Distributed at 1987 ACL Meeting, Stanford, and at DARPA Workshop on Evaluation, Wayne, Pennsylvania, 1989.

Toward Evaluation of NLP Systems

Dan Flickinger, John Nerbonne, Ivan Sag and Tom Wasow
Hewlett-Packard Laboratories,
Palo Alto, CA

Nov 30, 1989

1 Overview

At Hewlett-Packard Laboratories we have begun an effort to evaluate natural language processing systems, and we'd like to describe that effort to you here. To "evaluate NLP systems" means here to articulate, standardize and partially automate the presently very soft art of assessing the quality of work in natural language processing. In rough outline, we have carefully constructed a test suite of English sentences and we regularly monitor how well our own system does in processing them. Before describing this work in detail, we should emphasize that our evaluation effort is still work in progress. Far from regarding it at as a finished tool, we'd like to solicit your help in extending and polishing it.

2 Uses of an Evaluation Facility

Before examining the evaluation work we have done, it will be worthwhile reviewing the uses to which we hope to put this work. We certainly do not imagine that we'll ever plug a random NLP system into an evaluation tool, and then read out its grades: 78% correctness in lexical analysis, 24% in conversational cooperativeness, etc.

Evaluation will always be hindered by the surprising lack of common assumptions that are brought to NLP efforts, and by the increasingly wide range of uses to which NLP is put. To illustrate these points in turn, consider first how problematic it would be to rate the syntactic accuracy of a semantically driven parser, whose major purpose is to avoid many semantically anomalous, but syntactically correct parses. The second point, that different applications may require different standards, is perhaps well illustrated by comparing the needs of a style checker versus that of a specialized database lookup application, say a library catalogue. The style checker may do quite well by concentrating on those syntactic rules which differ dialectally or across registers or style levels, while the library application will certainly ignore such niceties.

To the extent to which assumptions are shared and similar applications are sought, however, more detailed comparisons become possible. A fair number of systems do aim at the same application, viz. database

query interface. Some assumptions are also shared: A good number of NLP efforts do distinguish syntax and semantics, and futhermore distinguish domain-independent semantics from application interpretations, allowing comparisons at each of these levels.

The limiting case in which most of the assumptions and intended applications are kept constant is the comparison of a single effort over time, and comparison here is relatively unproblematic. We have used an evaluation facility regularly over the last two and a half years for several reasons:

- to spot bugs
- to assess proposed modifications
- to monitor our own progress

None of these tasks are performed well by unaided humans, particularly not where systems are large and efforts cooperative. In larger efforts and systems there isn't any one individual capable of tracking bugs, unintended side-effects, and overall progress. The effort is simply too large and complex. In these systems, such as our own, an automatic evaluation effort quickly proves its worth.

3 Answering Objections

Before describing our evaluation facility in detail, let us anticipate the common objections to proposals such as this. Indeed some of these were our own objections when a senior manager associated with HP Labs's NL work first suggested an evaluation tool to us.

To begin, we dismiss the sophomoric objection that no finite set of sentences could provide a real test for a grammar or an NLP system. This objection seems to rest on the mathematically correct point that one could always provide *ad hoc* analyses for finite lists of data. We are not interested in ferreting out fraud at this level, however. Perhaps more to the sense of this objection, let us add that our interest doesn't lie in the particular *tokens* in the evaluation suite at all, but rather in the grammatical *types* they exemplify. Anyone who claimed to process just those tokens wouldn't engage any researcher's interest nor would it fruitfully occupy our evaluation facility.

The use of a fixed set of sentences to test an NLP system sounds a good deal like a methodology once common in linguistics, which generative linguists have warned against (Chomsky, 1957:15). This methodology involved collecting a "corpus" of utterances and then subjecting it to grammatical analysis. Linguists point to two dangers here: first, the corpus is inevitably smaller than the target set of sentences and potentially skewed; and second, sentence corpora contain sentences, but not ungrammatical strings, while humans have not only the ability to parse sentences, but also the ability to recognize strings as ill-formed. The latter ability is neglected if one only analyzes a corpus of (grammatical) sentences.

Neither of these dangers need be realized in a well-constructed evaluation tool. The data set may originally be skewed, but bias can be recognized and corrected, whether through criticism or through data from actual use. And the data may include ill-formed strings, the importance of which we return to below.

There are two further objections to our proposal for an NLP evaluation tool: first, that it concentrates on problems in syntax and semantics, which are regarded as "solved problems" in some circles; and second, that evaluation is much too large a problem.

The first objection is simply uninformed–there are dozens of unsolved problems in syntax and semantics, and these impede progress in building useful systems even now. Let coordination stand as one good example of these. It is also worth mentioning that we do not intend to evaluate only syntactic and semantic analysis, but also discourse analysis, some sorts of reasoning, and eventually much more.

To the second suspicion that evaluation is too difficult, we can best point to benefits we have received from a very imperfect evaluation tool. It may be too difficult to build a perfect evaluator, but it may also be unnecessary.

4 Desiderata for Construction of the Evaluation Facility

We turn now to the evaluation facility itself. It consists of a large set of English sentences annotated by construction type and code which asks the NLP system to evaluate the sentences against a database.

In constructing the suite of sentences we aimed at a level of style that is common to informal typewritten communication. We therefore avoided regionalisms and colloquialisms. We expect the normal (useful) unit of interaction between people and NLP systems to be a discourse, so that we include intersentential dependencies. For the same reason we might have attempted to assess sensitivity to purpose, inference and domain knowledge, but we weren't clear about how to do this.

A number of considerations affected our choices in selecting the sentences that appear in our evaluation suite.

Foremost among these was the desire to provide coverage of a wide variety of syntactic and semantic phenomena; we also made an effort to give some attention to discourse-level phenomena, especially those concerned with anaphora. Our suite is informed by our study of linguistics and reflects the grammatical issues that linguists have concerned themselves with over the past thirty years. Since the purpose was to get a reasonably comprehensive list of sentences, little effort was devoted to rationalizing the taxonomy employed. The classification employs a level of granularity that seems intuitive–in terms of constructions like "passive" or "relative clause"–without making any claims about theoretical significance of the categories (since we wished to avoid theory dependence).

In testing for coverage, we included not only well-formed sentences, but also anomalous strings. Although it is clearly desirable to be as tolerant of user mistakes as is practical, it is more important for the system to correctly interpret well-formed input. It has been our experience that failure to attend to grammatical details almost always results in introducing spurious ambiguities. For example, subject-verb agreement in English might seem like something that can be ignored, as the information it conveys (viz., the number of the subject) is already present. In complex sentences, however, agreement can function to determine which of two noun phrases is the subject of a clause, with attendant significant effects on the meaning. Thus, a system that failed to do grammatical agreement would not be able to distinguish the meanings of the following two sentences:

Grammar is Essential to Language Understanding

Subject-Verb Agreement [He understands/*understand English.]

Is there a newsman with Iranian contacts who work/works for the CIA?

The sources said that Reagan and Regan, who has/have been under heavy pressure to resign, have reached an agreement on the time of [the] departure. [S.F. Chronicle - Feb. 26, 1987]

Reflexive Pronoun Agreement [She gave herself/*themselves a raise.]

Who is the manager of the project leaders who gave herself/themselves a raise?

Relative Pronoun Agreement [I met a manager who/*which is working at the lab.]

List all members of the projects who/which are working at the lab.

Pronoun Case Marking [I/*me like him. / I like him/*he.]

The man who knew he/him found the witness lied.

Which gardener who knew he/him sold plant food bought dirt cheap?

Deletion of Subject Relative Pronouns

[I joined the project (which) Jones directs. / I met the woman *(who) was the project director.]

Did Jones know the woman (who) was the project director?

Commas

They questioned the secretary (,) and the president (,) then resigned.

Complementizer Omission [Who do you think (*that) left?]

Which employee did you believe (that) was working for Jones?

Conjunction Placement Rules [*Kim and Sandy, Lou left. / Kim, Sandy and Lou left.]

Kim walked in, picked up the proposal and passed by the dean's office.

Kim walked in, and picked up the proposal passed by the dean's office.

Figure 1: Sag's Examples of Grammar Influencing Interpretation

Is there a newsman with Iranian contacts who works for the CIA?
Is there a newsman with Iranian contacts who work for the CIA?

Our evaluation suite includes ungrammatical examples in order to check for attention to such grammatical details. Including these is essential for as Sag (in preparation) shows, even the most exotic grammatical distinctions play an important role in natural language understanding and generation. The general point is that a system that fails to detect ungrammaticality will introduce ambiguity and therefore interpret incorrectly. Examples of this *pernicious dysfunction* (taken from Sag, in preparation) are given in Figure 1.

Another goal was to make use of a limited vocabulary. The use of a limited vocabulary means that some perfectly natural constructions are exemplified only in sentences which sound contrived. Nonetheless, three factors legislated for limiting vocabulary size. First, providing good syntactic/semantic coverage is not dependent on having a large vocabulary; our suite is aimed at evaluating coverage, not vocabulary size. Second, part of our effort has been to build a database onto which our test sentences can be targeted, in the hopes of obtaining answers. The larger the vocabulary of the suite, the larger and more complex the database interface required. Third, a smaller vocabulary makes for a smaller and more portable system.

A final goal was to test the semantic correctness of processing against a database answer. The ability to do this is crucial if one is to effectively compare systems that are semantically driven, or which make use of nonstandard semantic representations. It is also necessary if any effective comparison between systems

is to be automated, since one would otherwise need a test for equivalence in semantic representations. But the need to test semantics against a database required a good deal of work in database construction, and moreover influenced our choice of constructions–some are easier than others to represent, and indeed some have resisted satisfactory representation (modalities).

The suite of sentences is organized in a file with annotations about which constructions they serve to exemplify. No attempt was made to annotate exhaustively, and it is clear that every sentence exemplifies more than one linguistic property.

5 Measures of Success

A fairly small amount of code runs the suite of sentences, checks the results at various stages of processing, and queries the database to determine an answer.

Several measures of success are available:

- accuracy of lexical analysis
- accuracy of parsing
- accuracy of domain-independent semantics
- correctness of database query generated
- correctness (and appropriateness) of answer

Since the test suite contains both well-formed and ill-formed input, the success rate measures how well both are recognized: an error can be either a well-formed sentence that isn't analyzed correctly (a false negative), or an ill-formed string that is analyzed (a false positive). It is clear that some applications will not care about the latter figure, but some just as clearly must care (style checkers). Furthermore, we distrust systems that fail to recognize ill-formedness in their haste to cooperate, and we believe that these systems will be guilty of the sort of misanalysis Sag's examples point up.

This leads to an important remark: **Not every measure of success is applicable to every system.** So for example, systems eschewing a level of domain-independent semantics obviously can't be measured on that scale. Similarly, some of the measures above are influenced by theory, e.g. parsing. Theories differ, even greatly, in the analysis trees they assign in parsing.

Two remarks are appropriate here. First, to the extent to which theories and NLP systems do share assumptions, meaningful comparisons are possible. Since many systems do provide a level of domain-independent semantics, comparisons between them are interesting. Second, even theories and systems with differing assumptions do concur in many points. For example, theories might assign different analysis trees, but they would all assign more than one to structurally ambiguous sentences. Even systems which are semantically driven ought to display this capability (or fall afoul of the *pernicious dysfunction* illustrated above) even if they would display the two parses only in different circumstances.

6 Prospectus and Invitation

We have mentioned that we use the test suite regularly ourselves, and find it valuable in internal monitoring. We offer it to others in the hope: first, that it may be of more general utility; and second, that the suite will benefit from extension and criticism. (Animadversions may be emailed to: hplnls@hplabs.hp.com.)

We would like to extend the suite in several ways: we have a catalogue of English constructions (and construction combinations) many of which are now exemplified in the test suite, and many of which need to be filled in; we would also like to have someone test the suite against a standard grammar such as Quirk *et al.* or *The Oxford Advanced Learner's.* Furthermore, it would obviously be helpful to understand the relation between the sentences in the test suite, and those actually input to processing systems in use.

Beyond the coverage of syntax and semantics, many gaps need to be filled. The behavior of pronouns in discourse is represented, but not that of other elements (notably tense). We haven't attempted to build in any measurement of system sensitivity to user purpose or speech act type, and we haven't found a way to a test a system's ability to resolve ambiguity, or to interpret the contextual specification of vagueness or generality.

There are further measurements of system performance that lie beyond range and accuracy of linguistic coverage. These include compatibility, modifiability, good software "citizenship", as well as size, speed, and installation and maintenance costs (including time and difficulty).

Finally, we are interested in the establishment of a general purpose evaluation system for NLP, i.e. one which could be used to compare a variety of systems. This would require not only the above work, but also the establishment of a database which could be regarded as standard. Certainly, the present *de facto* standard is that the database be accessible through the SQL query language, and we would favor the decision to use it, but the adoption of SQL as a database interface brings with it several issues connected with reasoning capabilities and NLP systems. Briefly, SQL offers less reasoning ability than NLP systems ought to have so that a commitment to the use of SQL is an admission that NLP systems must have their own reasoning capabilities–a potentially distorting factor. The use of more sophisticated database access facilities, which is the likely development path for NLP systems, is complicated by the absence of a standard here.

References

Chomsky, Noam. 1957. *Syntactic Structures.* The Hague: Mouton.

Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech, and Jan Svartik. 1985. *A Comprehensive Grammar of the English Language.* London: Longman.

Sag, Ivan. In preparation. Why Artificial Intelligence Needs Linguistics. Ms., Stanford University.