

Adapting an English Information Extraction System to Swedish

Kristofer Franzén
Information and Language Engineering
Swedish Institute of Computer Science (SICS)
Box 1263, SE-164 29 Kista, Sweden
franz@rics.se

This work was made possible thanks to generous funding from The Swedish Institute and The Swedish Foundation for International Cooperation in Research and Higher Education (STINT).

Abstract

This paper presents work on adapting the Proteus Information Extraction system to Swedish. It turned out that the cross-lingual adaptation as such was fairly straight-forward; however, the Proteus system design did not render itself that well to reconfiguration at such a low level as needed. To evaluate the adaptation, the system was tested on a Swedish version of the MUC-6 Scenario Template Task. The Swedish version performed excellently on a training corpus, but quite discouragingly on an unseen test corpus. As a consequence of that work, a new Information Extraction system is being designed and the layout of that system is described.

1 Introduction

A well-known problem in the area of Information Extraction regards the adaptation of an extraction system to handle a new class of events (Yangarber and Grishman, 1997). With the increasing interest in multi-lingual and cross-lingual information extraction, it becomes necessary to construct systems that are easily adaptable, not only to new extraction tasks, but also to new languages. This paper presents work on adapting the Proteus Information Extraction system (Grishman, 1995; Yangarber and Grishman, 1998) developed at New York University, to Swedish. The system has previously successfully been adapted to Japanese (Sekine and Nobata, 1998).

The topics covered in the following sections are: an introduction to the Information Extraction task, a description of the Proteus Information Extraction system, an account of the adaptations made to the system and some results from evaluating the adapted system. A description of our present work on designing a new information extraction system and the motivations behind it will conclude the paper.

2 Information Extraction

Information Extraction can be defined as the task of extracting instances of a predefined class of events from natural language texts, and to build a structured and unambiguous representation of the entities participating in these events and the relations between them.

While Information Retrieval (i.e., document retrieval) systems aim at returning a ranked list of documents as an answer to any arbitrary information need (posed in the form of a query), an Information Extraction system is tuned to a specific, well-specified, predefined and persistent information need. Input to the system is a stream of unrestricted text and it outputs a structured representation in the form of a filled template or database record for every instance of an answer to the information need.

In Figure 1, an actual information need that could be satisfied by an Information Extraction system is shown. The description is taken from the specification of the MUC-6 Scenario on Management Succession.

“This scenario concerns events that would be of interest to an analyst who tracks changes in company management. The event object captures the management post, the company, the current manager, and the reason why the post is or will be vacant. The relational and low-level objects capture information on who’s “in” and who’s “out”, where the new manager came from, and where the old manager is going. A relevant article refers to assuming or vacating a post in a company and must minimally identify the post and either the person assuming the post or the person vacating the post.”

Figure 1: The narrative description of the MUC-6 Information Extraction Task.

A short text in Swedish and parts of the templates that could be filled in with information from the text, based on the above task definition, are shown in Figure 2.

Information Extraction and its methods of evaluation have to a great extent been defined by the Message Understanding Conferences (Grishman and Sundheim, 1996; Sundheim, 1991; Sundheim, 1992; Sundheim, 1993; Sundheim, 1995). The conference series is organized in the form of a competition where the participating extraction systems are evaluated against key templates constructed by human annotators. The metrics used to evaluate the systems are standard precision and recall measures over the template slots:

$$Precision = P = \frac{CorrectAnswers}{AnswersProduced} \quad Recall = R = \frac{CorrectAnswers}{TotalPossibleCorrect}$$

These values are often combined into an F-measure:

$$F = \frac{(\beta^2 + 1)PR}{(\beta^2P + R)}$$

Where β is a parameter that represents the relative importance of Precision (P) and Recall (R).

<p><i>Karo Bio. Per-Olof Mårtensson har åter utsetts till VD efter att sedan förra våren ha varit ordförande. Mårtensson efterträds på ordförandeposten av Bertil Hållsten, tidigare chef för S-E-Bankens läkemedelsfonder.</i></p> <p>(‘Karo Bio. P-O M. has been reappointed president after serving as chairman of the board since last spring. Mårtensson is succeeded as chairman by B. H., former head of S-E-Banken’s pharmaceutical funds’).</p>	<p>POSITION VD (‘president’)</p> <p>COMPANY Karo Bio</p> <p>IN-PERSON Per-Olof Mårtensson</p>
	<p>POSITION ordförande (‘chairman’)</p> <p>COMPANY Karo Bio</p> <p>IN-PERSON Bertil Hållsten</p> <p>OUT-PERSON Per-Olof Mårtensson</p>
	<p>POSITION chef (‘head’)</p> <p>COMPANY S-E-Bankens läkemedelsfonder</p> <p>OUT-PERSON Bertil Hållsten</p>

Figure 2: A short text and the three simplified templates it would generate.

As Appelt and Israel (1999) point out, interannotator agreement has been as low as 60-80% in the MUC:s (depending on MUC-task), which indicates that information extraction is a difficult task also for humans. They claim that, depending on the complexity of the extraction task and the preparation time, among other things, it seems very hard for an extraction system to reach beyond 60% of human accuracy.

Obviously, one of the main problems for an information extraction system is how to account for the linguistic variation in which the information is expressed in the text. This difficulty concerns lexical and syntactic variation as well as variation at the level of discourse and pragmatics. Consider the following constructed examples:

Assam Pärks nye VD Fjun Färneryd . . .

(Assam Pärks’ new CEO Fjun Färneryd . . .)

Fjun Färneryd har utsetts till ny verkställande direktör för Assam Pärks AB.

(Fjun Färneryd has been appointed new chief executive officer of Assam Pärks AB.)

Fjun Färneryd, som igår utsågs till ny VD för Umeå-företaget Assam Pärks, . . .

(Fjun Färneryd, who yesterday was appointed new CEO of the Umeå-based company Assam Pärks, . . .)

F. Färneryd, 47 år och nybliven direktör för Assam Pärks, . . .

(F. Färneryd, 47 years old and newly appointed president of Assam Pärks, . . .)

Assam Pärks styrelse utsåg igår Fjun B. Färneryd till direktörsposten i ledningen för företaget.
(The board of Assam Pärks yesterday appointed Fjun B. Färneryd to the post as managing director of the company)

Assam Pärks har fått en ny VD. Fjun Färneryd satt tidigare i ledningen för Eckym Ropos, men fick lämna posten efter påståenden om insideraffärer.

(Assam Pärks has appointed a new CEO. Fjun Färneryd was earlier part of Eckym Ropos' management but had to resign after allegations of insider dealing.)

Just consider the difference in the first and the last example above, where, in the first case, a single noun phrase expresses relations that require supra-sentential inferential processing to deduct from the last example.

An Information Extraction system aims at text understanding, but only from the perspective relevant to the information need at hand. There is no need to resolve ambiguities in the text as long as they are not relevant to the present extraction task. Therefore most extraction systems make do with shallow parsing techniques (Grishman, 1995; Hobbs *et al.*, 1997) and local text analysis.

3 The Proteus Information Extraction System

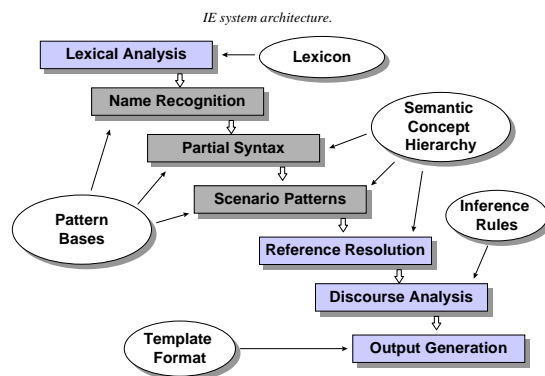


Figure 3: General architecture of the Proteus Information Extraction System.

New York University's Proteus system has a cascaded finite state transducer architecture common to many information extraction systems. It has a modular pipelined design consisting of domain-independent core components with domain-specific knowledge bases and task-dependent components for the specific scenario at hand. The lion's share of the linguistically interesting components of the system is defined in a number of pattern bases that are compiled at run-time into finite state transducers that perform deterministic, bottom-up, partial parsing and sequentially construct analyses of the text on increasingly higher levels of abstraction, building on the results from preceding transducers. The pattern bases contain rules that consist of a pattern part and an action part. A stylized rule to identify person names could look like

(Title)? CommonFirstName (MiddleInitial)? CapitalizedWord → Tag as Person!

The rule would give *Mr. Fjun B. Ferneryd* an annotation of type *Person* which could be referred to in consecutive rules, e.g., in a pattern that identifies a company's appointment of someone:

NounGroup(type = Company) VerbGroup(type = Appoint) NounGroup(type = Person)

The general information flow through the system is shown in Figure 3 and the functions of the different modules are briefly described in Table 1 (see next page).

As can be seen in Table 1, some parts of an information extraction system are independent of the specific extraction task at hand, and some modules have to be modified when the system is tuned to a new task. This does not mean that the functionality of the component in itself should be changed, but rather that some rules or some knowledge contents that modify the behavior of the module have to be changed. The same should apply when porting an extraction system to a new language. For an English-Swedish bilingual system, there should be different rule sets for lexical analysis components, syntactic analysis components, and scenario specific patterns as well as for the pattern generalizations, but the rules guiding anaphora resolution and discourse analysis could possibly be the same; many of the knowledge bases could be shared, but the lexical leaves of the semantic concept hierarchy have to be re-mapped.

4 Changes made to the system

We aimed at changing the system as little as possible, but still get a reasonably good result. The Proteus system was adapted to the Swedish extraction task in the following ways

- *Input format.* Since the lexical analyzer and the tagger of the Proteus system were not to be used, an SGML interface to the system was constructed to facilitate the input from any external resources. For want of better alternatives, the SWECCG tagger and disambiguator from Lingsoft (Karlsson *et al.*, 1995) was used for pre-processing the Swedish text, which then had to be postprocessed to deal with the inconsistent SWECCG output. The output was then transformed into the SGML format.
- *Rule predicates.* A rule in the system consists of a pattern matching part and an action part. Some of the predicates used in the pattern matching part of the rules were modified to allow for richer descriptions of the matched elements. Minor adjustments had to be made for Swedish (Latin-1) characters to be accepted in the pattern matching rules and their actions.
- *Domain and task independent rules.* Patterns for noun groups and verb groups had to be redefined, as well as patterns to identify, for example, people, organizations and locations.

Module	Description	Scenario Specific
Core modules		
Lexical analysis	Assigns part-of-speech tags to the text	no
Name detection and categorization	Identifies person names, company names, names of locations and possibly products	no ? no ?
Analysis of numerical expressions	Identifies monetary expressions, percentages and time/dates	no
Noun group detection	Identifies noun phrases without right modifiers.	no
Verb group detection	Verb + auxiliaries to identify main verb and tense.	no
Noun phrase detection	Full noun phrases for important scenario entities.	yes
Anaphora resolution	Resolution of pronouns and definite nouns involved in the scenario.	no
Scenario specific pattern matching	Top-level patterns for the specific extraction task.	yes
Discourse analysis	Co-reference analysis on the discourse level to merge events.	?
Inference rules	Formalizes world knowledge in rules so that text content fits template format. I.e., "last week" becomes a date, etc.	yes
Template generation	Produces the filled template for the specific task.	yes
Supporting modules		
Knowledge bases	Lists of common first names, corporations, locations and scenario specific entities.	no yes
Semantic Concept hierarchy	A hierarchy of concepts to support pattern matching, anaphora resolution and discourse analysis.	yes
Pattern production module	Allows for the user to produce scenario specific patterns interactively from examples.	no
Pattern generalization module	Meta-patterns that generalize patterns to match various kinds of subjunctive clauses, reduced clauses, passives etc.	no

Table 1: Common modules of an Information Extraction System.

- *Task specific rules.* Patterns for *entities* participating in the *events* of the task had to be redesigned, as well as the patterns for the *events* themselves.
- *Knowledge bases.* Several knowledge bases specific to Swedish were compiled to support the identification of names of people, organizations, locations and reportable positions,¹ etc.

The difficulties in trying to adapt the Proteus system to Swedish were not the linguistic differences (as expressed, for example, in the shallow parsing pattern matching rules), nor the differences in how the events were expressed in the different languages (as is expressed in the higher level rules); these patterns and rules were often surprisingly interchangeable, with small modifications, across the languages. What posed severe problems were the technical difficulties in changing a very complex system that was not initially built to be reconfigured on such a low level. Even though there is a graphical user interface to the English system in which the user can build patterns incrementally from examples, that tool would have required extensive work to function with the SGML input format and the other modifications made to the system.

5 Experiment and evaluation

For a comparison of the performance of the Swedish and the English systems, the Scenario Template Task of MUC-6 was chosen. This task concerns changes in corporate executive management personnel, as described in Figure 1. A Swedish corpus was compiled consisting of 34 financial news articles from *Tidningarnas Telegrambyrå* and *Affärsvärlden*. This training corpus contained 51 reportable events for which key templates were constructed. Rules were written and evaluated iteratively with the MUC-scorer² on the corpus until an F-score of 55.45 was obtained. In comparison, the systems at the MUC-6 evaluation ranged from about 48 to 56 in F-score on the test corpus (Sundheim, 1995). After extensive training, the Proteus system has since been boosted to perform at an F-score around 65 on the same task.

A test corpus consisting of 50 financial news articles from the same sources as the training corpus was compiled, as well as template keys, by an annotator not involved in the adaptation of the extraction system. The results from running the system on the test corpus seem quite discouraging with an F-score of around 28, but have not yet been fully analyzed. Further analysis will show if they are due to over-training of the system, faulty system design, or mismatches in the annotator's and the author's interpretation of the template filling rules.

6 Building a new system

The overall experience of trying to adapt the Proteus Information Extraction system to Swedish has led to the decision to build a new Information Extraction system. This

¹For example, according to the MUC-6 definitions, 'chairman of the board' is a reportable position while other types of chairpersons are not reported.

²The MUC-scorer is described in <http://www.muc.saic.com/scorer/Manual/manual.html>

system will be inspired by the general architecture of the Proteus system, but also by suggestions of improvements of that system that came up during and after the work cited in this paper. The new system will be built around a document manager which functionality is a subset of that in the Tipster Architecture (Grishman and others, 1996). This means that all internal functions are based on manipulating annotations of the text. We will aim at giving the system the following features:

- *Easily portable to new domains.* We recognize the need for a tool that facilitates for the non-expert user to write rules for a new extraction task without knowing the internals of the system or the syntax of the pattern matching language. Such a tool for example-based pattern acquisition exists in the Proteus system (Yangarber and Grishman, 1997).
- *Easily portable to new languages.* We will make every effort not to build language prerequisites into the system. For example, there will not be any restrictions on what features or feature values that may be found in an annotation.
- *Easily extensible.* Since there will be a well-defined interface to the document manager and a general set of methods to manipulate document annotations, and since there will be no restrictions on what the features of the annotations can be, the system is not limited to merely Information Extraction tasks, but can be extended to any document or text manipulating task.
- *Modular and flexible.* The system will have an object oriented design with distinct interfaces between the modules. If a new module is required for an analysis task, it should be easy to include it in the existing set of modules.
- *Platform independent.* The system will be implemented in Java and not dependent on external software in itself. Initially, the extraction system will be dependent on an external lexical analysis component.
- *Open Source.* Every part of the core extraction system will be open source and free to use for research or commercial purposes.

7 Conclusion and Further Work

Even though the Proteus system has previously successfully been adapted to Japanese, our experiences in adapting the system to Swedish have led us to believe that it will be worth the effort to build a new extraction system from scratch, taking into account portability not only on the task level, but also on the language level. Such a project has been initiated in collaboration with New York University and the result will eventually be publicly available.

This work has also led to the composition of the above mentioned test corpus for Swedish Information Extraction systems which will also be publicly available as soon as copyright issues are solved.

References

- Appelt, D. E. and Israel, D. J. 1999. Introduction to Information Extraction Technology. <http://www.ai.sri.com/~appelt/ie-tutorial/IJCAI99.pdf>. A Tutorial Prepared for IJCAI-99.
- Grishman, R. 1995. The NYU system for MUC-6, or where's the syntax? In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, Columbia, MD, November. Morgan Kaufman.
- Grishman, R. and Sundheim, B. 1996. Message Understanding Conference - 6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING96)*, Copenhagen, August.
- Yangarber, R. and Grishman, R. 1997. Customization of Information Extraction Systems. In *Proceedings of International Workshop on Lexically Driven Information Extraction*, Frascati, Italy, July.
- Yangarber, R. and Grishman, R. 1998. NYU: Description of the Proteus/PET System as Used for MUC-7 ST. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Virginia, USA, April. Morgan Kaufman.
- Hobbs, J. R., Appelt, D., Bear, J., Israel, D., Kameyama, M., Stickel, M., and Tyson, M. 1997. FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text. In Roche, E. and Schabes, Y., editors, *Finite-State Language Processing*, Language, speech, and communication. MIT Press, Cambridge, Massachusetts.
- Karlssoon, F., Voutilainen, A., Heikkilä, J., and Anttila, A., editors. 1995. *Constraint Grammar: A language-independent system for parsing unrestricted text*. Mouton de Gruyter, Berlin.
- Sundheim, B., editor. 1991. *Proceedings of the Third Message Understanding Conference (MUC-3)*. Morgan Kaufman, May.
- Sundheim, B., editor. 1992. *Proceedings of the Fourth Message Understanding Conference (MUC-4)*. Morgan Kaufman, June.
- Sundheim, B., editor. 1993. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*., Baltimore, MD, August. Morgan Kaufman.
- Sundheim, B., editor. 1995. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, Columbia, MD, November. Morgan Kaufman.
- Sekine, S. and Nobata, C. 1998. An Information Extraction System and Customization Tool. In *Proceedings of the New Challenges in Natural Language Processing and its Application*, Tokyo, Japan, May 25-26.
- Grishman, R. et al. 1996. TIPSTER Text Phase II Architecture Design. Technical report, Department of Computer Science, New York University, September.