PGR: Portuguese Attorney General's Office Decisions on the Web

Paulo Quaresma and Irene Pimenta Rodrigues

Departamento de Informática,
Universidade de Évora,
7000 Évora, Portugal
{pq|ipr}@di.uevora.pt

Abstract. A multi-agent based architecture for the Portuguese Attorney General's Office documents is presented.
The architecture has two layers: the first one uses natural language processing techniques to manage the legal text bases; the second one, uses dynamic logic programming to define agents, which cooperatively handle the interaction between users and the text bases.
The natural language processing layer uses a lexical dictionary and a Portuguese POS (part of speech) tagger to improve the results of the text search engine. Moreover, the retrieved documents are clustered in topics, helping the users to refine their queries and to select the desired documents.
The interaction layer uses agents to model cooperativity and to handle the multi-modal interactions (natural language sentences and graphical actions).
The proposed architecture was implemented in a Linux environment using Prolog.

1 Introduction

As the size and complexity of the legal text bases increase, users develop new desires [YS99]:

- Legal systems should be able to act as rational, autonomous, cooperative agents helping them in their searches;
- Legal systems should be able to represent and to use the knowledge conveyed by the legal texts in order to improve the search results.

We propose a new architecture, which is able to partially satisfy these new requirements through the integration of several areas of artificial intelligence, namely:

- Text information retrieval procedures – A text base search engine is used to handle the basic text retrieval operations;
- Natural language processing techniques – Linguistic information, such as lexical dictionaries and part-of-speech taggers, is used to improve the power of the text retrieval engine;

– Clustering procedures – Clustering algorithms are used to divide the set of retrieved documents into sets of topic-related texts;
– Logic programming and agents – Dynamic logic programming is used as the base tool to define rational and autonomous agents [PQ98,APP+00];

The proposed architecture has two layers:

– The information retrieval layer – This layer integrates the text search engine with the natural language processing techniques;
– The interaction layer – This layer is the responsible for handling the interaction between users and the information retrieval modules.

The two layers are implemented as autonomous agents, which are able to communicate between them and with other agents. Communication is achieved through the interpretation of the received actions in the context of the agents' mental state. Each agent models its mental state, namely its beliefs, intentions and goals, and plans its own actions trying to be as much cooperative as possible.

Cooperation is achieved through the inference of the other agents intentions from their actions. The inferred agent intentions are the input of a abductive planning procedure, which selects the actions needed to satisfy the agents goals.

In the context of legal information retrieval, one of the main goals of our work is to show the need for interaction management capabilities and to develop a system that is able to achieve a better degree of cooperativeness and to reduce the average number of interactions needed to retrieve the intended set of documents.

This architecture was applied to the Portuguese Attorney General information retrieval system in the context of a Portuguese co-funded research and development project [QR99] and the system is available in the web (http://www.pgr.pt – in Portuguese).

As it will be described in the next sections, some of the components of our system can be compared with other existent legal IR systems. For instance, our IR agent is based on SINO [GMK97] and it was changed in a way that has many similarities with the work described in [BvWM+99], namely, allowing the extraction of textual information using localisation, inference, and controlled vocabulary. As in [OS99], we are also able to use concepts and a concept taxonomy in order to retrieve sets of documents. On the other hand, in the dialogue management domain, the use of speech acts to recognise plans has many similarities with the work of Carberry [CL99] and Litman [LA87] and the representation of plans as mental attitudes was also the approach followed by Pollack [Pol90] in her work.

In the next section we will describe in more detail the proposed architecture. Then, in section 3, the information retrieval agents will be described and in section 4 we will present the dialogue management agents. Finally, in section 5 some conclusions and future work are pointed out.

2 Architecture

Our system is based in a two-layer agent based arquitecture.

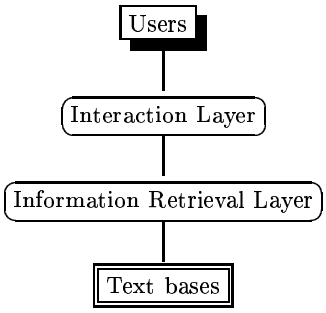

Fig. 1. Architecture

2.1 Interaction Layer

The first layer – Interaction Layer – is the responsible for handling the user actions and to answer them. In order to achieve these goals there are two sub-layers composed by two different and specialised agents, the interaction manager agent and the user agent, which handle all the interactions, modelling the users, inferring their intentions, and communicating with the information retrieval agents in order to obtain sets of documents.

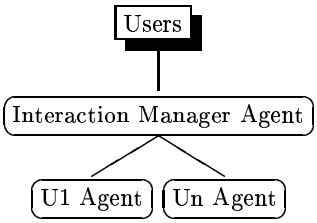

Fig. 2. Interaction Layer

The interaction manager agent receives all the users actions (sentences or graphical actions, such as clicking in a button) and redirects them to the specific and specialised user agent. In fact, each user has a correspondent user agent, which manages his model and is the responsible for inferring his goals and to try to satisfy them (by answering his direct and indirect queries). Figure 3 shows, for user agent i its links with the other agents.

The user model is a representation that each user agent manages, updating it after each event with the new inferred user attitudes. These attitudes will be the basis for a user agent planning procedure, which normally obtains as output a plan composed by actions that are sent to the information retrieval agent

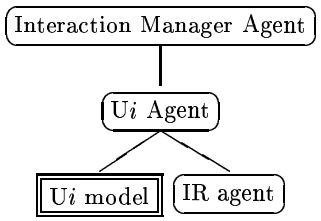

Fig. 3. User i agent

(in the information retrieval layer). Finally, the answers from the information retrieval layer are received by the user agent, and are sent to the interaction manager agent, which redirects them to the user. A detailed explanation of these interaction layer agents is presented in section 4

2.2 Information Retrieval Layer

The second layer – Information Retrieval Layer – is the responsible for accessing the legal documents and to inform the other agents about them. It is composed by the following agents:

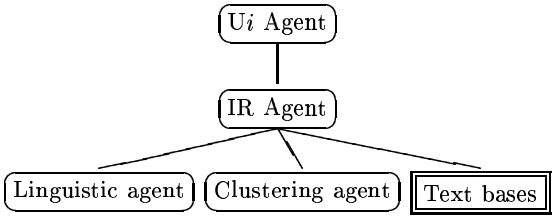

Fig. 4. Information Retrieval Layer

The IR agent receives queries from each user agent and it has to answer them. It can access directly the text bases, or it can use the linguistic agent to obtain the canonical forms and the morpho-syntactic tag for each word. Using this approach it is possible to handle plurals and verbal forms and to disambiguate situations where one word has several morpho-syntactic tags (nouns and verb, for instance). Finally, the set of retrieved documents from the text bases can be clustered into sub-sets of topic-related documents. The clustering agent uses an heuristic-based approach to obtain these sub-sets and its output relies in a previous topic classification made by an automatic neural network classifier. A detailed description of this layer is presented in the next section.

3 The Information Retrieval Layer

As it can be seen in figure 4 the information retrieval layer is composed by three agents (IR, linguistic, and clustering) and the text bases. In the next four sub-sections these modules will be presented.

3.1 Text bases

As it was already pointed out, the text bases are composed by the documents produced by the Portuguese Attorney General's Office. These text documents were scanned and, at present, the text base has around 7,000 documents since 1940 with near 10,000,000 words. The documents have a specific structure and they were saved in an XML format. In this format, each document section was associated with a specific XML tag. For instance, these are some of the existent sections/XML tags:

- Number
- Title
- Topics
- Date
- Author
- Conclusions
- Full text

The XML files were also changed by adding to them linguistic information obtained through off-line processing. Namely, it was added to each word:

- Its canonical form
- Its morpho-syntactic tag (verb, noun, ...)

The canonical form of each word is obtained through the access to a lexical database, POLARIS[1], which has more than 900,000 words with information about their canonical forms and possible morpho-syntactic tags. This database was produced in the context of a previous research project at the AI Centre of the New University of Lisbon.

The morpho-syntactic tag is obtained from the output of a neural network tagger [ML01]. The neural network is basically a two-layer network having as input the probabilistic value of a specific tag, taking into account the next word in the document, and producing as output the probabilistic value of each tag. The neural network was trained with a subset of around 5,000 words and its evaluation shows correct results higher than 95% (more information can be found at the cited paper).

At present, there are contacts with the Portuguese Ministry of Justice trying to increase the text bases with other legal documents from several Portuguese Courts.

[1] POrtuguese Lexicon Acquisition and Retrieval Interaction System

3.2 Information Retrieval Agent

The information retrieval agent is responsible for answering the user agents about the documents that match specific queries.

As a base tool, it was used an existent text search engine already used in the legal domain, SINO [GMK97], from the AustLII Institute. SINO uses inverted files to implement searches. It also supports *boolean*, *near*, *wildcards*, and *ranking* operators.

In order to use the linguistic information added to the XML documents, the IR agent may ask the linguistic agent (see next section) about the canonical form and the possible tags of each word in the user queries. The obtained linguistic data can be used in the document search. For instance, it is possible to ask for documents with any verbal form of a specific word, or for documents where a specific word appears as a noun.

The linguistic agent is also able to inform the IR agent about word synonyms and, as a consequence, it is possible to search for documents where a specific word or a synonym appears.

Finally, the retrieved set of documents can be clustered into sub-sets of topic-related documents. The clustering procedure is implemented by another agent: the clustering agent (subsection 3.4).

3.3 Linguistic Agent

As described in the previous section, the linguistic agent may receive requests from the information retrieval agent to obtain equivalent queries. By equivalent queries, we mean queries where each word is replaced by the correspondent canonical form. Moreover, each word is associated with a list of its synonyms.

In section 3.1 the lexical database allowing the linguistic agent to obtain canonical forms was already described. Synonyms are obtained through the use of two approaches:

- Using a juridical terms thesaurus
- Using an automatic extracted thesaurus

The first approach, the juridical thesaurus, is based on a manually constructed thesaurus, result of the research of the Portuguese Attorney General's Office. This juridical thesaurus has more than 6,000 terms, with the following relations:

- is equivalent to
  ex: law is equivalent to norm
- is generalised by
  ex: prime minister is generalised by minister
- is specified by
  ex: accident is specified by traffic accident
- is related with
  ex: desertion is related with traffic accident

This thesaurus allows the IR agent to expand queries to include all the values that are equivalent or more specific or related, with the initial query. For instance, the user query "documents about accidents?" is expanded to the query "documents about accidents or traffic accidents or ...", which includes all the related and the more specific terms of accident.

The result is an IR system, which has many similarities with the work described in [BvWM+99], namely, allowing the extraction of textual information using localisation, inference, and controlled vocabulary.

The second approach, the automatically extracted thesaurus, is based on a statistical approach to extract thesaurus from large text bases [PGL01]. The idea of this approach is to measure the similarity between words, which appear in texts with the same function (morpho-syntactic tags, complements, modifiers, ...). One of the major advantages with this statistical approach is the independence of the text domain and the text language.

3.4 Clustering Agent

The clustering agent (ca) is the responsible for calculating sub-sets of topic-related documents.

As it is well-known, document clustering is a complex process [Sal89] since it involves the choice of a representation for the documents, a function for associating documents and a method with an algorithm to build the clusters. One of the best clustering methods is the Scatter/Gather browsing paradigm [CDRKT92,CKP93,HP96] that clusters documents into topically-coherent groups. It is able to present descriptive textual summaries that are built with topical terms that characterise the clusters. The clustering and reclustering can be done on-the-fly, so that different topics are seen depending on the subcollection clustered.

Our implementation of the cluster algorithm assumes that each document has already associated a list of topics describing its classification. Then, the cluster algorithm has to divide the set of retrieved documents is such a way that:

1. The union of the set of documents associated to all the topics is the initial set of documents.
2. The intersection of the set of documents associated to any two topics is empty.

The clustering algorithm, basically searches the state space, using some heuristics in order to cut off the complexity. It can be classified as an informed search algorithm, a best first search with an evaluation function specially designed for this problem.

In the scope of this work, there are two possible approaches to the classification problem:

- Manual classification
- Automatic classification

In the first approach, the agent uses a manual classification from the Attorney General's Office. In fact, every document was manually classified accordingly with the taxonomy developed at the AG's Office. In the second approach, an automatic classifier was developed using a neural network [QR00].

4 The interaction layer

As it was pointed out in section 2 the interaction layer is based on two specialised logic agents [QR01]:

- An interaction manager, which receives the users web initial requests and it is responsible to establish a connection with a specific user agent;
- A specialised user agent that given an user request is able to cooperatively interact with him (inferring the user intentions, planning the system cooperative actions, and communicating with the information retrieval layer).

As it can be seen in figure 2, each user initially communicates with the interaction manager agent, which redirects the event to the specific user agent (launching the user process, if needed). In order to obtain a cooperative answer, each user agent infers the user intentions and it interacts with the information retrieval agent (figure 3). Afterwards, it updates the user model and communicates the answer to the interaction manager agent.

4.1 The interaction manager agent

The interaction manager agent receives the initial user requests, analyses them and redirects them to the respective user agent.

As it is not possible to have all the user agents running at the same time (our system may have thousands of users), our solution was to have alive only the user agents correspondent to the active users. So, one of the interaction manager tasks is to keep track of the active users and to launch the respective user agent, if needed.

4.2 The user agents

User agents are specific to each user and they are responsible for processing their requests. They are launched by the interaction manager in order to receive the user requests. Then, they consult the user models to obtain the interpretation contexts for the requests and to support the inference of the user intentions and their own plans. Afterwards, they access the text bases via the information retrieval agent and, finally, they answer the user (via the interaction manager).

In summary, to fulfil a user request the user gent must:

- Load the user model. The user model is a dynamic logic program where rationality, behaviour, knowledge and events are described.

- Interpret the user action in the context of the actual user model. The user agent must infer the user intentions and beliefs from its actions.
- Perform a set of inferred actions, in order to fulfil the user intentions (communicating with the information retrieval agent, if needed).
- Save the new user model.

User agents' behaviour In order to be collaborative user agents need to infer the user attitudes (goals, intentions and beliefs). As it was already pointed out, this task is also achieved through the use of dynamic logic programming rules.

The agent's mental state can be decomposed in several modules (see [QL95] for a complete description of these modules):

— Description of the effects and the pre-conditions of the possible user actions in terms of beliefs and intentions;

— Definition of behaviour rules that define how the attitudes are related and how they are transferred between the different agents (cooperatively).

After each event (for instance a user request) received by the user agent, the agents' model is updated with the description of the event that occurred. Then, the user agent calculates the new user model (well founded model of the logic program) and infers the user attitudes.

User agents' plans and actions The user agent actions are the result of an abductive planning process, which tries to satisfy the inferred agents' intentions.

Suppose I is the set of inferred agents' intentions at a given time and that A is the set of correspondent intended actions (each intention has as object an action). The planning process is started by the creation of a new set of logical constraints, such that, for each intended action there exists an associated constraint stating that the action must be performed.

After the creation of this set of constraints, the user agent will abduce the set of actions needed to satisfy the constraints. Note that in a general domain, this task may not be always possible. However, in this domain we assume that the user agent is always able to satisfy the user intentions through the communication with the IR agent (this means that it is always possible to answer a user query).

The abduced agent actions need to be performed and its results should be answered to the user (via the interaction manager agent). In order to perform the inferred actions it will be necessary to communicate them to th IR agent and to obtain the answers.

5 Conclusions

A logic programming agent based architecture for a cooperative legal information retrieval system was presented.

The system is divided in two layers: interaction layer and information retrieval layer. The interaction layer is composed by two kind of agents: the interaction manager, which interacts with the user and with the specialised user

agents; and the user agents, which model the users and interact cooperatively with them. The information retrieval layer is composed by three agents: the information retrieval agent, which is responsible for answering user agent queries about documents; the linguistic agent, which helps the information retrieval agent with synonyms, canonical forms, and morpho-syntactic forms; the clustering agent, which also helps the information retrieval agent grouping the selected documents in clusters of topic-related ones.

A cooperative behaviour is achieved through the integration of the dialogue processing techniques of the interaction layer with the IR approach of the information retrieval layer. In fact, this integration allows the system to interpret user queries in the context of the user model and to infer the *real* user intentions. Moreover, it is possible to, taking into account these intentions, cluster the retrieved documents into topics and to help users in the refinement of their queries.

The system was implemented using Prolog over the legal information retrieval system of the Portuguese Attorney General's Office (available in Portuguese from http://www.pgr.pt).

As future work, we intend to apply the system to other domains and to other legal information retrieval systems, namely to non-Portuguese legal documents.

References

[APP+00] J. J. Alferes, L. M. Pereira, H. Przymusinska, T. C. Przymusinski, and P. Quaresma. Dynamic knowledge representation and its applications. In S. Cerri and D. Dochev, editors, *Proceedings of the 9th International Conference on Artificial Intelligence - Methodology, Systems, Applications (AIMSA'2000)*, number 1904 in Lecture Notes in Artificial Intelligence, pages 1–10, Varna, Bulgary, September 2000. Springer Verlag.

[BvWM+99] Tania Bueno, Christiane von Wangenheim, Eduardo Mattos, Hugo Hoeschl, and Ricardo Barcia. Jurisconsulto: Retrieval in jurisprudencial text bases using juridical terminology. In *Proceedings of the ICAIL'99 – 7th International Conference on Artificial Intelligence and Law*, pages 147–155. ACM, June 1999.

[CDRKT92] D. R. Cutting, J. O. Pedersen D. R. Karger, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proc. 15th Annual Int'l ACM SIGIR Conf. on R&D in IR*, June 1992.

[CKP93] D. R. Cutting, D. Karger, and J. Pedersen. Constant interaction-time scatter/gathern browsing of very large document collections. In *Proc. of the 16th Annual Int. ACM/SIGIR Conf.*, Pittsburgh, PA, 1993.

[CL99] Sandra Carberry and Lynn Lambert. A process model for recognizing communicative acts and modeling negotiation subdialogs. *Computational Linguistics*, 25(1), 1999.

[GMK97] G. Greenleaf, A. Mowbray, and G. King. Law on the net via austlii - 14 m hypertext links can't be right? In *In Information Online and On Disk'97 Conference, Sydney*, 1997.

[HP96] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis:scatter/gather on retrieval results. In *Proceedings of the Nineteenth Annual International ACM SIGIR Conference*, Zurich, June 1996.

[LA87] Diane Litman and James Allen. A plan recognition model for subdialogues in conversation. *Cognitive Science*, 11(1), 1987.

[ML01] Nuno Marques and Jose Gabriel Lopes. Tagging with small training corpora. In *LNAI – Proceedings of the International Conference on Intelligent Data Analysis*. Springer-Verlag, 2001.

[OS99] James Osborn and Leon Sterling. A judicial search tool using intelligent concept extraction. In *Proceedings of the ICAIL'99 – 7th International Conference on Artificial Intelligence and Law*, pages 173–181. ACM, June 1999.

[PGL01] A. Agustini G. Lopes P. Gamallo, C. Gasperin and V. Lima. The use of syntactic context for measuring word similarity. In *ESSLI – Proceedings of the Workshop on Semantic Knowledge Acquisition and categorization"*, 2001. To appear.

[Pol90] Martha Pollack. Plans as complex mental attitudes. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communications*. MIT Press Cambridge, 1990.

[PQ98] Luis Moniz Pereira and Paulo Quaresma. Modelling agent interaction in logic programming. In Osamu Yoshie, editor, *INAP'98 - The 11th International Conference on Applications of Prolog*, pages 150–156, Tokyo, Japan, September 1998. Science University of Tokyo.

[QL95] P. Quaresma and J. G. Lopes. Unified logic programming approach to the abduction of plans and intentions in information-seeking dialogues. *Journal of Logic Programming*, 54, 1995.

[QR99] P. Quaresma and I. Rodrigues. Pgr: A cooperative legal ir system on the web. In Graham Greenleaf and Andrew Mowbray, editors, *2nd AustLII Conference on Law and Internet*, Sydney, Australia, 1999. Invited paper.

[QR00] Paulo Quaresma and Irene Pimenta Rodrigues. Automatic classification and intelligent clustering for wwweb information retrieval systems. *Journal of Information Law and Technology (JILT)*, 2, 2000. http://elj.warwick.ac.uk/jilt/00-2/ – Extended and revised version of the BILETA'2000 paper.

[QR01] Paulo Quaresma and Irene Rodrigues. Using logic programming to model multi-agent web legal systems – an application report. In *Proceedings of the ICAIL'01 - International Conference on Artificial Intelligence and Law*, St. Louis, USA, May 2001. ACM.

[Sal89] Gerard Salton. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley, 1989. Reading, MA.

[YS99] John Yearwood and Andrew Stranieri. The integration of retrieval, reasoning and drafting for refugee law: a third generation legal knowledge based system. In *Proceedings of the ICAIL'99 – 7th International Conference on Artificial Intelligence and Law*, pages 117–125. ACM, June 1999.