

Representing Textual Content in a Generic Extraction Model

Nancy McCracken

Center for Natural Language Processing
Syracuse University
4-230 Center for Science and Technology
Syracuse NY 13244
njm@ecs.syr.edu

Abstract

The system described in this paper automatically extracts and stores information from documents. We have implemented a text processing system that uses shallow parsing techniques to extract information from sentences in text documents and stores frames of information in a knowledge base. We intend to use this system in two main application areas: open domain Question & Answering (Q&A) and specific domain information extraction.

Extraction from Documents

The system described in this paper uses a Natural Language Processing system developed at the Center for Natural Language Processing to extract information from documents and store it in a knowledge base. In the past, applications were aimed at MUC-style information extraction that filled in templates of specific types of information. Our current goal is to produce a system that can extract generic frames of information about all entities and events in the sentences of the text and represent relationships between them. This type of system is approaching more complete text understanding in a practical way that does not require expensive processing such as full parsing of the documents.

The heart of the generic extraction system is a set of rules written for a finite-state system that recognizes the patterns of text. These rules are applied in several phases including part-of-speech tagging, bracketing of noun phrases, and categorization of proper noun phrases. Later phases recognize the surface structure of phrases in each sentence and map the phrases to the case frame of the verbs, recognizing the phrases taking the roles of agent, object, point-in-time, etc., and creating a frame representing an "event". The case roles are similar to those in case grammars (Fillmore 1968).

Consider the example sentence:

In addition to these most recent incidents, the Abu Sayyaf have bought Russian uranium on Basilan Island.

The main extractions from this sentence are centered around the verb "buy" with generic attributes of "agent", "object" and "location". Attribute values are linked to their table entries using id numbers.

id = 0
name = Abu Sayyaf
type = terrorist group

id = 2
name = Basilan Island
type = island

id = 3
event = buy
Episode 0
object = Russian uranium
agent = Abu Sayyaf = id 0
location = Basilan Island = id 2

In addition, there is a set of rules to recognize phrases that modify noun phrases, such as appositives, and to create relation attributes in the frame for each noun phrase "entity". These relation attributes were originally modeled after Sowa conceptual graph relations. (Sowa 1984)

He was one of 20 people abducted on May 27 from a luxury resort in the western Philippine province of Palawan.

id = 1
name = Palawan
type = island
Episode 0
isa = western Philippine province

Note that the attribute "isa" is used here for any description, which may include category information but is not restricted to it. Another phase of the processing resolves coreferences of pronouns and some definite common noun phrases, and introduces additional links representing the references.

During the text processing, we use an ontology consisting of about 140 categories of objects arranged in a hierarchy. This is used to categorize entities as far as possible from the textual clues, for example about people

and organizations, and from information in a geographic database. The category is put into the “type” attribute. The database is built both from outside sources and may be augmented by linguistic analysts for a particular domain.

Our system supports automatic acquisition of information from text and incrementally adds information to the knowledge base as more text is processed. Note that since our experience is primarily with news articles and also with some scientific reports, our application deals primarily with the existence and attributes of concrete objects and actions.

Knowledge Representation

The extracted information is represented directly as frames in a knowledge base. The knowledge model consists of the extraction frames with their document context, the ontology, and a set of axioms representing knowledge inferences. An inference engine can work directly on these frames, using abductive inference in general and several specific forms of inference to support the application areas. The inference engine accepts goal frames, where one or more unknown values are represented by variables, and returns a substitution of matches for the variables, ranked by the goodness of the match.

During the inference process, the inference engine searches the knowledge base for frames that match the goal frame. A frame matches if it contains the goal attributes and each of their values match, or an abductive inference rule will allow the frame to match with lower probability if not all of the values are present. Value matching can also use several forms of inference, for example, geographic entities can match if the matching value is regionally contained in the goal value. Another form of inference for value matching is to check the synonym sets of WordNet and to match words of the same meaning. Other forms of inference are based on the Iwanska frame model (Iwanska 2000), with natural language sets and intervals.

The inference engine may also use a set of axioms primarily representing equivalences of language forms. An example is nominalization: if a goal frame wants to find a person with an attribute “inventor” of value represented by a variable ?X, then it is o.k. to match a frame with the event “invent” where the agent is the person and the object is ?X. We do not currently attempt to include large amounts of additional world knowledge, as that is a large enterprise outside of the scope of our work.

One issue of representing knowledge from text is the issue of granularity of the value expressions. We have chosen to use the entire phrase as a value, e.g. “western Philippine province” instead of a finer-grained representation where the value would be “province”, with another frame showing that “province” is modified by “western” and “Philippine”. The coarser representation that we have chosen directly supports the visualization application, where people want to see the large chunks of

information and also creates fewer frame nodes for performance reasons. The drawback is that the inference engine then has to parse the form of the phrases to support value matching. It was not shown in the earlier examples, but in the frame representation, we also save the part-of-speech tagging to facilitate this reasoning.

Applications

The Q&A application is supported by a query processing system that uses query templates to analyze question forms and produce goal frames representing the query. The inferring system accepts the query frames and fills in the answers. The types of questions are those primarily looking for concrete answers as represented by Text REtrieval Conference (TREC) short text-based questions and by logs from web information services. The queries may ask for specific pieces of information, such as “Where is the Taj Mahal?”, or definitional questions, such as “What is anorexia?”, or a question that requires some reasoning about the type of the result, such as “What kind of animal is Winnie-the-Pooh?”

The information extraction application (or evidence extraction) has two additional processing parts. The first of these is a system that can learn particular types of entities and events for a specific domain and specialize the generic extractions to represent a more specialized labeling of the information extracted from sentences. For example, the verb “buy” is recognized as one of the “commerce” verbs with roles of “buyer”, “seller”, “goods”, and “payment” to replace the generic “agent” and “object” attribute labels. The specialization system may also need to restructure information from prepositional phrases to label the information for the specific roles. For each new domain of interest, the specialization rules are learned by a Transformation Based Learning system.

The second of the additional evidence extraction processing is still in the planning stages and will be a system to find multi-document patterns of information about entities and groups of events. It is particularly important to track references to proper nouns across documents, so that information can be coalesced that uses acronyms, nicknames, etc. It is also important to resolve time sequences to detect a sequence or pattern of events.

Evaluation of Extraction

Our evaluation measures are aimed at specific application areas; we don’t currently have an evaluation measure of the knowledge model directly. For the information extraction application, we evaluate extractions with precision and recall measures. Example text is hand annotated with the events and entities that should be extracted and used as the gold standard, and the system output is compared to this and evaluated for precision and recall.

An extraction evaluation in January 2002 used 26 documents with 299 sentences from the news text genre. Precision and recall for proper names are 92% and 94% and for their categorization is 97% and 71%. Precision and recall for numeric concepts are 93% and 88% and for their categorization is 94% and 94%. These high numbers are typical of the traditional information extraction for proper nouns. Recall for events and entities (including proper nouns, but also common noun phrases) is 93% and 61%. Here recall is whatever events and entities have frames built into the knowledge base by a rule – precision is not an issue. But which attributes are identified for the events and entities is an issue: recall and precision for event attributes is 69% and 48% and for entity attributes is 87% and 44%. These lower numbers reflect the difficulty of using shallow parsing techniques to detect sentence clause structure and to correctly do such tasks as prepositional attachment.

Discussion of Challenge Questions

Our system can be adjusted for working on new domains by learning the terminology of the new domain. We use Transformation Based Learning to learn the terminology of the new domain, so this only requires some hand annotation of training text in the new domain. On the other hand, the generic extraction rules of our system are written by linguistic analysts and essentially embody the sentence structures of a particular text genre. So we would need to make some hand adjustments to new text genres, such as transcriptions of spoken dialogs or email.

Knowledge in our model can be inspected and assessed by the visualization application, but we have not currently allowed for hand-editing. In the future, we are planning to have a system in which subject matter experts could edit the information and also attach confidence levels representing confidence in the source documents.

The issues of performance efficiency are relevant both to the document processing part of our system and to the inference engine performing on the knowledge base. Most of our experience has been with the document processing. One example is that for TREC in 2001, we used a dual processor PC server and processed 100,000 documents at a rate of approximately 70 documents per minute. This performance is sufficient for applications where documents can be processed overnight or pre-processed. It does not support processing large numbers of documents on-the-fly in a real-time Q&A system.

We do not yet have a large trial of the performance of the inference engine, which we are using in Q&A applications. However, I can discuss what steps we are taking to try to achieve good performance. The first strategy is to have a two stage answering system. The first stage uses keyword techniques to select a small number of documents, for example, 200, which may contain the answer. The second stage is to use the inference engine to search for matches only in those documents. Our other

strategy is to use the frame representation to reduce the number of nodes stored per document and to reduce the number of inference steps that the inference engine needs to perform. An alternative strategy that many people take is to map their frames or their information extraction to First Order Logic (FOL) and to use FOL inference to answer questions. For example, Harabagiu and Maiorano, (Harabagiu and Maiorano 1999), discuss a strategy in which a full parse is used to generate a FOL representation of candidate answer sentences. This strategy was used very successfully in the TREC2000 Q&A track. But we note that in the example sentence in the paper, there were 13 FOL inference steps to match the query FOL representation to the sentence FOL representation. A frame representation takes a 1 step inference match with 4 value matches. To offset this gain in the number of inferences, however, the value matching uses several special purpose matchers that understand synonyms, numeric intervals, etc.

As far as additional knowledge or theories, we are planning to implement a system of confidence levels that can be used to evaluate the confidence in the document source. We think that this should tie in with a belief system that incorporates opinions from the text as well as the document sources. We also need a better resource for semantics of words than the current WordNet, this includes synonymy, hypernymy (superclass) and hyponymy (subclass). Finally, we would also like to include a theory of context to be able to deal with information about users and to handle series of questions, where later questions may build on earlier ones.

Acknowledgments

Partial funding was provided for this project by DARPA under the EELD program.

References

- Fillmore, Charles J., 1968, The Case for Case. In Emmon Bach and Robert Harms (eds.), Universals in Linguistic Theory. New York, Holt, Rinehart, and Winston, 1968. pp.1-88.
- Harabagiu, Sanda M. and Maiorano, Steven J., 1999, Finding Answers in Large Collections of Texts: Paragraph Indexing + Abductive Inference, AAAI Fall Symposium on Question Answering Systems, November 1999, pp. 63-71.
- Iwanska, Lucja M., 2000, Natural Language is a Powerful Language Representation System: the UNO Model, in Natural Language Processing and Knowledge Representation, Chapter 1, edited by Iwanska and Shapiro, American Association for Artificial Intelligence, 2000.
- Sowa, J. F., 1984. Conceptual Structures, Information Processing in Mind and Machine, Addison-Wesley 1984.