A Multi-Agent Approach to Question Answering

Cássia Trojahn dos Santos[1], Paulo Quaresma[1], Irene Rodrigues[1] and Renata Vieira[2]

[1]Departamento de Informática, Universidade de Évora, Portugal
[2]Pós-Graduação em Computação Aplicada, Universidade do Vale do Rio dos Sinos, Brazil
{cassia,pq,ipr}@di.uevora.pt, renata@unisinos.br

Abstract. In this paper we present a multi-agent approach to question answering for the Portuguese language. Our proposal is composed by three modules: (1) document and query processing; (2) ontology construction; and (3) answer generation. Each module is composed by multiple cooperative agents which adopt distinct strategies to generate its outputs and cooperate to create a global result. This approach allows the use of different strategies and the reduction of errors introduced by individual methods. The cooperation among the agents aims to reach better solutions in each step of the processing.

1 Introduction

Question answering systems aim to retrieve "answers" to questions rather than full documents or even best-matching passages as most information retrieval systems currently do [6].

Traditional question answering systems employ a single pipeline architecture, consisting roughly of three components: question analysis, search, and answer selection [3]. Typically, each system employs one specific approach in such components. The systems are dependent on the answer search strategy that implement and they can not be able to find a correct answer. This way, there might be others strategies that would be successful at finding an answer.

Recently, the multi-agent approach has received attention from the question answering community. Following this approach, several tasks of query answering process can be distributed between the agents, in order to reach a more easily extensible system. Moreover, the use of several agents encapsulating different strategies in each task can lead the systems to obtain better solutions.

In this paper we present a multi-agent approach to question answering. Our proposal is to extend the architecture of the question answering system for the Portuguese language described in [14]. Such architecture follows a symbolic approach, based on linguistic processing components. This processing includes syntactical analysis of sentences, semantical analysis using discourse representation theory, and semantic/pragmatic interpretation using ontologies and logical inference.

Our approach is motivated by the success of ensemble methods in machine learning, which have shown great success in improving predictive accuracy. These systems typically employ multiple classifiers to first solve the same problem, then combine the results to provide a final ensemble answer [5]. According to [18] different machine learning techniques applied to the same data set hardly generate the same results – an algorithm A can construct an accurate model for concept X and a weak description for concept Y, while the algorithm B constructs an accurate model for concept Y and a weak model for concept X. Moreover, no algorithm can be the best choice in all possible domains. Each algorithm contains an explicit or implicit bias that leads it to prefer certain generalizations over others [4]. Then, the combination of different learning algorithms can lead to models more accurate. Although determining exactly how to best combine individual results is still an active area of research, a variety of ensemble methods have already been shown to improve predictive performance in various areas of natural language processing [16].

Our multi-agent proposal is composed by three modules: (1) document and query processing; (2) ontology construction; and (3) answer generation. Each module is composed by multiple cooperative agents which adopt distinct strategies to generate its outputs and cooperate to create a global result. This approach allows the use of different strategies and the reduction of errors introduced by the individual ones. Moreover, the cooperation among the agents aims to reach better solutions in each step of the processing.

This paper is structured as follows. Section 2 introduces our multi-agent approach, detailing each agent of the architecture. Section 3 describes some related works regarding multi-agent systems and question answering. Finally, Section 4 comments the final remarks and the future work.

2 The Multi-Agent Approach to QA

A QA system should be able to answer queries in natural language, based on information conveyed by a collection of documents. The answer to a specific question is a related set of words and the identification of the document and sentence, which was used as the source of information.

Figure 1 shows the proposed multi-agent architecture. The first module ("document and query processing") is responsible for the document and query processing. The agents of this module act in two steps. First, they extract information from the documents and create a knowledge base. Second, the agents process the query and create the semantic structure of the sentences.

Two types of agents compose the first module: *syntactical analysis agent* and *semantic and pragmatic analysis agent*. The *syntactical analysis agent* is responsible for processing the document or query sentences, generating the syntactic structure of sentences (i.e. syntactical tree, represented in Prolog).

The *semantic and pragmatic analysis agent* transforms the output of the *syntactical analysis agent* in another collection, where each document or query has a semantic representation (i.e, discourse representation structure, DRS [12]).

The ontology is taken into account in this processing. In the phase of document processing, a knowledge base containing instances of the ontology is built as a result.

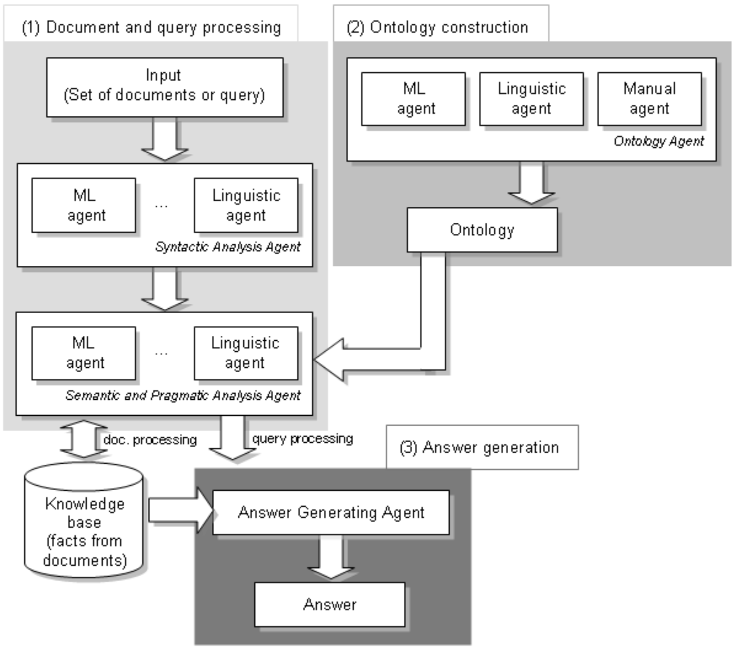

Fig. 1. Proposed multi-agent architecture.

These agents adopt distinct strategies (i.e, linguistic and learning methods) to generate its results and cooperate to create a final result. This approach is also adopted in the second module of the proposed architecture.

In the second module ("ontology construction") the domain ontology is constructed. Following the same approach adopted in the first module, multiple agents process the documents and cooperate to create an ontology that represents the domain. Each agent is responsible for applying a specific strategy. After, the individual results are combined in a global domain ontology, through cooperation among the agents.

Finally, in the third module ("answer generation"), the answer is generated: a set of words and the identification of the document and sentence where the answer was found. This module is composed by the *answer generating agent* that is responsible for interpreting the query in the knowledge base trough the unification of the discourse entities of the query with documents discourse entities.

In the next sections, each agent is described in more detail.

2.1 Syntactical Analysis Agent

The syntactical agents are responsible for the parsing, adopting machine learning and linguistic methods. According to [8], the problem is to select the most plausible syntactic analysis given the great number of analysis a typical parser with a sophisticated grammar may return. For this reason, we are interested in the use of several approaches for syntactical analysis.

The learning agents model the syntactic structure of sentences using machine learning methods which are capable of learning the structure given correctly annotated documents. Specifically, our agents adopt symbolic (i.e., decision trees [15]) and connexionist (i.e., artificial neural networks [9]) techniques.

Applying these techniques, the agent uses syntactically annotated examples to generate a parser represented by a model induced in the learning phase. This model can be, for instance, in the form of a decision tree from which rules can be extracted. The model can be used to parse previously unseen sentences.

Moreover, the learning agents are based on a shallow parsing approach. Following this approach, rather than produce a detailed syntactic analysis of each sentence, key parts of the syntactic structure are identified or extracted [13]. Such processing include identifying the major phrases in a sentence or extracting the subject, main verb and object from a sentence. According to [8], a full parse often provides more information than needed and sometimes less. In Question Answering, it is interesting the information about specific syntactic-semantic relations such as agent, object, location, time, rather than elaborate configurational syntactic analysis.

In other hand, the linguistic agents use parsers which give morpho-syntactical information of the sentences. For instance, the Palavras [1] parser has a good coverage of the Portuguese language and it has been used by our linguistic agents. As an example of the partial output of this agent, consider the following sentence:

- "A filha de Elvis Presley, Lisa Marie, confirmou seu casamento com o cantor Michael Jackson" (1)

- The daughter of Elvis Presley, Lisa Marie, has confirmed her marriage with the singer Michael Jackson (2)

The syntactical structure of this sentence is presented in Figure 2.

The output of these agents, containing the syntactical structure of the sentences is transformed into a equivalent Prolog representation. Next, the results are combined, generating the final syntactical structure. The advantage of the development of several strategies is the possibility to use these in error detection. According to [13], parsers often make different types of errors and thereby can complement each other.

```
STA:fcl
=SUBJ:np
==>N:art('o' F S <artd>)        A
==H:n('filha' F S <Hfam>)       filha
==N<:pp
===H:prp('de')  de
===P<:np
====H:prop('Elvis_Presey' M/F S)        Elvis_Presey
====,
====APP:prop('Lisa_Marie' F S)  Lisa_Marie
====,
=P:v-fin('confirmar' PS 3S IND) confirmou
=ACC:np
==>N:pron-det('seu' M S <si> <poss 3S>) seu
==H:n('casamento' M S <occ> <state-h>)  casamento
=ADVL:pp
==H:prp('com')  com
==P<:np
===>N:art('o' M S <artd>)       o
===H:n('cantor' M S <Hprof>)    cantor
===N<:prop('Michael_Jackson' M S)       Michael_Jackson
```

Fig. 2. Output of the Palavras parser.

2.2 Semantic and Pragmatic Analysis Agent

Semantic analysis associates a sentence with terms in the ontology. The semantic and pragmatic analysis agents rewrite the syntactical structure in a semantic representation, taking into account the rules obtained from the ontology. For generating these semantic structures, they adopt strategies based on machine learning and linguistic techniques. After generating the individual structures, the agents cooperate to merge their results, creating the final semantic structure. In the document processing phase, the final structure is used to generate a knowledge base containing instances of the ontology. In the query processing, this structure represent the semantic/pragmatic interpretation of the query.

The agents that adopt machine learning techniques try to extract the semantic/pragmatic information (instances of the ontology) taking into account a model induced from learning examples containing the syntactic structures of the sentences. The model is then applied to unseen syntactical structures and the informational sentences corresponding to the rules are extracted.

The linguist agents use a DRT (from discourse representation structure – DRS) to convert the syntactic structure into a semantic structure. Next, they do the pragmatic interpretation using the ontology. As an example of output of this agent, consider the sentence (1). First, the syntactic tree (Figure 2) is rewritten into a DRS. At present, the linguistic agent deals with factual sentences, i.e, sentences with existential quantification over the discourse entities. So, the discourse structures are sets of referents, existentially quantified variables, and sets of conditions, first order predicates.

The output of this first processing is shown in Figure 3.

```
X Y Z:
filha(X)
rel(de, X, Y)
nome(Y, 'Elvis_Presley')
nome(X, 'Lisa_Marie')
confirmar(X)
casamento(X)
rel(com, X, Z)
cantor(Z)
nome(Z, 'Michael_Jackson')
```

Fig. 3. Output of the semantic analysis by the linguistic agent.

Second, the agent takes as input the semantic information and interprets it using the rules obtained from the ontology and the information in the database. In order to obtain a good interpretation, the linguistic agent searches for the best explanation that supports the sentence logical form. This strategy for pragmatic interpretation was initially proposed by [10]. The knowledge base for the pragmatic interpretation is built from the ontology description. The inference in this base uses abduction and finite domain constraint solvers. As a result of this processing, a new DRS is generated (Figure 4).

```
X Y Z:
nome(X, 'Lisa_Marie')
filha(X, Y)
nome(Y, ' Elvis_Presley')
casar(X, Z)
nome(Z, 'Michael_Jackson')
cantor(Z)
```

Fig. 4. Output of the pragmatic analysis by the linguistic agent.

2.3 Ontology Agent

An ontology is used to model domain knowledge, defining classes and relation between these classes. In our architecture, the ontologies are generated from multiple agents encapsulating strategies based on machine learning, NLP techniques, and manual methods. It is made by merging the results of each agent, generating a global ontology. The objective is to explore the characteristics of different strategies.

The learning ontology agents are based in the automatic ontology building approach, via machine learning techniques. These agents receive as input annotated documents and learn hierarchical relations between concepts.

The strategies we pretend to use in our NLP agents are based on the approach proposed by [17] which aims to generate ontologies automatically from the document collection. Basically, this approach has the following steps:

- definition of an initial top-level ontology;
- identification of concepts referred in the documents and extraction of its properties;
- identification of relations between the identified concepts;
- creation of an ontology using the identified concepts and relations;
- merge of the created ontology with the initial ontology.

Finally, the manual agents receive as input an ontology manually created and they are responsible for converting the ontology in a standard format.

These three types of agents use OWL (Ontology Web Language) to represent theirs outputs. The global ontology is generated by cooperation among the learning, NLP and manual agents. It is used by the semantic and pragmatic analysis agents in the extraction of facts from the documents and generation of instances of the ontology, which are inserted in the knowledge base (document processing). In the query processing, the ontology is used to interpret the syntactic structure and convert it into a semantic structure. We point out that the global ontology is generated for a given domain/collection and, then, used in the processing of all questions related to this domain.

2.4 Answer Generation Agent

The answer generation agent processes the query in two steps:

- identification of the database referent that unifies with the referent of the interrogative pronoun in the question;
- retrieval of the referent proprieties and generation of the answer.

In order to illustrate this process, suppose the following question (3).

- "Com quem casou Michael Jackson?" (3)

- Who Michael Jackson married to? (4)

This question is represented by the following syntactic structure (Figure 5), when it is analyzed by a linguistic agent. The final output of this question, after semantic and pragmatic analysis using the knowledge ontology, is represented in Figure 6.

```
QUE:fcl
=PIV:pp
==H:prp('com')  Com
==P<:pron-indp('quem' M/F S/P <interr>) quem
=ACC    :pron-pers('se' M 3S/P ACC)     se
=P:v-fin('casar' PS 3S IND)     casou
=SUBJ:prop('Michael_Jackson' M S)       Michael_Jackson
=?
```

Fig. 5. Syntactic structure of the question.

```
Y X : nome(X, 'Michael_Jackson'), casar(X, Y)
```

Fig. 6. DRS of the question after semantic and pragmatic analysis.

In order to perform the first step of the answer generation, the agent keeps the referent variables of the question and try to prove the conditions of the semantic structures in the knowledge base. If the conditions can be satisfied in the knowledge, the structures are unified with the identifiers of the individuals.

The next step is to retrieve the words that constitute the answer. In this phase, the agent should retrieve the conditions about the identified referent A and choose which ones better characterize the entity. Our first option is to choose a condition with the predicate *nome*(nome(A,Nome)).

In order to choose the best answer to a question, the agent takes into account the syntactical category of the words that may appear in the answer and it tries to avoid answers with words that appear in the question.

3 Related Work

A multi-agent system, called MASAQ, for answering users' questions based on the knowledge base is presented by [7]. The system consists of four major components: (1) a natural language user interface; (2) an encyclopedic knowledge base covering 21 domains; (3) a communication protocol based XML and KQML; and (4) an executable agent specification language for developing domain-specific multi-agent systems for answering or reasoning about users' questions. Experiments have demonstrated that MASAQ can reason all the knowledge of 21 domains efficiently. An architecture composed by multiple answering agents is proposed by [2]. The agents adopt different processing strategies and consult different knowledge sources in identifying answers to given questions. They employ resolution mechanisms to combine the results produced by the individual answering agents. Experimental results show significant performance improvement over their single-strategy, single-source baselines (35.0% relative improvement in the number of correct answers and 32.8% improvement in average precision).

The proposal of [16] is to evaluate empirically whether combining the outputs of several systems can improve over the performance of any individual system. Seven different natural language algorithms were incorporate in the system (e.g.

only a few systems included any semantic processing, and even fewer included coreference). The ensemble experiments showed the utility of majority voting as a method for combining the output of such systems. A similar ensemble approach is presented by [11]. The system combines six radically different QA strategies in the TREC setting. They investigate the impact of various weighted voting techniques (including question type dependent).

Our approach approximates of the work proposed by [7] in sense of the use of agents in the several steps of query and answer processing. However, these authors do not use different strategies (i.e. multiple agents) in each step of the processing as we propose in our work. The interesting in their work is the agent's reasoning about users' questions. We also intend to explore this aspect.

Similarly to [2], we adopt several agents encapsuling different processing strategies. However, we are interested, specifically, in agents using different machine learning and linguist techniques and propose to merge the individual results by cooperation among the agents. Considering the first aspect, our proposal approximates of the work of [16] and [11], which intend to combine individual results, but without the use of cooperation mechanisms.

4 Final Remarks and Future Works

In this paper we presented a multi-agent approach to question answering. Our proposal aims to extend the architecture of the question answering system for the Portuguese language described in [14]. We proposed to use multiple cooperative agents which adopt distinct strategies to generate its outputs. This approach allows to use different approaches and reduce the errors introduced by the individual strategies. The cooperation among the agents aims to reach the better solutions in each step of the processing.

At present, the linguistic agents which are responsible for the syntactical and semantic/pragmatic analysis are implemented. The ontology has been manually created.

As future work, several tasks can be cited: (1) implement the learning agents, defining the techniques which will be used in each step of processing; (2) define and implement the methods to learning ontologies; (3) define and implement the mechanisms of cooperation among the agents; (4) explore the automatically creation of ontologies; and (5) test our system in different domains.

References

1. E. Bick. *The Parsing System Palavras.* Aarhus University Press, 2000.
2. J. Chu-Carroll, K. Czuba, J. M. Prager, and A. Ittycheriah. In question answering, two heads are better than one. In *Proceedings of HLT-NAACL 2003*, 2003.
3. J. Chu-Carroll, J. M. Prager, C. A. Welty, K. Czuba, and D. A. Ferrucci. A multi-strategy and multi-source approach to question answering. In *TREC*, 2002.
4. T. Dietterich. Limitations on inductive learning. In *Proceedings of the Sixth International Workshop on Machine Learning*, pages 124–128, 1989.

5. T. Dietterich. Machine learning research: Four current directions. 4(18), 1997.
6. S. Dumais, M. Banko, E. Brill, J. Lin, and A. Ng. Web question answering: Is more always better? In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 291–298, 2002.
7. Q. Feng, C. Cao, Y. Sui, Y. Zheng, and Q. Qin. Masaq: A multi-agent system for answering questions based on an encyclopedic knowledge base. In *Declarative Agent Languages and Technologies (DALT)*, 2005.
8. J. Hammerton, M. Osborne, S. Armstrong, and W. Daelemans. Introduction to special issue on machine learning approaches to shallow parsing. In *Journal of Machine Learning Research*, pages 551–558, 2002.
9. S. Haykin. *Redes Neurais Artificiais*. Bookman, 2001.
10. J. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. Technical Report 499, Ravenswood, November 1990.
11. V. Jijkoun and M. de Rijke. Answer selection in a multi-stream open domain question answering system. In *Proceedings of European Conference on Information Retrieval*, pages 99–111, 2004.
12. H. Kamp and U. Reyle. From discourse to logic:an introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory. 1993.
13. B. Megyesi. *Data-Driven Syntactic Analysis: Methods and Applications for Swedish*. PhD thesis, University of Kungl Tekniska Hogskolan, 2002.
14. P. Quaresma and I. Rodrigues. A logic programming based approach to the qa@clef05 track. In *Cross Language Evaluation Forum: Working Notes for the CLEF 2005 Workshop*, 2005.
15. J. R. Quinlan. C4.5: Programs for machine learning. 1993.
16. M. Rotaru and D. Litman. Improving question answering for reading comprehension tests by combining multiple systems. In *Proceedings of the AAAI 2005 Workshop on Question Answering in Restricted Domains*, 2005.
17. J. Saias and P. Quaresma. A methodology to create legal ontologies in a logic programming information retrieval system. In *Law and the Semantic Web*, pages 185–200, 2003.
18. H. Viktor and H. Arndt. Combining data mining and human expertise for making decisions, sense and policies. 4(2):33–56, 2000.