Question Answering for Dutch using Dependency Relations

Gosse Bouma
Alfa-informatica
Postbus 716
9700 AS Groningen
gosse@let.rug.nl
050-3635937

22nd September 2003

1 Summary

The project investigates the use of sophisticated linguistic knowledge and robust natural language processing for Question Answering (QA). In particular, we will investigate how syntactic and semantic dependency relations in the question and potential answer texts can be used to support QA. A demonstrator will be developed in cooperation with publisher *Het Spectrum*, who owns the rights of several major Dutch encyclopedias. In addition, we plan to participate in QA evaluation efforts of CLEF and similar conferences.

2 Description of Proposed Research

2.1 Motivation

Towards information navigation tools with integrated natural language processing capabilities. Both casual computer users and specialized information workers nowadays have access to large amounts of electronic information. However, on-line information sources are loosely structured, contain overlap, noise, and inconsistencies, are multilingual, and may contain speech as well as text. Current information retrieval technology, as exemplified by web search engines, only partially answers the demand for information navigation tools for such noisy and redundant data. In particular, there is a need for tools which help to locate relevant chunks of information efficiently, which extract and synthesize information from various sources, and which are interactive, in the sense that they can enter into an information dialogue with a user. The development of such tools requires a marriage between information retrieval and natural language processing (NLP) technology.

The question answering (QA) tracks of TREC are one of the best known examples of a research initiative aimed at the development of an information navigation tool with NLP capabilities. QA typically involves responding to ‘factoid’ (quiz) questions (*who, what, where, when, how much*) by providing a text snippet containing the answer from a collection of text documents. Successful systems often make use of ad-hoc tools (in particular for classifying the question (*time, place, person, company, amount,...*)), shallow NLP (for part of speech tagging, chunking (i.e. recognition of non-recursive base constituents), named entity recognition, and syntactic or semantic dependency labeling) and ontological knowledge sources (such as WordNet). The development of QA systems of this type provides a means to evaluate the effectiveness and feasibility of various approaches. In TREC 10, initial experiments were carried out for tasks which involve synthesis of information (in the so-called list task, which contains questions like *Name 4 countries that can produce synthetic diamonds*, where answers are typically a combination of information found in various documents), and dialogue capabilities (in the so-called context task, involving series of questions like *What grape variety is used in Chateau Petrus Bordeaux? How much did the futures cost for the 1989 vintage? Where did the winery's owner go to college? What California winery does he own?*). These tasks indicate that other aspects of NLP (generation, dialogue management) will have to be integrated with QA systems in order to provide the functionality needed for realistic applications.

Future information navigation tools will probably combine QA with information dialogue and language (and possibly speech) generation capabilities. A QA system which not only delivers text snippets, but also filters and synthesizes information, actually generates answers matching the question, and responds naturally to follow-up questions, can provide a significant improvement over current (web) search technology. The development of such applications requires that NLP techniques be taken even more seriously than they are in current QA research.

In the proposed project, we plan to implement a QA system for Dutch, which combines the techniques that have proven useful in other systems, with a dependency analysis based on full syntactic processing. The result of full parsing is a complete dependency analysis of the input question and the potential answer fragments. This has at least two advantages. First, full analysis may reveal dependencies which easily escape shallow approaches, and thus may contribute to accuracy. Statistical dependency parsing has matured to a state where it can be applied robustly and efficiently to large amounts of text. Integration of a dependency parser in a QA system therefore is practically feasible. Second, full syntactic analysis of the question and potential answer texts appears to be a requisite for systems which integrate QA with generation and dialogue capabilities. Dependency relations provide a relatively theory neutral level of representation, and are therefore an ideal representation level for interfacing with a dialogue manager or generation module.

A Practical Application. The Dutch publisher *Het Spectrum* published the first Dutch encyclopedia on CD-ROM (*De Prisma encyclopedie*). Recently, they have acquired the rights of the *Grote Winkler Prins* encyclopedia, which is used, among others, to produce the Dutch '*Winkler Prins* edition' of Microsoft's *Encarta*. A number of *Prisma* dictionaries can be used in combination with IFINGER (`www.ifinger.com`) to support intelligent text browsing. Het Spectrum is actively pursuing possibilities to exploit on-line versions of their encyclopedias and other reference works. As such, they are highly interested in tools which allow users to access this material in novel ways.

The development of a QA system requires, at the very least, sets of questions and suitable text collections in which answers may be found. Het Spectrum owns various resources which are relevant in this respect. For the current project, we will have access to at least one encyclopedia (*Grote Winkler Prins*), as well as the full set of questions (approx. 4000) of the game *Het Grote Winkler Prins Spel*. This is a quiz containing questions for which the answers can be found in the encyclopedia. Examples of questions are:

(1)
a. Op welk eiland ligt de vulkaan Etna?
   On which island is the vulcano Etna located?
b. Hoe heet de belangrijkste van de zeven aartsengelen
   What is the name of the most important of the seven arch angels?
c. Wat is een epigoon
   What is an epigone?
d. Waar staat de afkorting ABS voor in ABS-remsysteem

What does ABS mean in ABS brake system?

e. Wie werd in 1997 wereldkampioen Formule 1
Who became world champion Formula 1 in 1997?

f. Hoe heette Ho Tsji Minhstad vroeger
What is the original name of Ho Chi Minh city?

The advantages of this material are twofold. First, the quiz questions provide a large number of test items (even if a portion of it has to be discarded as being too whimsical), which would have to be developed by hand otherwise. Second, the encyclopedia provides an enormous amount of well-structured and carefully edited information. By itself, this already has a positive effect on accuracy. Early open-domain QA systems concentrated on this type of material (Kupiec 1993), and some QA systems for the Web (i.e. MIT's START, `www.ai.mit.edu/projects/infolab/`) still restrict themselves to selected sites containing reference material, rather than doing full web search. In addition, by using the additional information in the SGML-tags of the source, as well as indices for persons, organisations, and geographical locations, which are part of the encyclopedia, the accuracy of the QA system can be even further enhanced. An accurate QA system which makes optimal use of the structure and information already encoded in the text sources is an important asset for future products based on this material.

The progress of the project will be evaluated by Het Spectrum at regular intervals. As part of the evaluation it will be investigated how the technology developed within the research group can contribute to the efforts of the publisher towards on-line intelligent access of their reference works (perhaps in conjunction with Microsoft Research), and how other reference works of Het Spectrum (in particular, their dictionaries (including *Prisma* and *Kramers*) might contribute to our research.

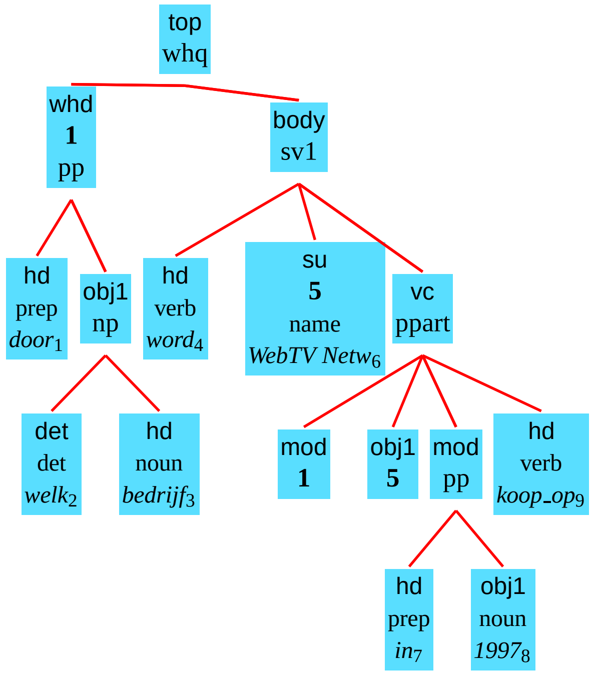
The performance of the system on less well-structured data and on different sets of questions will have to be evaluated in public QA evaluation competitions, such as the Dutch QA track organized for CLEF 2003. We will ensure that our system remains generic enough to be able to participate in such evaluations.

Strengthening Dutch NLP infrastructure. Research on Question Answering for English, as reported in the TREC QA tracks, has shown that successful QA systems must make use of both IR and NLP techniques. A wide-range of NLP techniques has been used for the task, often developed using valuable resources such as annotated corpora and dictionaries. Whether similar results can be achieved for other languages, for which less (accurate) tools and less (voluminous) resources are available, remains an open question.

Activities carried out by the Nederlandse Taalunie and within the European Euromap project were motivated by the desire to establish a language and speech infrastructure for Dutch suitable for supporting electronic access of information. Research projects like the NWO priority programme on language and speech technology and the initiative for a Corpus of Spoken Dutch have directly contributed to interdisciplinary research and the creation of resources.

The proposed project will combine various language resources for Dutch which have only recently become available and have not been used for NLP applications (the Alpino Parser (including a dictionary based on Celex and Parole), EuroWordNet, and (un-)annotated corpora such as the Alpino treebank, the Corpus of Spoken Dutch, the Twente NewsCorpus, and possibly the dictionary information owned by Het Spectrum). Therefore, it will provide valuable feedback concerning the quality and scope of these resources, which can be used to guide future initiatives in this area.

Scientific relevance. The Alpino Dependency Parser (Van der Beek, Bouma & Van Noord 2002, Bouma, Van Noord & Malouf 2001) is a wide-coverage and robust parser for Dutch which consists of a large lexicon (derived from Celex and Parole), a hand-written grammar, and a (maximum entropy)



⟨door, obj1 , bedrijf⟩
⟨bedrijf, det , welk⟩
⟨word, su, WebTV Networks⟩
⟨word, vc, koop_op⟩
⟨koop_op, mod, door⟩
⟨koop_op, obj1, WebTV Networks⟩
⟨koop_op, mod, in⟩
⟨in, obj1, 1997⟩

Figure 1: Dependency tree and relations for the question *Door welk bedrijf werd WebTV Networks in 1997 opgekocht?*

statistical disambiguation model trained on both annotated (over 7.000 manually annotated sentences) and unannotated data (several years of newspaper text). It is currently used to obtain dependency structures for the complete Twente Newspaper corpus (300 million words). The results will be used for improving the disambiguation model, for corpus-based acquisition of idiomatic expressions and valency information, and for retraining the part of speech tagger.

The system performs a full analysis of the input, and produces dependency trees as shown on the left in figure 1. The dependency labels assigned by the parser are adopted from the syntactic annotation guidelines of the Corpus Spoken Dutch project.

A given dependency tree defines a set of dependency relations, i.e., triples of the form ⟨`head word, dependency relation, dependent head word`⟩. The dependency tree in figure 1 defines the set of triples given on the right of that figure. The Alpino parser currently identifies dependency relations with an accuracy of approximately 84%. This compares well with results reported for identification of dependency labels in English text. Briscoe & Carroll (2002) and Carroll, Minnen & Briscoe (1998b) report accuracies between 76% and 83%. Note, however, that they use an annotation scheme which distinguishes only approximately 15 different labels, whereas our scheme uses 30 different labels, which makes the task of correctly identifying a label harder.

The Alpino parser is also robust and relatively fast. The methods for dealing with ungrammatical input for parsing speech recognizer output (Van Noord 2001), developed originally for the Groningen contribution to the NWO priority programme on language and speech technology, were incorporated into the Alpino Parser. This allows partial results to be returned in cases were a full parse fails. To parse unrestricted text, heuristics have been implemented which guess the syntactic properties of unknown words. Although parsing with a feature-based grammar of the kind used in Alpino remains computationally expensive, considerable improvements in efficiency have been achieved by including a part-of-speech tagger which filters unlikely tags suggested by lexical lookup (Prins & Van Noord 2001). The system is fast enough to make parsing of large corpora (up to 300 million words) practically feasible.

The use of a sophisticated module for grammatical processing has the potential of being able to identify syntactic and semantic dependencies that are beyond the possibilities of shallow parsers. For instance, the Alpino parser recognizes that in coordinations like (2), *Karslruher SC* is actually the subject of the VP *verloor daarin met .....*

(2) Karlsruher SC bereikte vorig seizoen de finale, en verloor daarin met 1-0 van het toen net gedegadeerde Kaiserslautern.
Karlruher SC made it into the finals last season, and lost with 1-0 from the just demoted Kaiserslautern

Similarly, full processing recognizes that the relative pronoun in (3), corresponds to the object of *gepubliceerd,* and that the relative clause modifies *werkgelegenheidscijfers* and not VS.

(3) De markt kijkt vol spanning uit naar de werkgelegenheidscijfers in de VS **die** vanmiddag om half vier Nederlandse tijd worden gepubliceerd .
The market is eagerly awaiting the employment statistics in the US, that are being published at three thirty, Dutch time, this afternoon.

Verbs selecting a non-finite verbal complement may either impose a reading where a grammatical object functions as subject of the complement (as in (4-a), where the object *Jeltsin* is the subject of *de wet te ondertekenen*), or where a grammatical subject is also the subject of the complement (as in (4-b), where *Abd ar-Rahmaan Moenief* is the subject of *schrijven*).

(4) a. Om Jeltsin te dwingen de wet te ondertekenen moet nu ook de Federatieraad zich uitspreken
To force Jeltsin to sign the law, the Federation council has to take a stand
b. Abd ar-Rahmaan Moenief , Jordanië 's bekendste schrijver , heeft met het Verhaal van een stad een biografie van Amman proberen te schrijven .
Abd ar-Rahmaan Moenief, Jordania's most famous writer, has tried to write a biography of Amman, with Story of a city.

Shallow processing is typically insufficient for recognizing such 'long-distance' and 'control' dependencies.

A full dependency parse of the question in figure 1 immediately reveals that *WebTV Networks* is actually the (logical) object of the verb *opkopen*, in spite of the fact that *WebTV Networks* functions as grammatical subject. It should be obvious that such information is useful, given potential answer strings like:

(5) Het bedrijf van miljardair Bill Gates heeft daartoe WebTV Networks voor zo'n achthonderd miljoen gulden opgekocht.
The company of billionaire Bill Gates has therefore bought WebTV Networks for approximately 800 million guilders.

The dependency tree for the question in figure 1 also shows that the question constituent actually functions as a modifier of the main verb *opkopen*. This is valuable information, as it tells us that a (short) answer to this question probably is a constituent which is a dependent of the verb *opkopen*. Note, however, that in the active sentence in (5), the answer constituent *het bedrijf van miljardair Bill Gates* is the *subject* of *opkopen*. This indicates that knowledge about the possible realizations of dependency relations (i.e. subjects are typically realized as *door*-modifiers in passive sentence) can help to make the performance of QA based on dependency relations more effective.

As the identification of dependency relations by Alpino is based on a full syntactic parse of the input, several processing steps performed independently in shallow parsing systems are performed implicitly by Alpino as well. In particular, the result of parsing is a parse tree, which contains dependency information, a POS tag and a stem for each word in the input, and syntactic bracketing. The most obvious omissions, from the perspective of QA, are named entity classification and more semantically inspired labeling of constituents (for instance, as temporal, locative, or other modifiers). Modules providing this type of information can be integrated within the course of the proposed project.

QA systems submitted to TREC have made only modest use of full grammatical processing. Typically, processing is limited to part of speech tagging, named entity recognition, and some form of shallow processing aimed at identifying non-recursive constituents. There are only few attempts to use dependency relations (Litkowski 1999, Litkowski 1999, Buchholz 2001). Systems which employ full parsing (Elworthy 1999) are rare, and have been used either to obtain syntactic bracketings or semantic representations. We feel that by restricting the output to syntactic bracketing only, important information is lost. Attempting to achieve a full semantic analysis, on the other hand, may be too ambitious for current parsing systems. Dependency relations provide a middle ground as they typically include only those aspects of syntax relevant for applications like question answering, and at the same time can be computed more robustly than abstract semantic representations. In the current project, we hope to show that accurate and robust dependency analysis can be used to boost the performance of a QA system.

2.2 Project Overview

2.2.1 Project Goals

The project concentrates on the development of a QA system for Dutch. Initially, we will build a prototype based on selecting those sentences from the text corpus whose dependency relations match best with the dependency relations found in the question. Performance issues are initially sidestepped by parsing the complete text collection off-line using the existing Alpino parser (as we have done before for corpora consisting of over 10 million words). The purpose of the prototype is to have a system in place quickly which can be used for testing and evaluation of additional components. Subsequent prototypes will include functionality resulting from the research projects described below (i.e. classification of the question, robust use of dependency relations and semantic roles, named entity classification, query expansion, pronoun resolution, robust handling of ambiguity) and the ability to parse on-line text fragments selected by a (web) search engine.

Within the project period, the system will be used for the following three applications:

- **QA for an encyclopedic database.** This application will be developed using the questions and encyclopedic data provided by Het Spectrum. It will make optimal use of the fact that this data is structured and comes with several indices.

- **Evaluation System**. To measure progress and to evaluate the performance of our system in comparison with others, we will participate in suitable evaluation contests, such as the Dutch QA contest organised for CLEF 2003 (`http://clef-qa.itc.it/`). The evaluation system will have to be able to operate on relatively unstructured data, such as newspaper text.

- **Demonstrator**. QA can be an integrated part of the demonstrator foreseen by the IMIX call for proposals and to be developed by all partners involved. In particular, we hope to integrate QA with language generation and dialogue components to support advanced information dialogues.

2.2.2 Research topics

Below, we describe thematically the main research topics for the three full-time project members (two PhD positions and one postdoc). All research must contribute to the project goals.

PhD A: Using dependency relations for QA. This subproject concentrates on the question how dependency relations can be used most effectively for QA. It addresses the following topics:

- **Analyzing the question.** QA relies heavily on automatic classification of questions into *question types* (*who, what, when, where, how much, etc.*). The question type helps to determine whether a potential answer string actually contains an answer to the question (i.e. an answer to a *who*-question should contain a person name). Being able to identify subconstituents of a potential answer which provide the actual answer to the question is essential for QA where only short (50 byte) answer sequences may be returned. Dependency analysis of the question provides several pieces of information useful for this type of classification. Given the question *Op welk eiland ligt de vulkaan Etna?*) it identifies the question constituent (*op welk eiland...*), the head word of the question constituent (*eiland*), the question word (*welk*), and the relation between the question constituent and the head of the sentence (*op welk eiland* is a locative complement of the verb *liggen*). On the basis of (a suitable subset of) the questions of *Het Grote Winkler Prins spel* and other sources, we will develop a module which classifies questions on the basis of a dependency analysis.

- **Linguistic realization of dependency relations.** The assumption behind the use of linguistic processing for QA is that questions and answers typically share much of their linguistic structure. If a question and a potential answer string have highly similar dependency structures, it is very likely that the potential answer string actually contains an answer to the question. However, there are also many cases where the structural relationship between question and answer is less direct. Questions for the name of a person, for instance, such as *Wie is de minister van Onderwijs?* contain a copula, and thus contain both a *subject* (*wie*) and *predicative complement* (*de minister van Onderwijs*). Answers, however, may be found in sentences containing a complex NP, where a name is an apposition to the head noun *minister* (e.g. *Gisteren sprak minister van Onderwijs, Maria van der Hoeven,...*, or vice versa (*Maria van der Hoeven, minister van Onderwijs voor het CDA, ...*). Using both linguistic sources and corpora, we will identify which dependency structures are realized in answers to various types of question.

- **Semantic Roles**. The dependency relations used by Alpino are adopted from the CGN project and are primarily motivated by syntactic considerations. For QA it is useful to label constituents with semantic relations as well, e.g. modifiers can be further classified as temporal, locative, causal, modal, etc. Verbal arguments can be labeled with role labels like agent, patient, cause, purpose, etc. We will integrate such knowledge in the Alpino system, concentrating on roles and distinctions that are most useful for QA. Suitable parts of the Alpino or CGN treebank will be annotated with semantic role information to allow the disambiguation module of the system to include this level of representation as well.

PhD B: Integration of external knowledge sources. QA can benefit from incorporation of semantic and conceptual knowledge. This subproject investigates and evaluates which external knowledge sources are most useful for QA, and how these can be integrated in our QA system. We will concentrate on development and integration of the following modules:

- **Named Entity Classification.** The lexical analysis module of Alpino contains a fair number of proper names, and heuristics for determining the grammatical category of unknown words. However, it does not subclassify proper names into persons, organizations, or locations. For QA, *named entity classification* has proven to be extremely useful. We will incorporate either a suitable off-the-shelf component for this task or develop such a component ourselves. For the recognition step (i.e. identification of parts of the input as names) we can use Alpino (which produces POS-tagged and bracketed representations of the input). The indices provided with the encyclopedic material can be used in combination with machine learning techniques to bootstrap a classifier.

- **Using Ontological Knowledge.** A potential answer may contain words which are (near) synonyms of the (content) words used in the question. For instance, a question containing the verb *overnemen* may be answered by sentences containing verbs like *kopen* or *verkrijgen* or even the noun *overname*. Similarly, to answer the question *welk land viel Polen binnen?* it is useful to know that *bezetten* is a typical consequence of *binnenvallen*. Another use for ontological knowledge is in question classification. To determine that *welke atleet won twee gouden medailles op de laatse Olympische spelen* asks for the name of a person, while *Welke club won de laatste Champions League finale?* asks for the name of an organization, one might use the information that an *atleet* (*athlete*) is a kind of person, whereas a *club* is a kind of organization. For English, WordNet has been used successfully (Harabagiu, Pasca & Maiorano 2000) to develop modules which provide ontological knowledge. For Dutch, EuroWordNet as well as the lexical resources available from Het Spectrum (i.e., the synonym relations available in the electronic version of the Dutch 'Kramers' dictionary), can be used to provide similar functionality. We will investigate the effectivity of question classification and query expansion based on ontological knowledge (i.e. as a means to improve recall of relevant documents and document fragments during the IR step of QA). Furthermore, we will investigate methods to match questions and answer strings where the head words are semantically related (i.e. near synonyms) rather than identical.

- **Pronoun Resolution.** The ability to track discourse referents over a sequence of sentences is important in QA, especially for producing answers based on longer stretches of text. For instance, one can imagine a system which is able to generate or summarize bio or CV information in response to questions like *wie is Hans Hoogervorst?* Bio statements typically refer to a single person by means of a variety of names, definite descriptions, and pronouns. Resolving the referent of pronouns such as *hij, zij, we,* and *het* and definite descriptions *Hans Hoogervorst, mr. H. Hoogervorst, de staatssecretaris, ...* is a prerequisite for synthesis of information from longer texts. Another example comes from the company *Alias I*, which markets a product that tracks persons in the news, by summarizing newspaper stories, using, among other things, named entity recognition in combination with coreference resolution (see `www.alias-i.com/product/mitap-faq.html`). We have access to an experimental but robust pronoun resolution system for Dutch based in part on dependency analysis and evaluated on a small annotated corpus (Bouma to appear). We hope to improve the accuracy and robustness of the system, among others by extending the development corpus considerably. The effect of pronoun resolution for QA (possibly also in combination with generation or summarization functionality provided by IMIX project partners) will be evaluated.

Postdoc: Integration, Training, and Evaluation of QA system. This subproject concentrates on optimizing the Alpino parser for QA as well as on system integration, testing, and evaluation.

- **Robust handling of dependency relations.** Parse errors often give rise to a situation where the wrong dependency relation has been assigned to a constituent. A prepositional phrase can for instance be assigned the labels *prepositional complement*, *modifier* and *locative or directional complement*. Deciding on the correct dependency relation for a PP is hard, both for humans and machines. However, errors in assigning the correct dependency relation label can probably be ignored to some extent in evaluating a potential answer. Therefore, we will experiment with various kinds of *back-off* strategies, in which the distinction between some dependency relation labels is ignored.

- **Robust parsing of potential answers.** QA systems using syntactic processing to date have either parsed complete collections of documents (selected by an IR system as being relevant to the question), text fragments (typically paragraphs) extracted from such documents by more detailed IR or IE machinery, or scattered sentence fragments returned by a Web search engine such as Google. Processing such texts imposes certain demands on the parser. Alpino has been used to parse text corpora consisting of ten million words and more, but further efficiency improvements may be required for QA. Also, parsing elliptical sentence fragments as returned by search engines may require further development of the techniques for handling ungrammatical and incomplete input.

- **Ambiguity** The parser produces many parses for a given sentence, ranked according to the score assigned by the disambiguation model. For questions, it seems promising to work with N-best lists of parses. For potential answer strings, only the best parse can probably be taken into account. We will perform a number of experiments to establish how the Alpino disambiguation component influences QA accuracy, and whether ambiguity can be allowed in parser output.

- **System integration.** The modules developed in the other two subprojects need to be integrated into a single QA system. The QA system will be used to build prototypes operating on full text, interfaces with a web search engine, and suitable for participation in evaluation contests. Furthermore, we foresee that QA functionality will be integrated into an IMIX demonstrator. Throughout the project we will collect data, and test and evaluate performance of various system components on the basis of realistic test data. Procedures will be established to measure progress during the project period.

3 International Perspective

Recently, considerable progress has been made in the area of full parsing of unrestricted text. The release of the Penn treebank (Marcus, Santorini & Marcinkiewicz 1994) has led to the development of statistical parsers, trained on treebank structures, which perform full syntactic analysis robustly and accurately (Collins 1999, Charniak 2000, Ratnaparkhi 1997). While such parsers were initially trained and evaluated on treebanks providing constituents with syntactic category labels (NP, VP, PP, *etc.*), there has been a growing interest in more theory neutral annotation scheme's based on dependency relations (Carroll, Briscoe & Sanfilippo 1998a). Dependency treebanks seem particularly useful for languages with a less strict word order than English, such as German and Czech (Skut, Krenn & Uszkoreit 1997, Hajicova, Panevova & Sgall 1998) and Dutch. The statistical parser described in Briscoe & Carroll (2002), which operates in a similar fashion as Alpino, can be applied to large amounts of unrestricted text, and produces output which is reliable enough to make it useful for a number of applications. The authors mention that initial experiments have been carried out aimed at applying their parser in a QA system.

Research on Question Answering in our view has not yet fully exploited the potential of deeper levels of syntactic analysis. As argued in section 2.1, full dependency analysis has the potential of identifying relations in potential answer strings which are beyond the capabilities of shallow approaches. This can contribute to both precision and recall of answer selection. At the same time, dependency parsing incorporates the functionality of modules found in systems combining several loosely coupled NLP modules (i.e. part-of-speech tagging, stemming, chunking). Our work on the Alpino grammar has shown that full dependency analysis can be efficient and robust.

The advantages of having a robust parser available, for QA for a language other than English, should not be underestimated. QA systems for Dutch do not seem to exist, although De Vries & Diekema (2002) describe a system which answers questions in English with answers retrieved from a Dutch text collection. Apart from an English-Dutch dictionary, no NLP for Dutch is applied. Most work on processing Dutch for information retrieval or classification purposes to date has concentrated primarily on lexical aspects, such as stemming and compound splitting (Kraaij & Pohlmann 1996, Spitters 2002, Kamps, Monz & De Rijke 2002). The proposed project can make a significant contribution to research on using advanced NLP for Dutch in information extraction and QA settings, and starts at a moment where there is a widely recognized need to develop QA for languages other than English (cf. the planned task at CLEF 2003).

The computational linguistics group in Groningen cooperates with Stanford University, Ohio University, and the University of Tübingen. An important theme in these cooperations is the development of robust and efficient parsers for linguistically motivated grammar formalisms.

Presentations concerned with the design and developement of the Alpino parser and treebank have been given, among others, at LREC (Language Resources and Evaluation conference, Gran canaria, 2002), IWPT (International Workshop on Parsing Technology, Beijing, 2001), Euralex (European conference on lexicography, Copenhagen, 2002), NAACL (North American conference of the Association for Computational Linguistics, Pittsburgh, 2001), WIAA (Workshop on the Implementation and Application of Automata, Pretoria, 2001), CONLL (Conference on Natural Language Learning, Taiwan, 2002), EACL (Conference of the European of the Association for Computational Linguistics, Toulouse, 2001), LINC (Linguistically Interpreted Corpora, Leuven, 2001), FSMNLP (Finite State Methods in Natural Language Processing, Helsinki, 2001) and NLPRS (Natural Language Processing Pacific Rim Symposium, Tokyo, 2002). Invited presentations were given in Baltimore (2002), Brighton (2002), Edinburgh (2002), Leuven (2003), and Utrecht (2001).

References

Van der Beek, L., G. Bouma & G. van Noord. (**2002**). Een brede computationele grammatica voor het Nederlands. *Nederlandse Taalkunde*, 7(4):353–374.

Bouma, G. (**to appear**). Doing Dutch pronouns automatically in OT. In: *Proceedings of the EACL 2003 Workshop on The Computational Treatment of Anaphora*.

Bouma, G., G. van Noord & R. Malouf. (**2001**). Alpino: Wide-coverage computational analysis of Dutch. In: *Computational Linguistics in The Netherlands 2000*. Rodopi, Amsterdam.

Briscoe, T. & J. Carroll. (**2002**). Robust accurate statistical annotation of general text. In: *Proceedings the third international conference on Language Resources and Evaluation (LREC 2002)*, 1499–1504. Gran Canaria.

Buchholz, S. (**2001**). Using grammatical relations, answer frequencies and the world wide web for TREC question answering. In: E. Voorhees & D. Harman, (red.), *Proceedings of the tenth Text Retrieval Conference (TREC 10)*, 502–509. Department of Commerce, National Institute of Standards and Technology, Gaithersburg, Maryland.

Carroll, J., T. Briscoe & A. Sanfilippo. (**1998**a). Parser evaluation: A survey and a new proposal. In: *Proceedings of the LREC 1998*, 447–454. Granada, Spain.

Carroll, J., G. Minnen & T. Briscoe. (**1998**b). Can subcategorisation probabilities help a statistical parser? In: *Proceedings of the 6th ACL/SIGDAT Workshop on Very Large Corpora*, 118–126. Montreal.

Charniak, E. (**2000**). A maximum-entropy-inspired parser. In: *Proceedings of the first meeting of the North American chapter of the ACL*, 132–139.

Collins, M. (**1999**). *Head-driven Statistical Models for Natural Language Processing*. proefschrift, University of Pennsylvania, (**1999**).

Elworthy, D. (**1999**). Question answering using a large NLP system. In: E. Voorhees & D. Harman, (red.), *Proceedings of the ninth Text Retrieval Conference (TREC 9)*, 355–361. Department of Commerce, National Institute of Standards and Technology, Gaithersburg, Maryland.

Hajicova, E., J. Panevova & P. Sgall. (**1998**). Language resources need annotations to make them really reusable: The Prague Dependency Treebank. In: *Proceedings of LREC 1998*, 713–718. Granada, Spain.

Harabagiu, S., M. Pasca & S. Maiorano. (**2000**). Experiments with open-domain textual question answering. In: *Proceedings of the COLING-2000*. Association for Computational Linguistics/Morgan Kauffman.

Kamps, J., C. Monz & M. de Rijke. (**2002**). Combining morphological and ngram evidence for monolingual document retrieval. In: *Proceedings of the Third Dutch Information Retrieval Workshop (DIR 2002)*, 47–51. Leuven.

Kraaij, W. & R. Pohlmann. (**1996**). Viewing stemming as recall enhancement. In: H.-P. Frei, D. Harman, P. Schäuble & R. Wilkinson, (red.), *Proceedings of the 19th ACM-SIGIR Conference on Research and Development in Information Retrieval*, 40–48.

Kupiec, J. (**1993**). Murax: A robust linguistic approach for question answering using an on-line encyclopedia. In: *Proceedings of SIGIR*, 181–190. ACM, Pittsburg, PA.

Litkowski, K. (**1999**). Question-answering using semantic relation triples. In: E. Voorhees & D. Harman, (red.), *Proceedings of the eigth Text Retrieval Conference (TREC 8)*, 349–356. Department of Commerce, National Institute of Standards and Technology, Gaithersburg, Maryland.

Marcus, M. P., B. Santorini & M. A. Marcinkiewicz. (**1994**). Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19(2):313–330.

Van Noord, G. (**2001**). Robust parsing of word graphs. In: J.-C. Junqua & G. van Noord, (red.), *Robustness in Language and Speech Technology*. Kluwer Academic Publishers, Dordrecht.

Prins, R. & G. van Noord. (**2001**). Unsupervised pos-tagging improves parsing accuracy and parsing efficiency. In: *IWPT 2001: International Workshop on Parsing Technologies*. Beijing China.

Ratnaparkhi, A. (**1997**). A linear observed time statistical parser based on maximum entropy models. In: *Second Conference on Empirical Methods in Natural Language Processing*, 1–10. Providence, R.I.

Skut, W., B. Krenn & H. Uszkoreit. (**1997**). An annotation scheme for free word order languages. In: *Proceedings of the Fifth Conference on Applied Natural Language Processing*. Washington, DC.

Spitters, M. (**2002**). Comparing feature sets for learning text categorization. In: *Proceedings of the 6th Conference on Contet-Based Multimedia Access (RIAO 2002)*, 1124–1135. Paris.

De Vries, A. P. & A. Diekema. (**2002**). Some experiments with the Dutch text collection. In: C. Peters, (red.), *Working Notes for the CLEF 2002 Workshop*. Rome.