CLR: Integration of FrameNet in a Text Representation System

Ken. Litkowski
CL Research
9208 Gue Road
Damascus, MD 20872
ken@clres.com

Abstract

In SemEval-2007, CL Research participated in the task for Frame Semantic Structure Extraction. Participation in this task was used as the vehicle for efforts to integrate and exploit FrameNet in a comprehensive text processing system. In particular, this involved steps to build a FrameNet dictionary with CL Research's DIMAP dictionary software and to use this dictionary (along with its semantic network processing capabilities) in processing text into XML representations. Implementation of the entire integrated package is only in its initial stages and was used to make only a bare submission of frame identification. On this task, over all texts, a recall of 0.372, a precision of 0.553, and an F-score of 0.445 were achieved. Considering only targets included in the DIMAP FrameNet dictionary, the overall F-score is 0.605. These results, competitive with the top scoring system, support continued attempts at a dictionary-based approach to frame structure extraction.

1 Introduction

CL Research participated in the SemEval-2007 task for Frame Semantic Structure Extraction. In participating in this task, we integrated the use of FrameNet in the Text Parser component of the CL Research Knowledge Management System (KMS). In particular, we created a FrameNet dictionary from the FrameNet databases with the CL Research DIMAP dictionary software and used this dictionary as a lexical resource. This new lexical resource was integrated in the same manner as other lexical resources (including WordNet and the *Oxford Dictionary of English* (ODE, 2004)). As such, the FrameNet dictionary was available as the basis for sense disambiguation. In the CL Research Text Parser, this integration was seamless, in which disambiguation can be performed against several lexical resources. This work attempts to expand on semantic role labeling experiments in Senseval-3 (Litkowski, 2004a, and Litkowski, 2004b).

In the following sections, we first describe the overall structure of the CL Research Knowledge Management System and Text Parser, describing their general parsing and text analysis routines. Next, we describe the creation of the FrameNet dictionary, particularly identifying design considerations to exploit the richness of the FrameNet data. In section 4, we describe our submission for the SemEval task. In section 5, we describe our results. Finally, we identify next steps that can be taken within the CL Research KMS and DIMAP environments to extend the FrameNet data.

2 CL Research Text Processing

The CL Research Knowledge Management System (KMS) is an integrated environment for performing several higher level applications, particularly question answering and summarization. The underlying architecture of KMS relies on an XML representation of texts that captures discourse structure and discourse elements, particularly noun phrases, verbs, and semantic roles (predominantly as reified in prepositions). The texts that are represented include primarily full texts as they may appear in several forms, but also include questions, topic specifications for which summaries are desired, and keyword search expressions.

Text processing is an integrated component of KMS, but for large-scale processing, a separate system, the CL Research Text Parser is frequently used. The same modules are used for both, with different interfaces. Text processing is performed in two stages: (1) syntactic parsing, generating a parse

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 113–116,
Prague, June 2007. ©2007 Association for Computational Linguistics

tree as output; and (2) discourse analysis, analyzing the parse tree and building sets of data used to record information about discourse segments (i.e., clauses), discourse entities (primarily noun phrases, but also including predicate adjective and adverb phrases), verbs, and semantic relations (prepositions). After the data structures are completed for an entire text during the discourse analysis phase, they are used to create a nested XML representation showing all the elements and providing attributes of each component.

The parser is grammar-based and produces a constituent structure, with non-terminals representing syntactic components and leaves corresponding to the words of the sentence. The parser generates some dependency relationships by using dynamic grammar rules added during parsing, particularly through sets of subcategorization patterns associated with verbs (and some other words in the dictionary). This allows the identification of such things as sentence subjects, preposition phrase attachments, and clause attachments. Syntactic ambiguity is handled by carrying forward a variable number of possible parses (usually 40, but user adjustable for any number), eliminating parses that are less well-formed.

The discourse analysis phase includes an anaphora resolution component and detailed semantic analyses of each sentence element. Many dependency relationships are identified during this phase. The semantic analysis includes a disambiguation component for all words (using one or more of the integrated dictionaries). The semantic analysis also identifies (for later use in the XML representation) relations between various sentence elements, particularly identifying the complement and attachment point for prepositions.[1]

To make use of the FrameNet data, it is first necessary to put it into a form that can be used effectively. For this purpose, a DIMAP dictionary is used. Such dictionaries are accessible using btree lookup, so rapid access is ensured during large-scale text processing. Syntactic parsing proceeds at about eight or nine hundred sentences per minute; the discourse analysis phase is roughly the same complexity. The result is that sentences are normally processed at 300 to 500 sentences per minute.

3 A FrameNet Dictionary

The integration of FrameNet into KMS and Text Parser is generally handled in the same way that other dictionaries are used. Specifically, there is a call to a disambiguation component to identify the applicable sense. After this, FrameNet data are used in a slightly different way. Disambiguation proceeds sequentially through the words in a sentence, but the labeling of components with frame elements is performed only after a sentence has been fully discourse-analyzed. This is necessary because the location of frame elements requires full knowledge of all components in a sentence, not just those which precede a given target (i.e., in left-to-right parsing and discourse analysis).

The main issue is the design of a FrameNet dictionary; DIMAP provides sufficient capability to capture all aspects of the FrameNet data (Ruppenhofer, et al., 2006) in various types of built-in data structures. First, it is necessary to capture each lexical unit and to create a distinct sense for each frame in which a lexeme is used. The current FrameNet DIMAP dictionary contains 7575 entries, with many entries having multiple senses.[2] For each sense, the FrameNet part of speech, the definition, the frame name, the ID number, and the definition source (identified as FN or COD, the Concise Oxford Dictionary) are captured from the FrameNet files.[3]

If there is an associated FrameNet lexical entry file that contains frame element realizations, this information is also captured in the appropriate sense. In DIMAP, this is done in an attribute-value feature structure. Each non-empty feature element realization in the FrameNet data is captured. A DIMAP feature attribute is constructed as a conflation of the phrase type and the grammatical function, e.g. “NP (Dep)”. The feature value is a conflation of the valence unit

[1] At present, the analysis of the complement and attachment points examines only the highest ranked attachment point, rather than examining other possibilities (which are frequently identified in parsing).

[2] We unwittingly used an August 2006 version of FrameNet, not the latest version that incorporated frames developed in connection with full-text annotation. This affects our results, as described below.

[3] The FrameNet dictionary data is captured using FrameNet Explorer, a Windows interface for exploring FrameNet frames, available for free download at CL Research (http://www.clres.com).

frame element name and the number of annotations in the FrameNet corpus, e.g., "Cognizer (28)". This manner of capturing FrameNet information is done to facilitate processing; the DIMAP feature structure is frequently used to access information about lexical items. Further experience will assess the utility of this format.

Frames and frame elements are captured in the same dictionary. However, they are not treated as lexical units, but rather as "meta-entries". In the DIMAP dictionary, frame names are entered as dictionary entries beginning with the symbol "#" and frame elements are entered beginning with the symbol "@". In these entries, different data structures of a DIMAP entry are used to capture the different kinds of relations between frames and frame elements (i.e., the frame-to-frame relations) that are found in the FrameNet data. Thus, a frame will have a "frame-element" link to each of its frame elements. It will also have attribute-value features listing its frame elements and their type (core, peripheral, or extra-thematic).

With a dictionary structured as described, it is possible not only to look up a lexical unit, but also to traverse the various links that are reachable from a given entry. Specifically, when a lexical unit is recognized in processing the text, the first step is to retrieve the entry for that item and to use the frame element realization patterns to disambiguate among the senses (if more than one of the same part of speech). After a sentence has been completely processed (as described above), the meta-entries associated with each lexical unit can be examined (and appropriate traversals to other meta-entries can be followed) in order to identify which sentence constituents fill the frame elements.

Specific routines for traversing the various FrameNet links have not yet been developed. However, this is primarily a matter of assessing which traversals would be useful. Similar traversals are used with other lexical resources, such as WordNet, where, for example, inheritance hierarchies and other WordNet relation links are routinely traversed.

4 The SemEval FrameNet Submission

To participate in the SemEval FrameNet task, the three test texts were wrapped into a standard XML representation used in processing texts. This wrapper consists only of an overall **<DOCS>** tag, a subtag **<DOC>** for each document, and a **<TEXT>** tag surrounding the actual text. The text was included with some minor changes. Since Text Parser includes a sentence splitter, we had to make sure that the texts would split into the identifiable sentences as given on each line of the texts. Thus, for headers in the text, we added a period at the end. Once we were sure that the same number of sentences would be recognized, we processed the texts using Text Parser, as described in section 2.[4]

As mentioned above, the FrameNet dictionary lookup occurred in a separate traversal of the parse tree after the discourse analysis phase. During this traversal, the base form of each noun, verb, adjective, or adverb content word was looked up in the FrameNet dictionary. If there was no entry for the word, no further FrameNet processing was performed. When an entry was found, each sense of the appropriate part of speech is examined in order to disambiguate among multiple senses. A score is computed for each sense and the score with the highest sense was selected.[5]

Having identified a sense in the FrameNet dictionary, this was interpreted as finding a FrameNet target, with the FrameNet frame as identified in the lexical entry. Since the character positions of each word in the source sentence are included in the parse tree information, this information was captured for inclusion in the output. (Further implementation to identify the frame elements associated with the target has not been completed at this time. As a result, our submission was only a partial completion of the FrameNet task.)

After completing the processing of each sentence,

[4]To make a submission for the FrameNet task, it was necessary to initialize an XML object into which the results could be inserted after processing each sentence. This is not a usual component of Text Parser, but was implemented solely for the purpose of participating in this task.

[5]At this time, all senses receive an identical score. The first sense is selected. Senses are unsystematically ordered as they were encountered in creating the FrameNet dictionary. This will be extended to compute a score based on the various frame element realization patterns associated with each sense.

all FrameNet frame information that had been identified was processed for inclusion in the XML submission for this task. In particular, the annotation sets required were incorporated into the XML object that had been initialized. (Our annotation sets included only the "Target" layer.) After all sentences had been completed, the XML object was printed to a file for submission.

5 Results

Our results are shown in Table 1, giving the recall, precision, and F-score for each text and over all texts. As indicated, these results are for only the target identification subtask.[6]

Table 1. Target Identification Scores			
Text	**Recall**	**Precision**	**F-Score**
Dublin	0.33403	0.53572	0.41237
China	0.51148	0.52525	0.51827
Iran	0.44828	0.66102	0.53425
All	0.37240	0.55337	0.44520

As indicated above, we used an early version of the FrameNet databases that did not include all the lexical units in the training and test texts. As a result, we did not have FrameNet entries for 30 percent of the words identified as targets in the test texts. Table 2 shows an estimate of the adjusted scores that would result if those lexical items were included..

Table 2. Adjusted Target Identification Scores			
Text	**Recall**	**Precision**	**F-Score**
Dublin	0.53445	0.65140	0.58716
China	0.57037	0.62097	0.59459
Iran	0.61494	0.72789	0.66667
All	0.56144	0.65132	0.60305

The results in Table 1 rank third of the four teams participating in this subtask. With the results in Table 2, our performance would improve to first for two of the texts and just below the top team for the other text.

[6]Corresponding to the "-e -n -t' options of the scoring program. In these tables, "Dublin" refers to **IntroOfDublin**, "China" to **ChinaOverview**, and "Iran" to **workAdvances**.

6 Future Steps

Participation in the FrameNet frame structure extraction task has demonstrated the basic viability of our approach. Many of the frames have been recognized successfully. We have not yet examined the extent to which the disambiguation among frames is significant, particularly since there are not many entries that have several senses. We have yet to develop specific techniques for making use of the frame element realization patterns. However, we believe that a reasonable performance can be expected since KMS and Text Parser produce output that breaks sentences down into the types of components that should be included as frame elements.

The architecture of KMS, Text Parser, and DIMAP provide significant opportunities for extending our performance. In particular, since these systems include the *Oxford Dictionary of English*, a superset of the *Concise Oxford Dictionary*, there is an opportunity for extending the FrameNet datasets. The COD definitions in FrameNet can be mapped to those in ODE and can be exploited to extend FrameNet frames to lexical items not yet covered in FrameNet.

References

Kenneth C. Litkowski. 2004a. Senseval-3 Task: Automatic Labeling of Semantic Roles. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text.* Association for Computational Linguistics. 9-12.

Kenneth C. Litkowski. 2004b. Explorations in Disambiguation Using XML Text Representation. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text.* Association for Computational Linguistics. 141-146.

The Oxford Dictionary of English. 2003. (A. Stevension and C. Soanes, Eds.). Oxford: Clarendon Press.

Josef Ruppenhofer, Michael Ellsworth, Miriam Petruck, Christopher Johnson, and Jan Scheffxzyk. 2006. FrameNet II: Extended Theory and Practice. International Computer Science Institute, University of California at Berkeley.