Keeping Context in Web Interfaces to Legal Text Databases

Paulo Quaresma and Irene Pimenta Rodrigues
Departamento de Matemática
Universidade de Évora
Largo dos Colegiais
7000 Évora
Portugal
{pq|ipr}@dmat.uevora.pt

CENTRIA – AI Center
Departamento de Informática
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2825 Monte da Caparica, Portugal

Abstract
In this work we propose a framework for a system that is able to build and to keep the context of the interactions in web interfaces to legal text databases.
We are using some of the results of the AustLII (Australasian Legal Information Institute) project with its legal inference system via World Wide Web as an interface for legal text databases. Namely, we are using SINO, a search engine for legal text databases (Greenleaf, Mowbray and King 1997). At the moment we are using SINO to process legal texts from our project partner (PGR, Procuradoria Geral da República - Portuguese Attorney General). The resulting database is used to test the ability of our system to generate SINO queries from user natural language questions. These queries are generated taking into account the dialog context that is built using the previous user questions and previous system answers.
Moreover we use YSH, an inference engine (Greenleaf, Mowbray, and van Dijk 1995) to build rule bases modeling some legal knowledge. This enable us to test some more advanced features, namely the ability of translating the quasi natural language sentences into our logic representation language (an extended DRS language).

1.Introduction

In this paper we present a system that has an intelligent web interface for a database with legal texts. The main purpose of the interface is to allow the search and the view of the database documents. The user poses a query and it should obtain as an answer the set of documents that satisfy the query.
Our system is a hybrid system in the sense that it uses different sources of knowledge in order to achieve its goals.

We are using some of the results of the AustLII (Australasian Legal Information Institute) project with its legal inference system via World Wide Web as an interface for legal text databases. Namely, we are using:

- SINO, a search engine for legal texts databases (Greenleaf, Mowbray, and King 1997). In our recent project funded by PRAXIS we are processing legal texts from the Portuguese Attorney General (PGR) using SINO. We are using the resulting databases to test the ability of our system to generate sino queries from user natural language questions. These queries are generated taking into account the dialog context that is build using the previous user questions and previous system answers.

- YSH, an inference engine (Greenleaf, Mowbray, and van Dijk 1995). We have built a rule base modeling some legal knowledge, namely ruling the conditions under which "A person has a legally right to obtain a pension for exceptional services". This enable us to test some other features of our system, namely the ability of translating the quasi natural language sentences into our logic representation language (an extended DRS language). Our system is able to interpret natural language sentences and transform them in a logic form; we have preprocessed some of the documents and augmented its text with the logic representation of some parts. This way we are able to test our rule base with the documents in the database (see section 4).

The intelligent web interface has several modules such as:
- The SINO search engine indexing all words in the documents, including the text and all the fields such as document conclusion, document descriptors and document administrative data. The documents are preprocessed in order to incorporate some linguistic and semantic information that may be used in the retrieval process such as: tagged words and expressions, and in some cases a logic formula representing the meaning of parts of the text.
- A module for performing and representing the inference of user attitudes, including a planner.
- A juridical terms taxonomy, a thesaurus, and a module that allow us to relate natural language expressions with terms in the taxonomy.

- A dictionary including language synonyms.
- A module for linguistic analysis including a partial parsing, a module for building a semantic representation of natural language expressions (DRS like), and a module for pragmatic interpretation (solving anaphora and building a discourse (dialogue) structure).

The intelligent web interface has three stages:
1. It tries to determine what are the user intentions with his query. In this stage we build a model of the user by trying to infer its attitudes (beliefs and intentions). Whenever the set of users intentions can not be fulfilled the system goes to the second stage.
2. It tries to clarify the user intentions in order to reach a set that can be fulfilled.
3. It tries to fulfill the user intention.

Our system also allows us to test the rule-bases that were build using the text data in the SINO system. Some of our legal texts have a section with a description of the case and a section with the conclusions. Given a rule-base that models some legal knowledge we can test it using the following approach:

1. We process the rule-base to extract the relevant words from the language rules.
2. We augment the set of relevant words using the thesaurus of juridical terms built by our partners (PGR).
3. We generate SINO queries to select a set of relevant texts from our text database.

For each selected text we run the rule-base system answering the YSH questions by transforming the quasi-natural language question into a extended DRS and then we see if the DRS that represent the text entails it.

This way a user may be able to see if a case description satisfies the rule-base. Comparing these results with the text conclusion section the user is able to conclude about the correctness of its rule-base.

2.Overview of the interface system

Our interface is multimodal; it allows the user to pose questions in SQL like boolean expressions (using menus and buttons) or in Natural language expressions. In any case the system tries to keep the interrogation context, i.e. in order to build an answer it uses all the previous user questions and system answers.

The user first query

A question by an user may be a natural language expression, such as:

Q: In which documents a person obtains a pension for exceptional services?

This question, after translating it into a DRS gives rise to:
$x, p, t, e: *person(x), pension_exceptional_services(p), evt(e, has(x,p)), occurs(e,t).*

In order to answer the user query it will be enough to look for the documents which entails this logic expression.

- But there are too many documents that are not about exceptional services pensions, we will be wasting time by trying to check all documents;
- and by now it is no possible to obtain a logic representation for all the documents representing its meaning.

So, what we will do is:

- to relate the logic form with "a boolean query expression" (a SINO query) that will probably occur in a document that satisfies the logic form, and select all documents that satisfy the boolean expression. (In this example the expression could be: favorable and descriptor ="pension for exceptional services".
- If a selected document has a logic form associated with it, we test the entailment, if not, we consider that it entails the intended situation. With this strategy we may get more documents then the desirable.

User questions may also have different formats:
- pension and exceptional (meaning that user want documents where the word "pension" and the word "exceptional" are present).
- descriptor = pension for exceptional services (meaning that the user wants the documents where the field descriptor has the value "pension for exceptional services").

These are already SINO queries, and what the system does is to transform them using our juridical thesaurus (see section 3).

Query Refinement

After obtaining the final user query, the system has the set of documents selected by the query and it can fulfill the user intention, unless there are contradictory intentions such as:

- 100 is the number of selected documents
- the user does not want to see more then 20

The recognition of user intentions is done through the analyses of the user acts. In fact each user act can be translated into a logical formula which is used to update a logic program describing the user model. In this user model we have rules relating acts and intentions and goals, allowing us to infer user intentions and plans. However, this inference may create a contradictory state between actual and previous intentions. In this situation, a contradiction removal process is has to be started.

In fact, when the system reaches contradictory intentions it enters in a process of documents refinement. The refinement is done by grouping sets of texts using the juridical thesaurus and the values of the field descriptor. We build the set of all documents descriptor values and use the thesaurus in order to collapse the set of values into a smaller set (see section 3).

As an example, suppose that the set of documents selected with the query ”accident” after the collapse operation gives rise to the following descriptor values:

- working accident
- traffic accident
- house accident
- other accident

As an answer to the query our system supplies the user:

- The number of documents selected
- The possibility of query refinement that includes the above set of descriptor values
- The possibility of dropping the intention of not seeing many documents (e.g. more then 20)
- The possibility of introducing a new constraint, a natural language or boolean expression that will be interpreted in the context of the first one.

Interpretation context

Our system takes into account the interpretation context of the user query in both kinds of queries, natural language sentences and boolean expressions.

When the user is interacting using natural language sentences it is necessary to keep the interrogation context in order to enable us to solve discourse phenomena such as pronominal, nominal temporal and anaphora; and to capture the user intended meaning of their sentences.

For instance if the user first question is:

Q1: In which documents a person obtains a pension for exceptional services?

After the system’s answer the user may ask:

Q2: In which of those documents is the person a fireman?

In order to obtain the meaning of the second sentence the system must solve the anaphora: "those documents" and "the person".

If, instead of Q2, the user used Q2':

Q2': In which documents the person is a fireman?

The system will infer that the user intention with this question is to ask:

Q: In which documents a fireman obtains a pension for exceptional services?

But if the second question were Q2'':

Q2'': In which documents a person has his request of a pension for exceptional services rejected?

The system is able to detect that the user is starting a new context, i.e. this sentence does not need the information of Q2 to be interpreted.

When the user first query is a boolean expression such as Q1:

Q1: pension for exceptional services

That is followed by Q2:

Q2: favorable

The dialog system is able to infer that the possible user intentions are:

1.Documents with the sequence of words "pension for exceptional services" and the word "favorable"
2.Documents with word "favorable"

Since the system infers these two possible intentions it will give two answers, one for each intention and it presents the intention before the answer.

If the user poses a new question such as Q3:

Q3: rejected

The dialog system is able to infer that now the possible user intentions are:

1.Documents with the sequence of words "pension for exceptional services" and the word "rejected"
2.Documents with the sequence of words "pension for exceptional services" and the word "favorable" and the word "rejected"
3.Documents with word "rejected"

If the system finds out that for the second intention there are 0 documents, it will assume that it is not a valid intention and it will present just the answers for the other two intentions.

The way context is taken into account by our system gives the user the idea that he is interacting with an intelligent agent.

3.Using the juridical Terms Thesaurus

The juridical terms thesaurus is a taxonomy with the descriptor values that has the relations:

- is equivalent to
ex: **law** is equivalent to **norm**
- is generalized by
ex: **prime minister** is generalized by **minister**
- is specified by traffic
ex: **accident** is specifi ed by traffic **accident**
- is related with
ex: **desertion** is related with **traffic accident**

This knowledge is used to:

- Expand queries.
  Whenever a query that specifies the value of a descriptor is made, we expand it with all the values that are: equivalent or more specific or related, with the initial descriptor value.
- Collapse sets of descriptor values.
  Whenever we have a large set of documents selected we collect all the possible pairs: descriptor value – document number. It is possible to join and collapse using the generalize relation this pairs in order to obtain a smaller set of pairs with the form: descriptor value – set of document numbers.

Example

Suppose a user wants to be informed about legal texts about "accidents":

- Documents about accidents?

The system expands the query using the thesaurus and it searches for all the related and more specific values. For instance, it will search for "accident AND traffic accident AND desertion AND …"

Then, the answer is used to collapse the set of texts into classes of answers grouped by the thesaurus

terms:

- X documents about accidents;
- Y documents about traffic accidents;
- Z documents about desertion
- …

As it was described in section 2, it is possible to the user to refine this question using the context of the interaction.

4.The Inference Engine

Another important goal of our project is to be able to model legal knowledge. In order to handle this problem we need:

1.A legal inference engine;
2.A knowledge bases representation formalism;
3.A user interface;
4.Rule bases describing the modeled legal knowledge.

As a legal inference engine we are using YSH, an inference engine from the AustLLI project (Greenleaf, Mowbray, and van Dijk, 1995). YSH implements rule-based inferencing, with the capability to model forward and backward chaining. Moreover it uses a 'quasi natural language' representation with entities declared to be a PERSON, a THING, or a PERSONTHING. The inferencing dialogues are generated dynamically.

In order to represent the legal texts we are using extended DRS - Discourse Representation Structures (Rodrigues, 1995; Kamp 1990). For the moment we are not representing the complete texts but only the sections where the cases are described. Another related problem is the construction of each DRS: as we do not have a complete parser for the Portuguese Language, the DRS construction is done semi-automatically. This is a problem that limits the generalization of our system and it should be handled as future work. The DRS are represented using a logic programming environment (Pereira et al. 1996) that supports non-monotonic reasoning, namely, default reasoning, and that is able to obtain the well-founded model of the logic program.

As user interface we are using a www environment with WYSH - the web-ysh user interface from AustLII (Greenleaf, Mowbray, King, Cant, and Chung, 1997) - and an interface with XSB-Prolog - the logic programming framework needed for inferences over the DRS. Finally, we need to describe the legal knowledge using the YSH rules. Due to the complexity of this task, for the moment we have chosen only some specific domains, namely, the legislation that defines when a person has a right for a pension for exceptional services.

As described in section 1, given a rule-base that models some legal knowledge we can test it using the following approach:

1. Extract the relevant words from the rules.
2. Use the thesaurus of juridical terms.
3. Generate SINO queries to select a set of relevant texts.
4. For each selected text, run the rule-base system answering the YSH questions by transforming the question into a extended DRS and then checking if the text DRS entails the question DRS (using the logic programming environment).

This approach enables us to see if the case description satisfies the rule-base. By comparing the results with the text conclusion section we can to conclude about the correctness of the rule-bases.

Example

In this section we will show an example over the legislation that defines when a person has a right for a pension for exceptional services. However, due to its extension and complexity, we will make some simplifications over the legislation:

GOAL RULE Pension for exceptional service art3 PROVIDES
A person has a legally right under art3 to obtain a pension for exceptional services ONLY IF
art3(1) applies AND art3(2) applies.

RULE Art3(1) applies ONLY IF
A person has made an action that is considered exceptional and relevant to Portugal (exceptional_action) AND (art3(1a) applies OR art3(1b) applies)

RULE Art3(1a) applies ONLY IF
A person has made an exceptional action in a war place (war_place) OR
A person has made an abnegated and courageous action OR
A person has made a high service to his country or to the Humanity

RULE Art3(1b) applies ONLY IF
(A person has made an humanity act OR
A person has made an act of dedication to the public cause)
AND a person has been injured or has deceased.

RULE Art3(2) applies ONLY IF
A person has constantly showed respect for the individual and collectives rights and freedom AND
A person has constantly showed respect for the prestige and dignity of the country.

RULE exceptional_action applies ONLY IF
A person has made an action which benefits the country AND
A person has made an action with a correct typology (typology) AND
A person has made an action without any remuneration AND
A person has made an action that goes beyond the duty of his functions.

RULE typology applies ONLY IF
A person has made an action that serves the national interests AND
A person has made an action that presupposes a capacity of high-availability.

RULE war_place applies ONLY IF
A person has made an action in a war place AND
A person has made an action that goes beyond the standard military actions (beyond_standard_military_actions).

RULE beyond_standard_military_actions applies ONLY IF
A person has made an action which is considered beyond the standard military actions by the Military Administration OR
A person has made an action which places the defense of others lives above the defense of his own live OR
A person has made an action which places the defense of others lives above his own injures.

Suppose that we want to test these inference rules over the legal texts:

1. We process the rules to extract the relevant words from the quasi-natural language rules. For instance, a word is considered relevant if it is a noun or a noun expression in the goal rule. In this example we would get *pension*, *exceptional services*, *person*, *right*.
2. We augment the set of relevant words using our thesaurus of juridical terms. In this example we have: *person -> citizen*
3. We generate SINO queries to select a set of relevant texts from our text knowledge base: 189 in our example.
4. For each selected text we run the rule-base system answering the YSH questions by transforming the question into an extended DRS which is tested to see if it is entailed by the DRS that represents each text. This process is done through the use of a logic programming framework implemented over XSB-Prolog. For instance, the question " Has the person made an action in a war place?" can be represented by the following DRS:

   $x,e1,e2,p,t1,t2: *person(x), occurs(e1,t1), holds(e2,t2), t1 Ì t2, agent(e1,x), war_place(p), evt(e2,in(x,p)).*

5. Using the inference process, we can have one of the following three results: the person has right to the pension, the person does not have right to the pension, or the result is unknown.
6. As a final point, the inferred results are compared against the legal texts using the conclusion section. The following SINO query can be used:

search pension for exceptional services;
(conclusions(has right), conclusions(satisfies))
NOT (conclusions(not satisfies), conclusions(not has right))

7.Analyzing the global results we are able to conclude about the correctness of the rule-bases.

Using these inference rules we have detected that initially most of the answers were "unknown". The problem is related with the need for a more complete and powerful domain knowledge representation. For instance, the question " Has the person made an action that goes beyond the duty of his functions?" has a great probability of being answered as "unknown" if there are no rules describing actions that go beyond the duty.

One example of such a rule could be:

RULE A person has made an action that goes beyond the duty of his functions applies IF
A person is a fireman AND
A person has made an action during an off duty time AND
The action presupposes the risk of his live.

Moreover, it is necessary to define rules for "risk of live":

RULE An action presupposes the agent's risk of live IF
The agent enters a burning house in order to rescue somebody AND
The agent has no backup support.

Clearly, this process must be refined again: it's necessary to define what is a rescue operation and what is a backup support.

In fact the creation of the domain knowledge base is one of the major tasks of our project. At the moment we are still building this knowledge base and we are using the following approach:

1.Identify the legal texts where the result is unknown;
2.Identify the reason of the result;
3.Add the needed domain rules in order to fully describe the situation;

After processing all the texts, we expect to have a good representation of the domain and we will be able to start the next phase:

•Use the system to suggest the result of new cases

The answer's suggestion can be explained and supported by the legal rules applied and by the domain knowledge used.

5.Conclusions

As described the proposed framework has the following main features:

- It is able to search large text databases using linguistic and semantic information, namely making the pragmatic interpretation of part of the documents;
- It is able to build and to keep the context of the interactions;
- It represents the user intentions and beliefs;
- It is able to revise attitudes and to infer new ones whenever is necessary, namely if there are inconsistencies;
- It uses a juridical taxonomy in order to classify the documents;
- It allows the definition of rules to model legal knowledge;
- It can apply the rules defined over the legal database and it can compare the results.

However, much work has still to be done:

- In order to improve the linguistic and semantic interpretation module we need to improve the natural language processing tools (taggers, corpora, syntactic and semantic analyzers,…)
- The dialogue manager should be generalized to handle more powerful interactions;
- The domain knowledge must be completely represented.

6. Bibliography

- [AKPT91] James Allen, Henry Kautz, Richard Pelavin, and Josh Tenenberg. Reasoning about Plans. Morgan Kaufman Publishers, Inc., 1991.
- [AP96] José Júlio Alferes and Luís Moniz Pereira. Reasoning with Logic Programming, volume 1111 of Lecture Notes in Artificial Intelligence. Springer, 1996.

- [Bra90] Michael Bratman. What is Intention?, in Intentions in Communication. MIT, 1990.
- [CL90a] P. Cohen and H. Levesque. Intention is choice with commitment. Artificial Intelligence, 42(3), 1990.
- [CL90b] Philip Cohen and Hector Levesque. Persistence, Intention, and Commitment, in Intentions in Communication, pages 33--70. MIT, 1990.
- [Car88] Sandra Carberry. Modeling the user's plans and goals. Computational Linguistics, 14(3):23--37, 1988.
- [Ebe92] Kurt Eberle. On representing the temporal structure of a natural language text. In Proceedings of the COLING'92, pages 288-294, 1992.
- [Esh88] Kave Eshghi. Abductive planning with event calculus. In Proceedings of the International Conference on Logic Programming, 1988.
- [GMD95] G. Greenleaf, A. Mowbray, and P. van Dijk. Representing and using legal knowledge in integrated decision support systems. In Artificial Intelligence & Law, 2(97-142), 1995
- [GMK97] G. Greenleaf, A. Mowbray, and G. King. Law on the Net via AustLII – 14 M hypertext links can't be right? In Information Online and On Disk'97 Conference, Sydney, 1997.
- [GMKCC95] G. Greenleaf, A. Mowbray, G. King, S. Cant, and P. Chung. More than wyshful thinking: AustLII's legal inference via the World Wide Web. In 6th International Conference On Artificial Intelligence and Law, Melbourne, 1997.
- [GS86] Barbara Grosz and Candice Sidner. Attention, intention, and the structure of discourse. Computational Linguistics, 12(3):175--204, 1986.
- [Hob90] Jerry Hobbs, Mark Stickel, Douglas Appelt, and Paul Martin. Interpretation as abduction. Technical Report 499, SRI International, December 1990.
- [KR93] Hans Kamp and Uwe Reyle. From Discourse to Logic: An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory. Dordrecht: D. Reidel., 1993.
- [KS86] Robert Kowalski and Marek Sergot. A logic-based calculus of events. New Generation Computing, 4:67--95, 1986.
- [LA87] D. Litman and J. Allen. A plan recognition model for subdialogues in conversations. Cognitive Science, (11):163--200, 1987.
- [LA91] Alex Lascarides and Nicholas Asher. Discourse relations and defeasible knowledge. In Proceedings of the 29th Annual Meeting of ACL, pages 55--62, 1991.
- [LA92] Alex Lascarides, Nicholas Asher, and Jon Oberlander. Inferring discourse relations in context. In Proceedings of the 30th Annual Meeting of ACL, pages 55--62, 1992.
- [LAD93] L.M. Pereira, C. Damásio, and J.J. Alferes. Diagnosis and debugging as contradiction removal. In Luís Moniz Pereira and Anil Nerode, editors, Proceedings of the Second International Workshop on Logic Programming and Non-monotonic Reasoning, pages 316--330. The MIT Press, 1993.
- [LO93] Alex Lascarides and Jon Oberlander. Temporal connectives in a discourse context. In Proceedings of the 6th European chapter of ACL, EACL'93, 1993.
- [MS88] Mark Moens and Mark Steedman. Temporal ontology and temporal reference. Computational Linguistics, 14(2):15--28, June 1988.
- [Mis91] Lode Missiaen. Localized Abductive Planning with the Event Calculus. PhD thesis, Univ. Leuven, 1991.
- [PR93] J.Pinto and R.Reiter. Temporal reasoning in logic programming: A case for the situation calculus. In D.S. Warren, editor, Proceedings of the 10th ICLP. MIT Press, 1993.
- [Per90] Raymond Perrault. An Application of Default Logic to Speech Act Theory, in Intentions in Communication, chapter 9, pages 161--186. MIT, 1990.
- [Per91] F.Pereira and M.Pollack. Incremental interpretation. Artificial Intelligence, 50:40--82, 1991.
- [Pol90] Martha Pollack. Plans as Complex Mental Attitudes, in Intentions in Communication, chapter 5, pages 77--104. MIT, 1990.
- [QL95] Paulo Quaresma and José Gabriel Lopes. Unified logic programming approach to the abduction of plans and intentions in information-seeking dialogues. Journal of Logic Programming, (54), 1995.
- [Qua97] Paulo Quaresma. Inference of Attitudes in Dialogues. PhD Thesis, Universidade Nova de Lisboa, 1997. In Portuguese.
- [RL92] Irene Pimenta Rodrigues and José GabrielPereira Lopes. Discourse temporal structure. In Proceedings of the COLING'92, 1992.
- [RL93a] Irene Pimenta Rodrigues and José Gabriel Lopes. Building the text temporal structure. In Progress in Artificial Intelligence: 6th Portuguese Conference on AI. Springer-Verlag, 1993.
- [RL93b] Irene Pimenta Rodrigues and José Gabriel Lopes. Text processing in a logic programming framework: The interpretation of tense and aspect. In Proceedings of the fourth International Workshop on Natural Language Understanding and Logic Programming, NLULP4, 1993.
- [Son91] F.Song. A Processing Model for Temporal Analysis and its Aplication to Plan Recognition. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada, 1991.