Speech Grammars for Textual Entailment Patterns in Multimodal Question Answering

Daniel Sonntag, Bogdan Sacaleanu
DFKI - German Research Center for AI
Stuhlsatzenhausweg 3, 66123 Saarbruecken
E-mail: sonntag@dfki.de, sacaleanu@dfki.de

Abstract

Over the last several years, speech-based question answering (QA) has become very popular in contrast to pure search engine based approaches on a desktop. Open-domain QA systems are now much more powerful and precise, and they can be used in speech applications. Speech-based question answering systems often rely on predefined grammars for speech understanding. In order to improve the coverage of such complex AI systems, we reused speech patterns used to generate textual entailment patterns. These can make multimodal question understanding more robust. We exemplify this in the context of a domain-specific dialogue scenario. As a result, written text input components (e.g., in a textual input field) can deal with more flexible input according to the derived textual entailment patterns. A multimodal QA dialogue spanning over several domains of interest, i.e., personal address book entries, questions about the music domain and politicians and other celebrities, demonstrates how the textual input mode can be used in a multimodal dialogue shell.

1. Introduction

Open-domain QA systems are now much more powerful and precise, and they can be used in speech applications (Sonntag 2010). In particular, expanding queries with ontology-based additional terms for information retrieval queries have produced good results in enhancing recall while only minimally decreasing precision. From the present-day perspective, the prominent query expansion techniques are among the more simple techniques (Fellbaum et al. 2008). To understand a greater number of queries and provide more answers in a multimodal speech-based dialogue system, more advanced approaches basically have to deal with three challenges:

- Robust question understanding (NLU) when using both speech and written text input. Here, a domain-specific, preferably unambiguous, interpretation of the question must be found. For our purposes we call this interpretation a concept query.
- Semantic (i.e., a RDF[1] or OWL[2] based) query interpretation in terms of domain-specific concepts from the model supported by the backend systems that provide the answers.
- The combination of robust question understanding and ontology-based answer retrieval so that we can speak of sophisticated approaches to question answering where speech and text input can be used to retrieve precise facts from ontological knowledge bases.

Both QA processes (robust question understanding and semantic query interpretation) may take background knowledge into account, which is explicitly stated in neither the text-based inquiry nor the fact base, in order to produce answers. In this text, we describe a semantic middleware (dialogue shell, Figure 1) where the aforementioned combination of robust question understanding and knowledge-based answer retrieval can be integrated into a common, unique QA architecture. We will focus on the robust question understanding task, i.e., the interpretation of textual questions that are derived from automatic speech recognition (ASR) grammars while using a speech-based multimodal dialogue shell and QA system.

2. Dialogue Shell

The dialogue shell has a three-tier architecture: an application layer where different inputs and outputs are combined to a multimodal user interface; a query model/semantic search layer which comprises of the query understanding step and the formulation of a concept query (see, e.g., Geurts et al. 2003); and the dynamic knowledge base layer which hosts the instance ontologies for the data lookup (we use YAGO as our knowledge base, see Suchanek et al., 2007). Technically, the generic dialogue framework follows a programming model which eases the interface to external third-party components (e.g., the automatic speech recogniser, natural language understanding (NLU), or synthesis component). In the context of semantic search, however, the ontology-based platform (ODP, see www.semvox.de) uses ontology concepts in a model-based design. This means that all internal and external module messages are based on ontology structures. The dialogue system contains an ontology-based rule engine for processing *dialogue grammars* and an external service connector (the formal query (NLU result) is *analysed* and mapped to one or more services that can answer (parts of) the query), cf. semantic query interpretation.

[1] http://www.w3.org/RDF/

[2] http://www.w3.org/TR/owl-features/

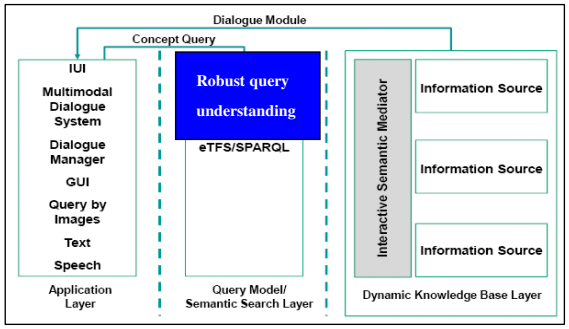

Figure 1. Dialogue Shell Architecture

This step typically involves several sub-steps, including the decomposition of the query into smaller parts. The query is sent to several backend services (in our case, a semantic mediator provides the access to the YAGO database). Results (using the respective backend's knowledge representation) are mapped onto the custom knowledge representation (Allemang und Hendler 2008) used in the dialogue shell and in the NLU component (we use typed feature structures). Our idea is that this NLU grammar for speech input can be reused to build more robust multimodal text-based question understanding by automatically generating textual entailment patterns.

3. Robust Multimodal Question Understanding

Multimodal interfaces cover a large spectrum of input modalities from the user: speech, written text, gestures, body language, eye or lip movements, etc. While their integration and synchronisation is a major goal in developing intelligent user interfaces, the input modalities are usually interpreted according to *separate* models and aligned to a shared model. Though practical enough to generate acceptable results, this method often uses a shared model that is more general than the individual models taken apart. It also becomes more course-grained as the number of modalities increases.

To improve this methodology, we consider the two input modalities at this stage, namely speech and written text, and present a method of interpretation based on a *common* model built on the grammar for speech inputs. This method is advantageous because changes in the model are automatically propagated to the modalities supporting it. As a result, integration of different modes of interaction becomes therefore more smooth. This, in turn, means that written text input components (e.g., in a textual input field) can deal with more flexible input according to derived textual entailment patterns.

The ODP grammar framework provides one location to specify the speech grammar and to map the question interpretation onto ODP-specific ontology concepts. This makes the grammar very powerful, but also difficult to develop and maintain. In order to support the dialogue engineer of the system in developing such a mixed grammar, a proactive editing environment for the ASR and NLU grammar has been developed as an Eclipse plug-in (Figure 2, also see Sonntag et al. 2009).

Figure 2. ASR and NLU Monitor

However, the actual automatic speech recognition system encapsulates the interpretation of *raw text transcriptions* (also known as *speech hypotheses*) of the audio speech input, making its reusability difficult for written text. In order to avoid the dependency of the dialogue system on proprietary subcomponents of the employed speech recognisers (we use NUANCE as a connected third-party component), a standalone natural language understanding (NLU) component has been considered to semantically interpret textual input. Since both the ASR and the language understanding components use a grammar that specifies the words and sequences of words in order to define the input language that they can accept, it is to be assumed that these grammars have much in common. Hence, given the existence of a speech grammar, the development of the text grammar should leverage the available grammar structures to a large extent and not duplicate them. A tool of grammar conversion between these two predefined annotations has been made available for this purpose. This tool accounts not only for the utterances being expected but also for their semantic interpretation as defined in the original speech grammar.

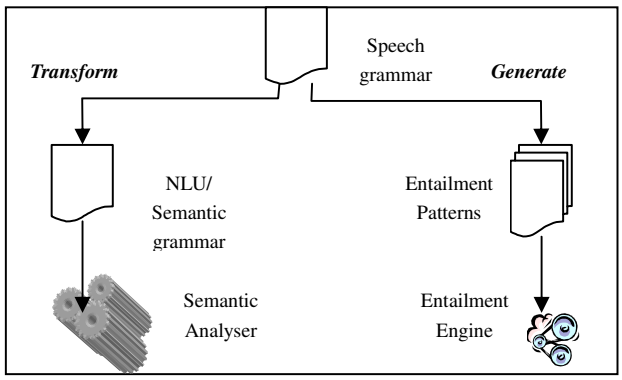

Figure 3. Leveraging Existing Speech Grammar Knowledge

The first traditional approach (Figure 3, left), robust question understanding (NLU) of written text, uses a domain-specific analyser built upon context free speech grammars that have rules associated with semantic attachments representing the intended meaning of an utterance. Though very widely used, this method requires additional handwritten grammars that can be expensive and can only be created slowly since they have to be more

comprehensive than speech grammars. In addition, due to the very nature of human language, such traditional grammars invariably fail to achieve full coverage of unseen data.

Where is PERSON_NAME born?
- What is the birthplace of PERSON_NAME?
- Where is PERSON_NAME original from?

What is PERSON_NAME's nationality?
- What nationality is PERSON_NAME?
- Which nation does PERSON_NAME belong to?
- What is the citizenship status of PERSON_NAME?

What is PERSON_NAME's profession?
- Which profession has PERSON_NAME?
- What is PERSON_NAME's job?
- What is PERSON_NAME doing for a living?

Figure 4. Entailment patterns and possible hypothesis

By acknowledging the impossibility of stating all the surface forms by which a concept can be expressed *a priori*, the second approach (Figure 3, right) overcomes some of the issues raised by the first one by avoiding the manual development of rules for any new written query that has to be accounted for. It uses *textual entailment* in order to associate syntactically different utterances conveying the same meaning. This method verifies the entailed semantics between any new utterance and a set of domain patterns already having a semantic interpretation. The speech grammar is the starting point for the generation of an initial set of entailment patterns, i.e., patterns that cover the available knowledge for the given special domain. The patterns are automatically generated sentences from the language covered by the speech grammar (Figure 4).
The process of verifying whether a question posed by the user is entailed in one of the available patterns is based on a simple method of association-based word alignment. Association-based word alignment generally undergoes three steps:

- lexical segmentation, when boundaries of lexical items are identified;
- correspondence, when possible similarities are suggested in line with some correspondence measures;
- alignment, when the most likely semantically similar word is chosen.

In a first step, we tokenize the question and its translations into a list of words. Next, we employ several alignment techniques based on string similarity measures, lexical semantic resources, and part-of-speech (POS) tags. They all act like filters on a full alignment, where each source word is associated with all target words. The following filters have been considered in the development:

- Part-of-speech (based on TnT - Brants, 2000)
- Lexical semantic resources (WordNet - Miller, G. A. et al., 1993**;** Roget Thesaurus)
- String similarity for direct alignments and misspellings
  - Dice coefficient
  - Lowest common subsequence ratio

The following fabricated example describes how the alignment component works:

Question: *What is the birthplace of Angela Merkel?*
Pattern: *Where is Angela Merckel born?*

To begin with, full alignments for every source word are generated that are the target of a filtering process as described below. Every alignment has a Boolean value of *true* if already aligned and a weight associated with it:

what:	{[where, is, Angela, Merckel, born]}	false
is:	{[where, is, Angela, Merckel, born]}	false
the:	{[where, is, Angela, Merckel, born]}	false
birthplace:	{[where, is, Angela, Merckel, born]}	false
of:	{[where, is, Angela, Merckel, born]}	false
Angela:	{[where, is, Angela, Merckel, born]}	false
Merkel:	{[where, is, Angela, Merckel, born]}	false

We first use the POS filter in order to exclude unlikely alignments based on the part-of-speech tags of the words being considered. Beside one-to-one alignment of words with similar POS tags we allow for the following additional mappings

- noun to adjective (i.e., *birthplace* vs. *original*)
- verb to noun (i.e., *born* vs. *birthplace*)

in order to account for conceptually related words:

what:	{[where]}	true
is:	{[is, born]}	false
the:	{[]}	false
birthplace:	{[is, born]}	false
of:	{[]}	false
Angela:	{[Angela, Merckel]}	false
Merkel:	{[Angela, Merckel]}	false

Next, the lexical semantic resources based filter looks up words in a thesaurus and matches them against those in the actual alignment. This filter is responsible for aligning synonyms (WordNet) and conceptually related words (Roget Thesaurus):

what:	{[where]}	true
is:	{[is]}	true
the:	{[]}	false
birthplace:	{[born]}	true
of:	{[]}	false
Angela:	{[Angela, Merckel]}	false
Merkel:	{[Angela, Merckel]}	false

Finally, the alignment methods based on string similarity measures are used to best discover direct alignments and *misspellings* (e.g., "Merkel" vs. "Merckel"):

what:	{[where]}	true
is:	{[is]}	true
the:	{[]}	false
birthplace:	{[born]}	true
of:	{[]}	false
Angela:	{[Angela]}	true
Merkel:	{[Merckel]}	true

For the case of a full alignment between the question and a pattern, we can consider both utterances semantic similar or entailed, though this is rather an exception as the majority of alignments are partial. In order to distinguish between several possible partial alignments and also keep the assumption of entailment, we have developed a system of weights so that alignments of semantics-bearing words (i.e., nouns, verbs and adjectives) are better scored than function words. This way we avoid the situation of considering two utterances entailed based only on a large overlap of words with little lexical meaning.

4. Multimodal QA Dialogue

Our resulting multimodal QA dialogue spans over several domains of interest, i.e., personal address book entries, questions about the music domain, and politicians or other celebrities. Questions about personal address book entries and questions about the music domain are answered with the help of a direct lexico-semantic mapping to ontology-based knowledge sources according to the specified speech grammar. However, the questions about politicians and celebrities can be answered with the help of the textual entailment patterns when text input is used. The following dialogue illustrates how we combine the different question processing possibilities into a coherent dialogue.

(1) U: "Open my personal address book. What do you know about Claudia?"

(2) S: "There's an entry: Claudia Schwartz. The personal details are shown below. She lives in Berlin." + Google Map Display of street coordinates.

(3) U: "Which is Claudia's favorite kind of music? Do you know the bands she likes most?"

(4) S: "Nelly Furtado" + Displays videos obtained from YouTube. (Rest API)

(5) U: "How did experts rate her last album?"

(6) S: Shows an expert review according to the BBC Linked Data Set.

(7) U: "Show me other news."

(8) S: Opens a browser + **Text field** and a new agency Internet page (featuring Angela Merkel)

(9) U **writes:** "Where was Angela Merkel born? / In which town was Angela Merkel born?" etc.

(10) S: "She was born in Hamburg."

(11) U **speaks again:** "And Barack Obama?"

(12) S: "He was born in Honolulu."

(13) U: "Show me Angela Merkel's career."

5. Conclusion

We described a multimodal dialogue shell for QA and focussed on the robust multimodal question understanding task. In order to avoid the dependency of the dialogue system on proprietary subcomponents of the employed speech recognisers, a standalone natural language understanding (NLU) component has been developed to semantically interpret textual input. The textual interpretation is based on automatically generated textual entailment patterns. As a result, we can deal with written text input and different surface forms more flexibly according to the derived entailment patterns. The multimodal QA dialogue demonstrated how the textual input mode can be used in the multimodal dialogue shell.

6. Acknowledgements

This work has been supported by the German Federal Ministry of Economics and Technology (01MQ07016). Thanks go out to Robert Nesselrath, Yajing Zang, Günter Neumann, Matthieu Deru, Simon Bergweiler, Gerhard Sonnenberg, Norbert Reithinger, Gerd Herzog, Alassane Ndiaye, Tilman Becker, Norbert Pfleger, Alexander Pfalzgraf, Jan Schehl, Jochen Steigner, and Colette Weihrauch for the implementation and evaluation of the dialogue infrastructure. The responsibility for this publication lies with the authors.

7. References

Allemang und Hendler (2008). Dean Allemang and James A. Hendler: Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL, Morgan Kauffman, 2008.

Brants (2000). Thorsten Brants: TnT - A Statistical Part-of-Speech Tagger. *6th Applied Natural Language Processing (ANLP '00), April 29 - May 4*, Pages 224-231, Association for Computational Linguistics, Seattle, USA, 2000.

Fellbaum et al. (2008). Christiane Fellbaum, Peter Clark and Jerry Hobbs: Towards improved text understanding with WordNet. In: Angelika Storrer, Alexander Geyken, Alexander, Alexander Siebert and Kay-Michael Würzner (eds.): Text Resources and Lexical Knowledge – Selected Papers from the 9th Conference on Natural Language Processing KONVENS 2008, Mouton de Gruyter, Berlin, New York, pp. 81-90, 2008.

Geurts et al. (2003). Joost Geurts, Stefano Bocconi, Jacco Van Ossenbruggen and Lynda Hardman: Towards Ontology-driven Discourse: From Semantic Graphs to Multimedia Presentations. In: Proceedings of the Second International Semantic Web Conference (ISWC), pp. 597–612, 2003.

Miller, G. A. et al. (1993). G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller: Five Papers on WordNet. *Technical Report, Cognitive Science Laboratory*, Princeton University, 1993.

Sonntag (2010). Daniel Sonntag: Ontologies and Adaptivity in Dialogue for Question Answering. AKA/IOS Press, 2010.

Sonntag et al. (2009). Daniel Sonntag, Gerhard Sonnenberg, Robert Neßelrath and Gerd Herzog: Supporting a Rapid Dialogue Engineering Process. In: Proceedings of the 1st International Workshop on Spoken Dialogue Systems (IWSDS), Kloster Irsee, 2009.

Wang und Neumann (2009). Rui Wang and Günter Neumann: An Accuracy-Oriented Divide-and-Conquer Strategy for Recognizing Textual Entailment. In: Proceedings of the 1st Text Analysis Conference, Gaithersburg, Maryland, National Institute of Standards and Technology (NIST), 2/2009.

Suchanek et al. (2007). Fabian Suchanek, Gjergji Kasneci and Gerhard Weikum: Yago: A Core of Semantic Knowledge. In 16th International World Wide Web conference (WWW). New York, NY, USA: ACM Press, 2007.