The Role of Syntactic Analysis in Textual Case Retrieval

Bradford Mott[1], James Lester[1], and Karl Branting[2]

[1] LiveWire Logic, Inc.
Morrisville, NC, USA
{lester,mott}@livewirelogic.com
[2] BAE Systems
Columbia, MD, USA
karl.branting@baesystems.com

Abstract. In this paper, we argue that syntactic analysis is most likely to improve retrieval accuracy in textual case-based reasoning when the task of the system is well-defined and the relationship between queries and cases is specified in terms of this task. We illustrate this claim with an implemented system for syntax-based answer-indexed retrieval, RealDialog.

1 Introduction

A critical step in case-based reasoning is retrieval of cases relevant to the current problem. In textual case-based reasoning, (hereinafter *TCBR*), cases often consist of semi-structured or unstructured text. The solution to the problem may consist of the documents themselves, as in FAQ or recommender systems [Bur99], or may instead be derived from one or more retrieved documents through some adaptation process, as in question-answering systems [SSW$^+$98].

The input to the retrieval step of TCBR consists of a probe (e.g., a problem description or question) and a collection of cases. However, there can be considerable variability in the character of both probes and cases.[3]

This paper argues that the retrieval requirements of TCBR depend on the nature of task addressed by the TCBR system and on the relationship between the probe and the cases. Specifically, syntactic analysis of the probe and cases is most likely to improve retrieval accuracy beyond what can be achieved through term-vector retrieval when the task of the TCBR system is precisely specified and the relationship between probes and cases is specified in terms of this task.

The next section describes the role of text retrieval in TCBR, and Section 3 illustrates the role of syntactic analysis in retrieval in RealDialog, a web-based conversational agent system for enterprise knowledge management.

[3] In this paper, we assume that probes, regardless of their character, consist of meaningful text. We exclude probes that are not amenable to syntactic analysis, such as term vectors or key-word lists.

2 Text Retrieval in TCBR

Since both probes and cases typically consist of text in TCBR, the retrieval step of TCBR is an instance of the larger problem of retrieval of text documents based on a text probe that has been studied by the information retrieval, (*IR*), community for decades. Mainstream IR typically makes no assumption concerning the purposes for which the retrieval is being performed. Instead, the probe represents the lexical characteristics of the desired document itself rather than a specification of the problem for which the desired case is a solution.

IR researchers have long had the intuition that analysis of the syntactic structure would improve retrieval. This intuition is based on the plausible assumption that the meaning of texts is more accurately reflected by the syntactic structure of those texts than by their representations as term-vectors.

Unfortunately, efforts to demonstrate the effectiveness of syntactic structures in retrieval have a long history of failure [Fag87,CD90,SOK94]. The primary focus of syntactic analysis in IR has been on *phrase identification*, using term sequences to distinguish among documents. Phrases may be either syntactic units, such as noun phrases, or sequences based on co-occurrence statistics [LJ96]. Phrase identification has been shown to improve retrieval accuracy when combined with statistical techniques [CTL91], but there is little evidence that more complex syntactic analysis improves the performance of generic IR.

We hypothesize that the reason that syntactic analysis has not been shown to contribute significantly to generic IR is that in generic IR both the task for which the retrieval is being performed and the relationship between the probe and the cases are unspecified. In the absence of any knowledge about what aspects of probes and cases are relevant to the underlying problem, the importance of syntactic similarities and differences is difficult to assess. In contrast, if the task is known and the relationship between the probe and cases is specified in terms of this task, as is typically the case in TCBR, then syntactic similarities and differences between a probe and cases are likely to be much more discriminating. For example, if the probe is a question and cases represent answers, then the probe and case must stand in a question-answer relationship, and syntactic information bearing on this relationship discriminates well between relevant and irrelevant cases.

Two additional factors may contribute to the greater effectiveness of syntactic analysis for TCBR than for conventional IR. First, TCBR typically has higher precision requirements than generic IR. In generic IR, precision is often less important than recall because the user typically has an opportunity to make an independent judgment about document relevance. In a typically CBR system (although not in many conversational CBR systems [DA01] case adaptation and reuse are performed by the system. Depending on the amount of adaptation knowledge available, system performance may depend critically on the quality of the retrieved cases. A second factor is that parsing technology has dramatically improved in recent years. Efficient and accurate techniques now exist for POS tagging [Bri95,Rat96] and chunking [SB00], and lexicalized probabilistic context free grammars produce much more accurate parses of unrestricted text than past parsers [Col03,Cha01].

We illustrate our hypothesis that syntactic analysis leads to improved retrieval accuracy when the task and the relationship between probes and cases is well-specified with two examples: open-domain question answering; and syntax-based answer-indexed text retrieval.

In open-domain question answering, the task is to find fact-based, short-answers to questions from any domain. Such answers are sometimes referred to as "factoids." NIST has organized question-answering competitions since 1999.[4] The factoid task is quite specific: find passages in a document that answer the question expressed in the probe. The relationship between the probe and the document is itself therefore constrained by the question-answer relationship.

Although competitors in the question-answering competitions have employed a wide variety of different techniques, retrieval typically is performed in two steps. High-recall generic IR techniques are used retrieve a set of candidate cases (document segments), often using thesauri (such as WordNet) to expand the set of terms in the question to include synonyms. The question is then analyzed to determine the kind of entity being sought (e.g., a date in "When was the French revolution?"; a person in "Who invented the transistor?"). Each of the candidates is then evaluated to determine whether it provides the information sought in the question. All successful question-answering competitions syntactically analyze both the question and potential answer text. One of the most successful question-answering systems uses theorem-proving techniques to confirm that a candidate text does in fact answer the question [MCHMCL].

The next section describes the role of syntactic analysis in our work in syntax-based answer-indexed text retrieval.

3 Syntax-Based Question-Indexed Case Retrieval

In closed-domain question answering, the domain of knowledge is circumscribed. An important form of closed-domain question answering is *question-indexed case retrieval*, the TCBR task in which a question is specified by the user and the most similar question in a question base must be identified and its associated response returned [BHK$^+$97]. Syntactic analysis plays a central role in question-indexed case retrieval. For applications in which precision is an important metric, i.e., applications in which false positives are highly penalized, syntactic analyses are essential to the success of the matching phase. Because of the discriminating power of syntactic analysis, syntax-informed question-indexed case retrieval techniques enable a system to successfully distinguish semantically different cases from one another, even though they are highly similar lexically. Simple term-vector techniques are not sufficiently powerful to make such judgments successfully.

Syntax-based analysis contributes to the successful analysis of utterances and a question retriever's ability to identify the most appropriate cases, and it is central in a number of commercial applications. Syntax-directed question retrieval

[4] See the NIST TREC homepage at http://trec.nist.gov/.

is implemented in RealDialog^{TM}, a web-based conversational agent system we and our colleagues have developed for enterprise knowledge management. RealDialog's interface is shown in Figure 1. Users type queries into a text field, and answers are displayed in a conversation area. Optionally, additional information can be displayed in a web-display panel.

Given a question posed by a user, RealDialog performs a syntactic analysis on the query. The full details of syntactic analysis in RealDialog are beyond the scope of this paper. However, the basic steps are as follows. The first step is tokenization of the user's statement, that is, division of the input in a series of distinct lexical entities. Tokenization includes spell-correction and interpretation of apostrophes. The second step is syntactic analysis. In RealDialog, this consists of part-of-speech tagging and parsing. The result of the tokenization, tagging, and parsing is a parse tree [LBM04]. The resulting syntactic structure is then used to retrieve the textual answer contained in the case with the question which is most similar to the question posed. Retrieval is performed with a fuzzy matching technique that takes into account syntactic information in the question component of each case, together with lexical and synonymy information that is relevant to the query and the case questions.

To illustrate, suppose the user asks, "How can I change my hard drive format from FAT to NTFS?" in a hypothetical tech support application. The system indexes into the case library to find a case whose question component is, "Can you explain how I convert my drive from FAT format to NTFS?" It then provides a procedure walking the user through the conversion process.

A critical feature of question-indexed case retrieval is its precision: it avoids false positives stemming from probes and case questions that are lexically similar but semantically different. To illustrate, consider the following pairs of questions:

```
(U1-a)  How can I change my hard drive format from FAT to NTFS?

(U1-b)  How can I change my hard drive format from NTFS to FAT?
```

Utterances U1-a and U1-b are lexically identical: both have exactly the same term-vector representation. However, the semantics of the two requests are fundamentally different. In U1-a, the user requests a change from FAT format to NTFS format, while the request in U1-b is to make the change from NTFS to FAT. The correct response to U1-a is a description of the steps required to change the drive's format, while the correct response to U1-b is a suggestion to reconsider the request because of the implausibility arising from FAT's limited functionality relative to NTFS. The prepositional phrases in the two requests are reversed, but a term-vector approach would be unable to identify the differences.

Next, consider next the following pair:

```
(U2-a) How do I make the client authenticate the server?

(U2-b) How do I make the server authenticate the client?
```

Utterance U2-a asks how one can have the client authenticate the server, but U2-b asks how one can have the server authenticate the client. Semantically, the utterances are opposites of one another, but again, they are lexically indistinguishable. Syntax must be called into action to determine the agent and object roles played in each utterance to correctly ascertain the intended meaning.

Next consider the following:

```
(U3-a)  How do I replace the CPU that is beside the DIMM?

(U3-b)  How do I replace the DIMM that is beside the CPU?
```

Utterance U3-a asks about the replacement of the CPU while U3-b asks about the replacement of the DIMM. Syntactically, the CPU serves as the verb's object in U3-a, and DIMM is the object of the preposition. In contrast, in U3-b DIMM is the verb's object and CPU is the prepositional object. The utterances' bags of words are the same, but the meaning of the utterances is very different. Without a syntax analysis they would have appeared identical but parsing enables them to be correctly analyzed.

The question-indexed case retrieval task illustrated above is required in a broad family of commercial question answering applications. RealDialog has been deployed at a number of companies including two Fortune 500 firms. Its applications have included outward-facing deployments in which it is available to users visiting business' web sites and inward-facing deployments in which it is used by customer service representatives and retail store associates to help find the answers to users' questions more efficiently.

4 Conclusion

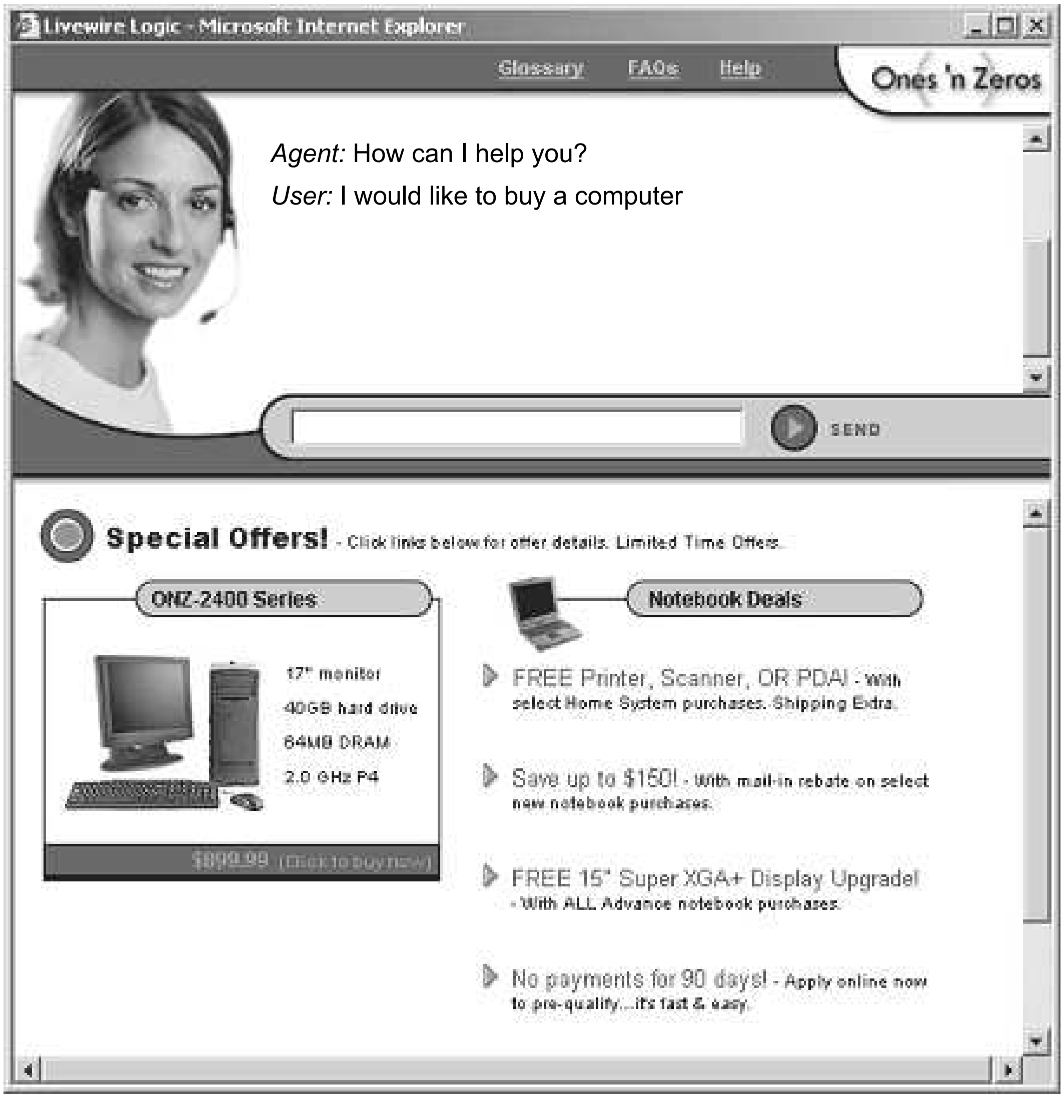
We have argued that syntactic analysis of the probe and cases is most likely to improve retrieval accuracy beyond what can be achieved through term-vector retrieval when the task of the TCBR system is precisely specified and the relationship between probes and cases is specified in terms of this task. We illustrated this claim with RealDialog, an implemented commercial system for question-indexed case retrieval in which syntactic analysis is essential for accurate performance.

References

[BHK$^+$97] R. Burke, K. Hammond, V. Kulyukin, S. Lytinen, N. Tomuro, and S. Schoenberg. Question answering from frequently-asked question files: Experiences with the FAQ finder system. Technical Report TR-97-05, University of Chicago, Department of Computer Science, 1997.

[Bri95] E. Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565, 1995.

[Bur99] R. Burke. The wasabi personal shopper: A case-based recommender system. In *Proceedings of the 11th National Conference on Innovative Applications of Artificial Intelligence*, pages 844–849. AAAI Press, 1999.



Agent: How can I help you?

User: I would like to buy a computer

Special Offers! - Click links below for offer details. Limited Time Offers.

- FREE Printer, Scanner, OR PDA! - With select Home System purchases. Shipping Extra.
- Save up to $150! - With mail-in rebate on select new notebook purchases.
- FREE 15" Super XGA+ Display Upgrade! - With ALL Advance notebook purchases.
- No payments for 90 days! - Apply online now to pre-qualify...it's fast & easy.

Fig. 1. The RealDialog interface.

[CD90] W.B. Croft and R. Das. Experiments with query acquisition and use in document retrieval systems. In J. Vidick, editor, *Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 349–368. ACM Press, 1990.

[Cha01] Eugene Charniak. Immediate-head parsing for language models. In *Meeting of the Association for Computational Linguistics*, pages 116–123, 2001.

[Col03] M. Collins. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637, 2003.

[CTL91] B. Croft, H. Turtle, and D. Lewis. The use of phrases and structured queries in information retrieval. In *Proceedings of SIGIR'91*, pages 32–45, 1991.

[DA01] H. Muoz-Avila D. Aha, L. Breslow. Conversational case-based reasoning. *Applied Intelligence*, 14(1):9–32, 2001.

[Fag87] J. Fagan. *Experiments in Automatic Phrase Indexing For Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods*. PhD thesis, Cornell University, Computer Science Department, 1987. Technical Report 87-868.

[LBM04] James Lester, Karl Branting, and Brad Mott. Conversational agents. In Munindar Singh, editor, *Practical Handbook of Internet Computing*. CRC Press, 2004.

[LJ96] David D. Lewis and Karen Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 39(1):92–101, 1996.

[MCHMCL] D. Moldovan, C. Clark, S. Harabagiu, and S. Maiorano. Cogex: A logic prover for question answering. In *Proceedings of HLT-NAACL*, pages 87–93, HLT-NAACL.

[Rat96] A. Ratnaparkhi. A maximum entropy part-of-speech tagger. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*. University of Pennsylvania, May 17–18 1996.

[SB00] Erik Tjong Kim Sang and Sabine Buchholz. Introduction to the CoNLL-200 shared task: Chunking. In Claire Cardie, Walter Daelemans, Claire Nédellec, and Erik Tjong Kim Sang, editors, *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop, Lisbon, 2000*, pages 127–132. Association for Computational Linguistics, Somerset, New Jersey, 2000.

[SOK94] Alan F. Smeaton, Ruairi O'Donnell, and Fergus Kelledy. Indexing structures derived from syntax in TREC-3: System description. In *Text REtrieval Conference*, 1994.

[SSW$^+$98] Tomek Strzalkowski, Gees C. Stein, G. Bowden Wise, Jose Perez Carballo, Pasi Tapanainen, Timo Jarvinen, Atro Voutilainen, and Jussi Karlgren. Natural language information retrieval: TREC-7 report. In *Text REtrieval Conference*, pages 164–173, 1998.