Language Processing for Question Answering*, 2003.

[8] R. Sun, J. Jiang, H. Cui Y.F. Tan, T.-S. Chua, and M.-Y. Kan, "Using syntactic and semantic relation analysis in question answering," in *The Fourteenth Text REtrieval Conference*, 2005.

[9] K.C. Litkowski, "Exploring document content with xml to answer questions," in *The Fourteenth Text REtrieval Conference*, 2005.

[10] S. Dumais, M. Banko, E. Brill, J. Lin, and A. Ng, "Web question answering: Is more always better?," in *Text Retrieval Conference*, 2001.

[11] D. Ravichandran and E. Hovy, "Learning surface text patterns for a question answering system," in *Proceedings of the 40th ACL conference*, 2002.

[12] D. Radev, W. Fan, and H. Qi, "Probabilistic question answering on the web," in *Text REtrieval Conference*, 2002.

[13] Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James H. Martin, and Daniel Jurafsky, "Semantic role labeling using different syntactic views," in *Proceedings of the Association for Computational Linguistics 43rd annual meeting (ACL-2005)*, 2005.

[14] M. Palmer, D. Gildea, and P. Kingsbury, "The proposition bank: An annotated corpus of semantic roles," *Computational Linguistics*, vol. 31, no. 1, pp. 71–106, 2005.

[15] Zhiping Zheng, "Answerbus question answering system," in *Human Language Technology Conference (HLT 2002*, San Diego, CA., March 2002.

[16] X. Li and D. Roth, "Learning question classifiers," in *International Conference on Computational Linguistics , 2002*, 2002.