Introduction to the Research Direction

Research Memo #1 [1]
Iddo Lev
November 24th, 2005

Preface

The description here provides the background, motivation, and the general idea of the research direction. Spelling out more details will be done in future research memos. This research direction is still too broad for the scope of a PhD dissertation, so at the end I briefly outline what part of it will be taken up as the dissertation work.

1 Understanding

In my opinion, the ultimate goal of NLP is to create a computer that really *understands the meaning and information* conveyed by a Natural Language (NL) text or speech. From a practical point of view, we would like computers to read and understand information in texts so that they could answer correctly questions about that information.[2] From a scientific perspective, if a computer is able to understand a class of texts correctly, knowing how it does it might shed some interesting light on how humans do it.

We need to define what is meant by "understanding the meaning and information in a text." We take the operationalized definition: the ability to draw inferences from the text and answer questions about it in the same way that humans would given the same input. Since humans do not always agree about the inferences and answers to questions from a given NL text, we concentrate our research here on those cases where most people do agree, cases that are quite clear cut.

2 Knowledge

2.1 Kinds of Knowledge

Understanding, as defined above, requires the computer to possess and utilize a lot of knowledge of various kinds:

1. Factual knowledge: facts about concept-instances in the world, e.g.: Abraham Lincoln was the president of the United States between the years 1861 and 1865.

2. Conceptual knowledge: concepts and their inter-relationships (e.g. as captured in ontologies).

[1] I intend this to be the first in a series of short essays that explain the vision and details of my research. The essays are far from perfect, but they're better than having no essays, and they can support discussion of my work.

[2] For simplicity, I talk here about NL texts, but the ideas carry over (perhaps with some required changes) to other forms of NL input.

(a) World knowledge and commonsense knowledge: Such information is rarely explicitly expressed in texts because texts are designed for human communication and so are written with the assumption that the reader has this knowledge.

(b) Topic-specific knowledge: For example, if the text describes the malfunctioning of a car, the computer needs to have knowledge about car operation and diagnostics to be able to explain the problem and suggest useful repairs.

3. Knowledge of language

   (a) Lexical (semantic) knowledge: how lexical items and their arguments (e.g. verb subcategorization frames) map to the conceptual knowledge.

   (b) Structural knowledge: morphology, syntax, structural semantics, discourse structure, etc.

The knowledge has two "dimensions" or "modalities": (a) qualitative or content-knowledge, and (b) quantitative or statistical knowledge. What the quantitative modality adds is information about the frequency of certain phenomena or pieces of knowledge in the world, or language, or in particular texts of interest. For example, knowledge of language includes what subcategorization frames each verb can appear with and how each of these is related to the conceptual knowledge, and the quantitative dimension adds information about how prevalent each of these frames is in language use.

2.2 Issues

There are three main issues regarding knowledge:

1. Representation: How do we represent the knowledge. What formal languages do we use and what do the representations mean.

2. Computation and Reasoning: How do we calculate the representations, and how do we use them to support reasoning and inference (on various levels).

3. Acquisition: How do we acquire the knowledge. What are the methodologies for investigating relevant phenomena in order to develop the knowledge frameworks, and how can the qualitative and quantitative information be manually, semi-automatically, and automatically acquired.

2.3 Combinations and Continuums

It is impossible today to supply a computer with all this knowledge, since its quantity is hugely enormous (and it is an AI-complete task – if a computer had all this knowledge, it would be able to do anything a human can). Clearly, each project needs to concentrate on a small subset of this knowledge (hopefully, there is a gradual cumulative progress in the coverage of this knowledge). There are several possible ways to carve out a chunk to work on.

One continuum of choices pertains to whether we allow input texts on arbitrary topics, on the one end of the scale, or texts drawn only from a particular topic or domain, on the other end of the scale. Since NL in general can talk about any topic, if we want a computer to understand texts on arbitrary topics, it would need to possess vast amounts of world knowledge and topic-specific knowledge. As this is impossible today, allowing arbitrary topics necessarily implies a very rudimentary level of understanding of the information conveyed in the texts. In contrast, restricting the input texts to a particular topic or domain allows for a much higher level of understanding of the information conveyed in the texts since it is possible to map out a much larger percentage of the domain-specific and general world knowledge necessary for understanding in the domain.[3]

Another continuum pertains to the degree of precision versus "robustness" in the language coverage and comprehension. On the one extreme end of the scale, there are applications that require a very precise level of linguistic analysis in order to get at the meaning of the text correctly. Such applications include understanding technical texts written in a controlled subset of NL, such as those used by the aero-space industry (see more in section 6). These applications actually require the computer to reject NL inputs that fall outside the scope of the defined grammar. The high precision is usually achieved by using hand-written grammars and knowledge. On the other extreme end of the scale, there are applications that are defined to not put any restrictions whatsoever on the NL input, and they are supposed to always produce some answer. In cases that are similar enough to what the system has been trained on, it should be able to produce a quite correct analysis with high confidence, and in other cases, it should produce the best guess it can based on what it knows. The robustness is usually achieved by using statistical models that are trained on a lot of corpus data and are thus able to adapt themselves as best they can to the variety of inputs they might encounter.[4]

These two continuums are often correlated together in what can be characterized roughly as having deep and precise understanding in a narrow domain at one end of the scale versus shallow or imprecise processing of arbitrary texts on arbitrary topics at the other end.

There is some work that aims to reach some point closer to the middle of this continuum and combine the advantages of both sides. Starting from the "broad but shallow" end, there has been a gradual increase in the sophistication of probabilistic models and in the quality and quantity of knowledge they utilize. For example, whereas early models of syntax used Markov chains, later models learned to assign probabilities on context-free grammar rules, and still more recent models use more sophisticated representations that do not suffer from the naive independence assump-

[3]Restricting attention to texts in a specific domain does not necessarily mean restricting attention to some domain-specific subset of NL. The language could still be general in the sense that all linguistic phenomena on all levels (morphological, syntactic, semantic, etc.) could still be used.

[4]Another name for "robustness" could be "input tolerance" or "any text". Sometimes the term "broad coverage" is used for this purpose, but I think it is not accurate enough, since "coverage" might imply at least some understanding of the covered phenomena, which is not really possible today on arbitrary texts. Another term that is sometimes used in the context of statistical parsers is "parse-anything", but I think it is also inaccurate because such parsers often produce an output that is a wrong parse or not even a valid parse.

tions of PCFGs, and which take into account information about lexical items [add references]. Starting from the "deep but narrow" end, high-quality hand-written syntactic grammars have been augmented with probabilities and optimized on corpora to help them select the most likely parse out of the many possible parses (Toutanova et al., 2005; Kaplan et al., 2004). The latter work has enhanced the hand-written broad-coverage LFG grammar in the XLE system with various strategies for robustness in the face of unknown words or sentence structures, and the system is reported to achieve performance (on unrestricted texts) which is at least as good as the best purely statistically-trained parsers available today.[5]

Another research direction that aids the goal of moving towards the middle of the scale is building very large repositories of lexical semantic knowledge, be it simply at the word level (WordNet), or with some more structural information (VerbNet, PropBank), or with additional mapping to the conceptual level (FrameNet). It is this kind of rich resources of knowledge that allowed systems like (Pasca and Harabagiu, 2001) to gain a significant edge over other systems that relied on a more impoverished level of knowledge. Although the sheer size of the English lexicon, including the number of connections between words, requires a huge effort to capture all of this knowledge, and the existing resources are still incomplete, having some knowledge is clearly better than no knowledge. No amount of computational sophistication could serve as a substitute for the knowledge itself (see also section 9.1).

3 The Frontier of Linguistic Knowledge

3.1 Annotation and Pre-Annotation Research

As discussed above, more sophisticated representations and more high-quality knowledge are what's required for increasing the quality of current NLP systems. This is required across the continuum of applications mentioned above: At the precise end of the scale, there is need for more knowledge to improve recall (cover more NL inputs); at the robust end of the scale, there is need for more knowledge to improve precision.

How can the current level and quality of NL understanding be enhanced in applications that allow arbitrary texts in arbitrary topics? Around 1990, when researchers wanted to enhance the sophistication of statistically-based parsing, they decided to start a huge effort of annotation in the form of the Penn Treebank. To achieve that, they first spent time writing a 300-page manual that essentially codified a syntactic grammar and provided guidance for annotators. Then the annotation effort commenced, and when it was done, the resulting corpus was available for researchers to develop probabilistic machine learning algorithms that could learn these representations.

In order to develop probabilistic models for the other parts of structural linguistic knowledge, a similar effort to the Penn Treebank is required. However, around 1990 there was a more-or-less broad consensus on what syntactic representations should be used, at least for the core part of NL, thanks to several decades of research on syntax in Linguistics that preceded the treebank effort (see more on that below). In

[5] This shows that the claim that hand-written grammars cannot scale up and compete with broad-coverage purely statistical parsers is false.

contrast, the level of consensus, understanding, and coverage in structural semantics in Linguistics today is less than it was for syntax in the 1990s. There are basic issues that have not yet been resolved. A good attempt to do so was the FraCaS project,[6] but many issues still remain unresolved, and more progress on them is needed before good annotation schemes can be developed. For example, there is still no *unified* semantic representation that is worked out in enough detail and that covers the main phenomena of structural semantics: quantifiers, scope ambiguity, plurality ambiguity, constraints on ellipsis and anaphora, time and event structure, modality, etc.

It is very important to invest the time and resources to do the necessary preliminary linguistic research in structural semantics and then the research for developing a good annotation scheme, parallelling the research that went into the 300-page Penn Treebank annotation document as well as the linguistic research that preceded it. Otherwise, the very expensive cost (time and money) of annotation will be a wasted effort since the quality of the results it could produce would be low and would not justify the cost (see the quote at the end of section 9.2).

Although there are precise computational grammars today that do incorporate some semantics (Copestake et al., 2002; XLE, 2002), the semantics side is not nearly well-developed as the morphological and syntactic sides of the grammars, and the resulting semantic representations (when the grammar is used in broad-coverage applications rather than on small examples) have not been tested as far as their ability to support correct inference.

In fact, even in syntax, there is still much room for improvement in the Penn Treebank annotation scheme itself. The analysis of some syntactic phenomena there is an approximation that could be improved (e.g. the internal structure of noun phrases), or is completely missing (e.g. correct analysis of ellipsis). It is especially important to get high-quality parse trees since we want to use them as the basis for doing structural semantics. Taking structural semantics and the structural syntax-semantics interface into account will help point out gaps in the current annotation scheme that would not be felt as important when only the syntax level is considered. For that, we need to pay closer attention to theories of syntax and semantics in Linguistics and recent developments in those areas.

In conclusion, as a first step for improving the quality of semantic understanding in current probabilistic models in NLP, research needs to be done on how to create good annotation schemes, just as was done before annotation actually started in the Penn Treebank project. But in order to do that, we need to spend time doing linguistic research on issues of representation in syntax, structural semantics, and the syntax-semantics interface.[7]

3.2 Reasons for Working on Structural Linguistic Knowledge

Granting that the above picture is correct, one might still wonder whether it is a useful research strategy to spend time concentrating on developing the structural linguistic knowledge, rather than other parts of the knowledge space. What could be gained by doing so? This section lays out several answers to this question.

[6] http://www.cogsci.ed.ac.uk/∼fracas/

[7] Actually, this point also applies to lexical knowledge, but this is not the focus of this paper.

First, the categories of knowledge discussed above are ordered along a scale: factual knowledge is the most sizable one; the amount of conceptual knowledge is smaller than that but still huge; the amount of knowledge that links words to concepts is smaller still though is very large (consider how many words and argument frames there are in a language); and finally structural language knowledge, while still very large, is the smallest of the lot. This scale also coincides with a scale of variability and change: facts are most frequent to change, less so with conceptual and lexical knowledge, and least with structural knowledge of language. Since this latter knowledge is the smallest in size and least changing, we have the best chance of doing a good job of capturing most of it.

Second, this knowledge will provide a tremendous enhancement to the quality of virtually all NLP applications across the entire spectrum of deep/narrow versus shallow/broad, since all NLP applications need to analyze NL input. In particular, more complete and precise structural linguistic knowledge will extend the range of phenomena and level of understanding that can be handled by applications dealing with high-quality precise understanding of NL in combination with conceptual knowledge and inference; and it will enhance the accuracy of systems such as arbitrary-text question-answering systems (Pasca and Harabagiu, 2001), since the matching they perform between queries and texts will be done not only based on word alterations and syntactic parsing but also based on a semantic analysis.

Third, the more structural language knowledge a computer has, the better able it is to *automatically acquire* factual knowledge correctly from crawling texts on the internet: possessing only knowledge of syntax allows only rudimentary acquisition of simple patterns of facts that appear explicitly in the text; but having more sophisticated semantic representations and inference based on them will allow it to understand semantic operators better as well as to combine correctly separate pieces of information that appear throughout the texts. This is precisely the utility of semantic representations, that they capture the *content*, i.e. meaning, of a text, and make it possible to relate the pieces of information to each other, merge them, compute entailments between them etc., things that are not possible if one relies only on the *form*, i.e. syntax, of the texts.

An additional justification for the second and third benefits above is that of all the kinds of knowledge, the categorical basis of the structural knowledge of language is virtually completely domain-independent, and so it is useful in a very wide range of cases. It is true that this knowledge exhibits differences across domains as far as its quantitative dimension is concerned, i.e. certain constructions may be more likely in one kind of domain than another (e.g. fiction vs. legal texts). Nevertheless, this knowledge does possess a form of generality that parts of the other kinds of knowledge lack. For example, while the "upper ontology" part of the conceptual knowledge is domain-independent, lower parts of the ontology hierarchy become gradually more specific and less broadly applicable.

The research strategy suggested here could be naturally combined with investigating the more general and domain-independent parts of the other kinds of knowledge, since those kinds are smaller in size than the rest of the knowledge. Still, they are larger than the structural linguistic knowledge. Moreover, there is only a limited amount of work that can be done in one dissertation. So I will concentrate here on

the structural linguistic knowledge, and leave it to others to work on the lexical and conceptual knowledge, though I may collaborate with them and use existing resources to the extent that their coverage happens to include what I need.

4 The Connection to Linguistics and NLP

Does the above research direction mean I simply advocate doing more research in linguistic semantics? After all, the idea of mapping out the knowledge of language is not new and is one of the main aims of the entire field of Linguistics. But there are a few points on which the planned research here diverges from common practice in Linguistics.

The first point is that linguistic research often focuses on quite *rare* phenomena or languages. There is a good rationale for that: just as physicists often discovered fundamental principles by examining rare and extreme phenomena, so the hope in Linguistics is that rare or esoteric phenomena would shed light on important unifying principles of natural language, and human thought more generally.

Nevertheless, for the purpose of producing practical computer applications, the language phenomena to investigate should be prioritized differently, where the most frequently used phenomena are researched before the rare ones. This strategy will result in computer systems that operate correctly in more cases. It's not that in general computers don't need knowledge of esoteric NL phenomena, but rather if we as designers are forced to choose (due to lack of time and manpower resources) which of two pieces of linguistic knowledge to feed the computer, the knowledge about the more frequent phenomenon will by definition result in a correct analysis more frequently than the knowledge about the less frequent phenomenon. There are still many gaps in the coverage of linguistic theory, even regarding basic and frequent language phenomena, that remain to be filled.

The second point is that many researchers in Linguistics, and especially formal semantics, still base their work on *artificial* data. Thus, investigated sentences are often invented rather than taken from real corpora. The disadvantage of doing so is the risk of investigating artificial sentence structures that no human would ever use. Such data do not advance the theories' ability to make predictions about real language use, and the discussion ignores many phenomena that appear in real texts. Moreover, example sentences in semantics papers are often analyzed out of context. This makes it more difficult to assess what possible readings a sentence has. For example, it is rarely the case that people use a full sentence such as "three boys lifted four pianos" without further modifiers or without a context that makes it more clear what is meant. It is true that sentences in NL are often underdetermined as to the situation they describe, but it is unlikely that a sentence such as the above has as many as ten or twenty possible readings, as suggested by some theories in semantics, and it is very hard to assess what it does mean without examining its uses in real contexts. Paying attention to natural data, whether in broad-topic corpora such as a collection of newswire text, or in a collection of texts in a specific domain, is a good idea whether or not one is also interested in the quantitative dimension (prevalence) of the phenomena.

The third point is that the knowledge in linguistic papers and books is often stated only semi-rigorously and so is not immediately applicable for a computational system.[8] Such papers might still have very good ideas and insights about the phenomena. But there is still a lot of work to make the knowledge into a rigorous form that is actually usable by a computer. Moreover, even in theories that are mathematically precise, such as HPSG (Pollard and Sag, 1994) and LFG (Dalrymple, 2001), a lot of work is necessary to flesh out the computational details (e.g. (Oepen et al., 2002)).

In particular, a common practice of theory evaluation in structural semantics in Linguistics is a manual inspection of the proposed logical forms and an informal assessment of whether they make intuitive sense. But the main purpose of a semantic formalization is to support predictions about entailment, and by extension, one could view their aim as supporting automatic computational inference.

As far as the relation to NLP is concerned: There has been a lot of work aimed at producing computer applications that do useful things with language input. But just as linguistic theories ignore computational issues for the most part, NLP work has tended to emphasize formalisms and techniques while downplaying linguistic theory as well as the actual study of language phenomena. For example, once it was realized that context-free grammars (and their probabilistic versions) can capture some aspects of language syntax, much of the effort went into devising parsers and algorithms for processing and learning these grammars. Yet to what extent those programs are actually useful in real applications that need to understand language has not been given enough attention. It is true that most NLP work today processes real NL data, but there hasn't been enough work of manual inspection of the NL phenomena that appear in the data as a basis for developing more accurate probabilistic models. For example, some recent efforts in NLP to incorporate some structural semantics use ad-hoc semantic representations that are not developed based on a principled study. Such an approach is useful for creating a baseline performance by seeing what can be achieved by utilizing off-the-shelf ML technology using a simplistic model of semantics, but it is not a viable strategy for making any serious research progress beyond baseline performance. The research program here is founded on the belief that no amount of computational sophistication can replace a meticulous investigation of the subject matter itself (i.e. language phenomena). (See sections 9.1 and 9.2 for more discussion and quotes).

In short, this project aims at both computational rigor and a principled, scrupulous study of language phenomena as they appear in context in real NL data, while prioritizing the study of the phenomena according to their usefulness for practical applications, and evaluating the proposed semantic theories and formalisms by testing their ability to support inference.

[8] I have many examples, but any particular example I give might create the false impression that I'm singling out that work for reproach. Lack of rigor can be found in many papers.

5 Evaluation

The research direction suggested here is intended to produce linguistic knowledge that is actually useful for computational understanding real NL texts. It is therefore important to discuss how this knowledge is to be evaluated.

The initial form of evaluation of the basic, categorical part of the structural semantic knowledge of language, separate from its quantitative dimension and from any other form of knowledge, is testing how well this knowledge can support inference. I do not intend at all to suggest that just having structural linguistic knowledge is sufficient for answering questions and doing inference. In fact, most real NL questions require a lot of lexical, conceptual, and even factual knowledge. Nevertheless, there are certain kinds of structure-based semantic inferences that *can* be made using only this knowledge with little or no need for the other kinds of knowledge. Such inferences include: logical inferences (e.g. syllogism, inferences based on logical connectives), numerical inferences (e.g. from "at least n P" infer "at least m P" for $0 < m < n$), and even restricted forms of sentence-bound scalar and conventional implicatures (see e.g. (Zaenen et al., 2005)). It is interesting to investigate exactly what inferences can be made with no or little recourse to lexical and conceptual knowledge, because these are the kinds of inference that we do have a chance of capturing completely in a computer, assuming we have all the domain-independent language knowledge. A good example for a test suit for this knowledge is the list of entailments and non-entailments compiled during the FraCaS Project (FraCaS, 1996, pp.63-120). A good task for testing the knowledge is solving logic puzzles from their textual descriptions (Lev et al., 2004).

After sufficient progress is made with developing and testing the structural linguistic knowledge as discussed above, the next stage of evaluation may diverge into various directions. Leaning towards the deep-understanding end of the continuum, this knowledge may be combined with an already well-developed body of conceptual knowledge in particular domains, and the level of understanding in those domains may be tested to see to what extent more sophisticated inferences can be correctly made from the NL input. Leaning towards the other end of the scale, an annotation scheme can be developed based on the accumulated structural knowledge, NL text corpora can be annotated, and statistical models can be developed to learn from this data, as well as to acquire its quantitative dimension, in order to produce analysis of NL texts with higher quality than is available today. The accuracy of these models in tasks that require understanding can be tested and compared to the accuracy of current systems.

6 Applications

Below is a list of applications that require a high quality of understanding of the meaning and information conveyed in texts. To reiterate, high-quality linguistic knowledge is insufficient by itself to solve these tasks (except for the very restricted comprehension tests) but it is necessary for these tasks in combination with other knowledge. The applications here have the property that all (or almost all) people agree about the meaning of their texts and about the correct answers to queries about

those texts. Moreover, the answers to the queries usually do not explicitly appear in the texts and require inference based on the various pieces of information that are distributed throughout the texts. These applications are also mostly at the deep-and-precise end of the continuum because they are more readily useful or available for testing the linguistic knowledge at the first evaluation stage. Nonetheless, as discussed above, the knowledge would be useful for applications across the continuum.[9]

The comprehension exams are various tasks designed to test the computer's level of understanding of a NL sentence or text (just as they test humans' understanding). They are different from general reading comprehension tasks which really tests world knowledge. No one has yet built a system that can pass exams such as the ones below across a broad enough sample of NL while using domain-independent NL-understanding components,[10] and so this research will push the boundaries of what is possible in computational language understanding, and in basic AI research more generally. The second list gives a few examples of real-world applications that would be very valuable to people, and which crucially rely on a high quality of understanding of text meaning.

Comprehension Exams

1. Identifying logical entailment, equivalence, and contradiction relations between (the readings of) two sentences.
   - A good example of such entailments is the list compiled during the FraCaS Project (FraCaS, 1996, pp.63-120).
   - The PASCAL RTE ("Recognizing Textual Entailments") Challenge[11] is overall *not* a good example of such entailments because the pairs of sentences there are related in diverse ways, including analogies and similarities, but mostly the problem is that identifying the entailments requires an unrestricted amount of general world knowledge.
2. Logic puzzles, such as those on LSAT and GRE exams (see (Lev et al., 2004)).
3. Math puzzles, such as those on SAT exams. (Solving such puzzles given their English description was one of the tasks that was suggested in discussions of the next DARPA Grand Challenge.)
4. Solving Advanced Placement (AP) tests on specific topics. (See e.g. Project Halo (HALO, 2004), which addresses the knowledge representation part of such a task, but not yet the language comprehension part).

Real World Applications

1. NL interface to databases: expressing a query by a NL question rather than an SQL query (see (Androutsopoulos et al., 1995) for a survey).

[9] Note that applications at the "any-text" end of the scale, by the fact that they allow arbitrary texts on arbitrary topics, tend to have much less consensus regarding what the correct answers are.

[10] [TBD: Mention e.g. Bobrow's STUDENT and GUS systems and explain.]

[11] http://www.pascal-network.org/Challenges/RTE/Introduction/

2. NL interface to a computational law[12] system: Transforming texts that describe precise regulations to representations that the computer could reason with in order to determine whether a given case complies with the regulations (e.g.: which courses a college student must take in order to fulfil the requirements of a study program.[13])

3. Understanding manuals and other technical and specification texts written in a restricted subset of NL, or "controlled language" (e.g. as used by the aero-space industry.[14])

4. Answering questions based on information given in a text drawn from a restricted class of texts.

7 Dissertation Plan

Investigating all of structural language knowledge is still a very large task. So this dissertation is going to concentrate on particular phenomena in syntax and especially structural semantics and the syntax-semantics interface. The details will be laid out in a future research memo. Generally speaking, I would like to develop a semantic representation language that captures correctly main topics in semantics that are ubiquitous in NL, such as: quantifiers, scope ambiguity, plurality ambiguity, comparatives, constraints on ellipsis and anaphora, time and event structure. I would like to investigate how these representations can be calculated from sentence parses, and how well they can support inferences that require little knowledge beyond structural linguistic knowledge. As explained above, the developed knowledge will be tested in such tasks as entailments and logic puzzles.

8 Summary

This research aims at developing structural linguistic knowledge, especially structural semantic knowledge, in a computational form, by investigating issues of representation and inference. This knowledge is necessary for understanding the meaning and information conveyed by NL texts, and it will be useful for NLP applications across the board, from precise systems that utilize a lot of conceptual knowledge in a specific domain to system that aim at processing arbitrary texts on arbitrary topics.

This research aims at both computational rigor and a principled, scrupulous study of language phenomena as they appear in context in real NL data, while prioritizing the study of the phenomena according to their usefulness for practical applications, and evaluating the proposed semantic theories and formalisms by testing their ability to support inference. For the dissertation, evaluation will be mainly on tasks that require precise semantic understanding with little or no recourse to non-linguistic knowledge, such as in structural semantic entailments and logic puzzles.

[12] See e.g. http://complaw.stanford.edu.
[13] Example text: http://cs.stanford.edu/Degrees/mscs/degree.php.
[14] See e.g. http://www.ics.mq.edu.au/∼rolfs/controlled-natural-languages/.

I believe that the kind of interdisciplinary research that I'm aiming at, which combines both computer science and linguistics, provides the grounds from which very interesting, useful, and exciting results will emerge. Since this research touches on the fundamental issues that are really necessary for making progress across the board in NLP, I believe it can have a big impact in the field.

9 Appendix

9.1 Paying Attention to Language and Representation

From Manning (2005):

> Why isn't NLP just an instance of applied machine learning? Because we're meant to know and care something about the details of language! Language is so rich that representation is what matters most. The field of NLP hasn't been paying enough attention to that. It hasn't been what's cool. A lot of smart people spend a lot of time looking at math instead of data! ... it's about time NLP people started paying more attention to the problems again, that is: issues of representation, doing data analysis, a.k.a. linguistics.[15]

9.2 The Importance of Linguistics to NLP

Some people may be slightly surprised by the heavy reliance on and synthesis of linguistic theories in this work, because of the common trend in NLP nowadays to regard them as somewhat too theoretical and irrelevant for practical NLP work. This unfortunate situation is not imaginary, as the well-known apocryphal quote attributed to Fred Jelinek, Head of IBM's Speech Recognition Group, indicates: "Every time I fire a linguist, the performance of my system goes up." And Oepen et al. (2002) cite on the first page of their book a statement made about their work by an anonymous NAACL 2000 reviewer, which reads: "Relevant only to some extent. State-of-the-art parsers are moving away from complex feature structure systems."

But the contrary is true. For example, the success of probabilistic parsers to learn the Penn Treebank corpus, and the success they had to the extent that they were integrated in useful NLP applications, is due to the partial consensus that existed in Linguistics regarding syntactic representations after several decades of research on Syntax. This consensus was codified in the 300-page manual that guided the annotators of the Treebank.

Here is a relevant quote from the preface to (Manning and Schütze, 1999):

> ... it is now generally accepted in Statistical NLP that one needs to start with all the scientific knowledge that is available about a phenomenon when building a probabilistic or other model, rather than closing one's eyes and taking a clean-slate approach.

[15] So as not to misrepresent the claims made, I'll add that Manning also said: "Maybe not the kind most commonly seen in U.S. linguistics departments [but] a new kind of data-driven inductive linguistics." I think the most important point is the need to look at and analyze data using sophisticated representations and theories.

> . . . the last thing we would want to do with this textbook is to promote the unfortunate view in some quarters that linguistic theory and symbolic computation work are not relevant to Statistical NLP.

And another quote from the description of the LSA 2005 course, "Why NLP Needs Linguistics: a case study" given by Annie Zaenen:[16]

> The evolution of computational linguistics from symbolic to statistical models has given the false impression that linguistic knowledge has become irrelevant for NLP. In fact nothing could be further from the truth: as soon as one moves away from simple Information Retrieval, one needs annotated corpora to be able to train the stochastic models that are en vogue. The problem is that the development of annotation schemes is often done rather haphazardly, creating the risk that the annotations based on them will in fact lead to incorrectly trained models.

References

Androutsopoulos, I., G.D. Ritchie, and P. Thanisch. 1995. Natural language interfaces to databases–an introduction. *Journal of Language Engineering* 1:29–81.

Copestake, Ann, Dan Flickinger, Carl Pollard, and Ivan Sag. 2002. English Resource Grammar website. http://lingo.stanford.edu/erg.html.

Dalrymple, Mary. 2001. *Lexical Functional Grammar*, volume 34 of *Syntax and Semantics Series*. Academic Press.

FraCaS. 1996. Using the framework: Deliverable 16 of the FraCaS project. http://www.cogsci.ed.ac.uk/∼fracas/.

HALO. 2004. Project website. http://www.cs.utexas.edu/users/mfkb/RKF/projects/halo.html/.

Kaplan, R., S. Riezler, T. H. King, J. T. Maxwell III, A. Vasserman, and R. Crouch. 2004. Speed and accuracy in shallow and deep stochastic parsing. In *Proc. of HLT-NAACL'04*.

Lev, Iddo, Bill MacCartney, Christopher D. Manning, and Roger Levy. 2004. Solving logic puzzles: From robust processing to precise semantics. In *Proc. of the 2nd workshop on text meaning and interpretation, ACL'04*.

Manning, Christopher D. 2005. Refocusing on linguistic representations. Talk given at NLaSP Colloquium, Stanford University, January 19th, 2005.

Manning, Christopher D., and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. MIT Press.

Oepen, Stephan, Dan Flickinger, Jun ichi Tsujii, and Hans Uszkoreit. 2002. *Collaborative language engineering: A case study in efficient grammar-based processing*. CSLI Publications.

Pasca, Marius, and Sanda M. Harabagiu. 2001. High performance question/answering. In *Proc. of SIGIR*, 366–374.

Pollard, Carl, and Ivan Sag. 1994. *Head-driven phrase structure grammar*. Studies in Contemporary Linguistics. The University of Chicago Press.

Toutanova, Kristina, Christopher D. Manning, Dan Flickinger, and Stephan Oepen. 2005. Stochastic HPSG parse disambiguation using the Redwoods corpus. *Research on Language and Computation* 3:83–105.

[16] http://web.mit.edu/lsa2005/courses/descriptions/235.html

XLE. 2002. Xle website. http://www2.parc.com/istl/groups/nltt/xle/.

Zaenen, Annie, Lauri Karttunen, and Richard Crouch. 2005. Local textual inference: Can it be defined or circumscribed? In *Proc. of ACL*.