ACTA UNIVERSITATIS UPSALIENSIS
Studia Linguistica Upsaliensia
7

Resourceful Language Technology

Festschrift in Honor of Anna Sågvall Hein

Edited by Joakim Nivre, Mats Dahllöf and Beáta Megyesi

UPPSALA
UNIVERSITET

Acta Universitatis Upsaliensis.
Studia Linguistica Upsaliensia 7.
214 pp.

ISSN 1652-1366
ISBN 978-91-554-7226-9
urn:nbn:se:uu:diva-8933 (http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-8933)

Tabula Gratulatoria

Lars Ahrenberg
Ingrid Almqvist
Jens Allwood
Bo Andersson
Jan Anward
Ulla Birgegård
Agneta Emanuelsson Blanck
Kristina Blomqvist
Lars Borin
Sven-Erik Brodd
Rolf Carlson
Sofia Gustafson Capková
Silvie Cinková
Robin Cooper
Mats Dahllöf
Bengt Dahlqvist
Helge J. Jakhelln Dyvik
Susanne Ekeklint
Joakim Enwall
Angela Falk
Danuta Fjellestad
Markus Forsberg
Eva Forsbom
Birgitta Garme
Barbara Gawronska
Christer Geisler
Björn Granström
Maria Toporowska Gronostaj
Britt-Louise Gunnarsson
Ebba Gustavii
Eva Hajičová
Johan Hall
Harald Hammarström
Gerd Haverling
Monica Hedlund
Hans Helander
Johan Heldt
Inga-Lill Holmberg
Merle Horne
David House
Cilla Häggkvist
Bo Isaksson
Carina Jahani
Kerstin Jonasson
Olle Josephson
Päivi Juvonen
Maarit Jänterä-Jareborg
Arne Jönsson
Birsel Karakoç
Fred Karlsson
Marousia Ludwika Korolczyk
Kimmo Koskenniemi
Hans Kronning
Merja Kytö
Einar Lauritzen
Caroline Liberg
Rolf Lundén
Lennart Lönngren
Ingrid Maier
Beáta Megyesi

Ulla Melander Marttala
Gernot Müller
Anette Månsson
Ingela Nilsson
Jens Nilsson
Mattias Nilsson
Joakim Nivre
Elisabeth Wåghäll Nivre
Bengt Nordberg
Torbjørn Nordgård
Coco Norén
Niklas Norén
Mikael Nordenfors
Maria Ohlsson
Leif-Jöran Olsson
Jarmila Panevová
Eva Pettersson
Ulf Pettersson
Helena Pontén
Aarne Ranta
Jan Olof Rosenqvist
Göran Rönnerdal
Markus Saers
Filip Salomonsson
Anju Saxena
Marianne Wifstrand Schiebe
Mojgan Seraji
Petr Sgall
Margareta Attius Sohlman
Per Starbäck
Eva Strangert
Ove Strid
Siv Strömquist
Stefan Strömquist
Lars-Göran Sundell
Kerstin Thelander
Mats Thelander
Jörg Tiedemann
Per Weijnitz
Åke Viberg
Gun Widmark
Henrik Williams
Caroline Willners
Martin Volk
Gustav Öquist

Preface

As the first holder of the first chair in computational linguistics in Sweden, Anna Sågvall Hein has played a central role in the development of computational linguistics and language technology both in Sweden and on the international scene. Besides her valuable contributions to research, which include work on machine translation, syntactic parsing, grammar checking, word prediction, and corpus linguistics, she has been instrumental in establishing a national graduate school in language technology as well as an undergraduate program in language technology at Uppsala University. It is with great pleasure that we present her with this Festschrift to honor her lasting contributions to the field and to commemorate her retirement from the chair in computational linguistics at Uppsala University. The contributions to the Festschrift come from Anna's friends and colleagues around the world and deal with many of the topics that are dear to her heart. A common theme in many of the articles, as well as in Anna's own scientific work, is the design, development and use of adequate language technology resources, epitomized in the title *Resourceful Language Technology*.

Uppsala in May 2008,

The editors

Contents

Searching Parallel Treebanks for Translation Relations

Lars Ahrenberg

Linköping University
Department of Computer and Information Science

1 Introduction

The relation of a translation to its original is of interest both to machine translation research and translation studies. In the latter field most of the work goes back to the so called linguistic period with Catford (1965), Vinay and Darbelnet (1977), and van Leuven-Zwart (1989) as prominent contributors. After what has been termed the cultural turn in translation studies, the preoccupation with translation relations is sometimes described in negative terms, even as a kind of dead end (e.g., Snell-Hornby, 2006), but especially in text books this aspect of translation studies is still prominent. Ingo (2007) is a recent example.

There can of course be no denial that target factors such as readership, historical period, perceived or explicit purpose, and cultural function determine the character of a translation. And so does the translator and her individual preferences and styles, at least in literary translation. But it is reasonable to make a distinction between description and explanation. Thus, I take it that categories such as interlinear (or word-for-word) translation, semantic, communicative, instrumental, and adaptive translations have an objective character where evidence for the categorization of a translation reside, wholly or partly, in the relation between source and target texts.

Translation relations primarily serve a descriptive purpose, but an important one. The characterization of a translation as a whole must be supported by evidence at the micro-level. This also means that characterization of the translation norms that are dominant for a certain historical period or type of translation rests on micro-level concepts. In addition, translation relations serve a pedagogical purpose in naming and explicating the possibilities and variants that are available to a translator. For the machine translator they can be of great value for evaluation and profiling of systems.

Parallel corpora are prime resources for the study and modelling of translations used in both computational linguistics and translation studies. In parallel corpora we do not only find alignments at the word level, but today also syntactic annotations and alignments at the phrase level. Thus, the parallel corpus has

become a parallel treebank. Neither syntactic annotation nor word alignment are processes that can be performed automatically with sufficient accuracy for reference data, which entails that accurate phrase alignment is hard to obtain without substantial manual efforts. Given that human resources are limited, one would like to make as much use of the manually reviewed annotations and alignments as possible.

The design and development of search interfaces to parallel treebanks is a challenge. A potential problem with any search interface is the mismatch in conceptualisation and vocabulary between users and developers. In the case of parallel treebanks, developers tend to be computational linguists, but if the resource is to be targeted at translators, translation scholars, and even interested laymen, the problem is real. As long as the interest is with the translation of words, there is less of a problem, but when attention is focused on structural and semantic changes the vocabularies and concepts are quite different. There is a great leap from alignment to notions such as shift or modulation!

There are but two ways to overcome the distance between user and treebank; either the users learn the details of the annotation, or the system offers search terms that are more easily understood by the users. These are not mutually exclusive, of course. In this paper, however, I will focus on the latter approach, and look into the question how far a parallel treebank that is accurately annotated and (word) aligned can support queries that are couched in terms of translation relations. My guess is, though, that descriptions such as Non-translation of noun or Reordering of head and dependent are easier to use, at least for the ordinary translator or the language teacher, than an interactive tool for drawing labelled trees. Part of the solution is then to define a vocabulary and mapping from such descriptions to sets of constraints that can be applied to the actual data.

If successful, we can reap other benefits as well. As indicated above, it would allow us to find instances of translation relations without having to find them manually first. At least, we would be able to scan the search space more effectively. Assuming that word alignment and syntactic annotation can be performed automatically with sufficiently good performance, we could also hope to develop programs that categorize translations automatically at the macro-level with respect to the type of translation, and support its judgement with profiles at the micro-level. In this paper I will discuss these issues from the perspective of a specific parallel treebank, the LinES parallel treebank, an outgrowth of research started in the PLUG project (Sågvall Hein, 2002).

2 Translation Relations

The relation of translation to original is manifested at many levels. Here I focus on the micro-level, i.e., on units smaller than the sentence such as words and phrases. I will use the term translation relation for a type of correspondence

between a micro-level item of a source text and the word or words that we deem constitute its translation in the target text.

An early attempt at systematizing translation relations is Catford (1965). He introduced the notion of translation shift to denote "a deviation from formal correspondence", where formal correspondence means that a source unit is translated by a target unit that has the same number of words, and where each word is equivalent in meaning with its source word. Catford further classified the shifts as (i) level shifts, and (ii) category shifts. Level shifts are restricted to shifts between grammar and lexis, while the category shifts are further divided into (a) rank (or unit) shifts, (b) structure shifts, (c) class shifts, and (d) intra-system shifts.

Another major contribution is Vinay and Darbelnet (1977) who defined a number of different types of correspondences: loan, calque, literal translation, transposition, modulation, equivalence, and adaption. In fact, they argued that these relations could be identified at three levels of correspondence: the lexical level, the phrase or grammar level, and the level of the message as a whole. An interesting aspect with these categories is that they define a kind of distance scale from target item to source item with identity at the one end and pragmatic associations at the other end.

van Leuven-Zwart (1989) develops a very detailed taxonomy that combines two grounds for classification: a formal difference in grammar or lexis, and a level such as stylistic or syntactic-semantic at which the effect of the change is located.

Ingo (2007) recognizes four aspects of text that are of constant importance to translation: the grammatical structure, the semantics, the text varieties, and the pragmatic aspects. As for the first aspect he uses a fairly elaborate taxonomy of formal shifts at the clause level in terms of clausal rank (Sw. satsgrad) and shows that languages differ with respect to their use of different ranks.

3 Word Alignment

In computational linguistics the notion of alignment became prominent with the increasing interest in parallel corpora in the 1990s. The basic idea is that units are aligned, whether at the sentence level or at the word level, when they correspond under translation. While alignment can be given a formally precise definition, it is notoriously difficult to apply consistently to empirical data and thus tends to become semantically overloaded. In contrast, the translation relations proposed within translation studies are usually provided with informal definitions that make intuitive sense.

Word alignment is usually defined as a relation between source tokens (words) and target tokens. Thinking of translation as the production of a target sentence from a source sentence Brown et al. (1990) write: "a word is *aligned* with the word that it produces". If a word produces nothing it remains unaligned, and

if it produces more than one word it is aligned with all of them. Later works on statistical machine translation and alignment have used slightly different definitions, but the important thing to note for the moment is that an alignment presupposes a specific tokenization of sentences and associate each token of one tokenization with zero or more tokens from the other. Moreover, alignment is subject to the condition that if two tokens have some token in common in their alignment, then they must share all tokens that they are aligned to. This case represents a correspondence that applies to a sequence as a whole, and which cannot be reduced to several correspondences among the parts.

In developing gold standards for evaluation a distinction between sure and possible, or clear and fuzzy, alignments is often made. This is a reflection of the difficulties people have experienced in agreeing on alignment judgements. It is not, however, a good strategy for alignment in reference corpora, where categories should reflect properties of the data consistently.

4 Word Alignment in LinES

The Linköping English-Swedish Parallel Treebank (Ahrenberg, 2007b), henceforth LinES for short, is developed with the primary goal of investigating how common words and grammatical constructions are treated when translated from English into Swedish. It can also be used as a test-bed for various studies of parallel data.

All tokens in LinES are annotated with information about stem, part-of-speech, morphological properties, and head token. The head token is given by a dependency analysis covering the complete segment that the token is part of. A segment is usually a single sentence, but it can be a phrase, or two or more sentences in sequence.

Word alignment in LinES is performed semi-automatically using the following major guidelines (see Ahrenberg, 2007a, for details):

- A content word of the source text is aligned with a content word of the target text iff they are related both structurally and semantically, though not necessarily synonymous. Multi-word units such as English compounds, and verb-particle constructions, are often aligned to a single content token on the other side,
- A function word, or functional multi-word unit, of the source text is aligned with a target function word iff they correspond structurally and have related functions.
- Words for which corresponding words cannot be found are aligned to a null element,
- Whenever possible a large alignment, i.e., one where two or more tokens on one side are aligned with two or more tokens on the other side, should

be analysed into smaller alignments. For instance, a phrase such as *this afternoon* when translated by *i eftermiddag*, is aligned using three small alignments *this* $\sim$ *0*, *afternoon* $\sim$ *eftermiddag*, and *0* $\sim$ *i*.

The reason for preferring small alignments is that the larger alignments will be created anyway in the phrase construction process.

A word alignment is primarily a relation between sets of tokens. From the alignment relation we can easily obtain an image function, I, that maps each source token on a set of tokens from the target sentence. Let s be a source token and S be the set to which s belongs under the word alignment A. Then $I(s) = T$ if and only if $A(S, T)$.

5 Phrase Alignment

In statistical machine translation phrase-based models have come to replace word-based models. A phrase is then usually taken to be any sequence of alignments that cover connected parts of both source and target training data. In LinES, a phrase is defined as a connected sequence of tokens that includes the head for all tokens that are part of it. To distinguish this entity from other kinds of phrases, I will refer to it as a *headed phrase*. This definition is an attempt to achieve a reasonable balance between recognising too few phrases, say only complete units, i.e., a head with all its direct or indirect dependents, or too many phrases, if any connected sequence of tokens is allowed to count as a phrase.

As an example, consider the nominal syntagm *a very nice car*. Sub-sequences of this syntagm that are recognized as headed phrases are *very nice*, *nice car*, and *very nice car*. The sequence *a very*, however, is not recognized as a headed phrase, since both words have their heads outside the sequence.

Some of the headed phrases generated in this way do not usually count as proper syntactic constituents. For instance, the sequence *trees in* is a headed phrase in the sequence *the trees in our garden*, as prepositions are regularly analysed as heads of prepositional phrases.

To determine the alignment of a source phrase, the image of every token in the phrase is determined. Taking the union of all such images we obtain a set of target tokens that constitutes an image of the phrase. We call this a *minimal image* of the phrase. Other images can be constructed, too, as null-aligned target tokens may exist. For instance, consider the the following sentence pair:[1]

EN: It is my childhood revisited.
SE: Det är som ett besök i barndomen.

The word alignments have been formed as follows: *my* $\sim$ *0*, *childhood* $\sim$ *barndomen*, *revisited* $\sim$ *besök*, *0* $\sim$ *som*, *0* $\sim$ *ett*, *0* $\sim$ *i*. This means that the

[1] From Saul Bellow's *To Jerusalem and back: a personal account*. Translation by Caj Lundgren.

minimal image of the headed phrase *my childhood revisited* will be *{besök, barndomen}*. By adding null-aligned target tokens to the image of the source phrase, we can obtain the maximal image *som ett besök i barndomen* which in this case is the desired image for the phrase. The basic principle employed in LinES is to add null-aligned function words, provided their heads can be found in the minimal image, or they constitute the head to the tokens already included. This principle works well in this example, as the words *ett* (a), and *i* (in) have the noun *besök* as head, and *som* (as) is analysed as the head of *besök*. In some cases, further discussed below, added content words may also be included.

English absolute constructions often correspond to complete clauses in Swedish as in *Removing the top lid will ensure...* being translated by *Om du tar bort det översta locket ser du till...* . In this case we would like to include the subjunction *Om* as well as the pronoun *du* in the image of the phrase *Removing the top lid*.

6 Searching Parallel Treebanks

As explained in the introduction, I believe that many users would be helped by having access to taxonomies of translation relations in their search for data in a parallel treebank. The relations produced by linguistically oriented translation scholars could then be taken as a starting point. However, they cannot be used as given. First, terminology is not uniform, but this problem can be handled by providing definitions. A more serious problem is that they rely, implicitly or explicitly, on knowledge that is not in the system. For instance, the system does not have access to meanings, referents or situations so it would be quite hard for it to distinguish a modulation ("a change of perspective on a situation") from an equivalence ("something said with the same effect in the given situation") using the categories of Vinay and Darbelnet (1977). Those modulations that involve a systematic formal change, such as the use of a passive form as the translation of an active form, may be recognized, but they should then preferrably be described as such.

What we want then is a taxonomy based on formal criteria, but presented in a vocabulary that makes sense to a user that is not so familiar with the formal concepts and abbreviations found in the parallel treebank.

6.1 Word-level Mappings

A simple means to classify alignments at the word level is to consider the number of tokens that are part of the alignment, positioning the number of source tokens before the number of target tokens as in 1-1, or 1-2. The number 0 would then be used for null alignments, of course. An advantage of this kind of representation is that it can immediately be interpreted as formal constraints on data to be delivered. On the other hand, this representation does not dif-

Descriptive term	Numerical type	Translation relations
Omission	1-0	
a) of content word		Deletion, Implicitation
b) of function word		Level shift, Economisation
Addition	0-1	
a) of content word		Explicitation, Compensation
b) of function word		Explicitation, Level shift
One-to-one translation	1-1	Loan, Calque, Literal translation, Class shift, Semantic shift
Split	1-n, $n > 1$	
a) adjacent		Decompounding, Depronominalization
b) non-adjacent		Transposition
Fusion	n-1, $n > 1$	
a) adjacent		Compounding, Pronominalization
b) non-adjacent		
Grouping	m-n, m,$n > 1$	Transposition, Modulation, Equivalence, ...

Table 1: A taxonomy of word alignment types and their relation to some traditional translation relations.

ferentiate between tokens that are adjacent, and tokens that are non-adjacent when there are two or more tokens in an alignment. Another problem is that it does not distinguish different kinds of tokens, and in LinES, there is a difference in how function words and content words are aligned. Also, it is quite far removed from ordinary language, so it should be provided with a terminology, and that terminology could make up for the other short-comings as well. Table 1 shows a proposal for a taxonomy.

A further analysis could be made on the basis of the annotation, and the syntactic annotation, in particular. This concerns the parts-of-speech, the morpho-syntactic properties and the dependency functions. As LinES uses the same set of values for both English and Swedish, a simple comparison based on identity of values could be used. This approach is not good enough, though, and especially with nominal and adjectival inflection, and complex verb forms, English and Swedish are sufficiently different to warrant a more elaborate approach. The LinES interface allows the user to specify any pair of identical or contrasting properties in the search interface as constraints on a query. For example to search for nouns that have been translated by verbs, or plurals corresponding to singulars.

6.2 Phrase-level Mappings

The space of possible phrase level mappings is quite large. Some of the types of word-level mappings, such as omissions and additions, are found on the phrase level as well, but other more complex relationships arise. The major types are the following:

- Dispersal: Adjacent tokens can be aligned with non-adjacent tokens,

- Reordering: The order of the images of two tokens may be the reverse of the order of the tokens themselves,
- Head switch: The head of a token can be aligned with a dependent of that token, or with a co-dependent token,
- Restructuring: The image of a phrase need not be a phrase.

Note for instance that what is merely an omission or addition when we consider individual tokens may cause restructuring in the larger context. As an example, consider the sentence *Han behöver träffa en läkare* as the translation of *He needs a doctor*. The image of *needs a doctor* will be the non-adjacent words *behöver, en, läkare* which is not a phrase of the target sentence. Observing that the head fills a gap in the string, this is a case where content words arguably can be added to a phrase, but surely we do not wish to include any addition in a phrase, even if it is close to it.

The image of a headed phrase under alignment may be related to it formally in many different ways. Table 2 is not an exhaustive list, but an illustration of what can be obtained based on implementable criteria. The correspondence of these types to the translation relations recognized by the tradition is not always one-to-one. Moreover, the correspondence is less constrained at the communicative end of the scale. But this is not to be expected as our data does not give information on the concepts needed at that end, such as perspective or situation.

A few technical terms will be needed to understand the definitions. We shall say that a target phrase is *equally sized (modulo function words)* with a source phrase iff every content item of the source is aligned with a content item in the target. Thus, what has happened to the function words is irrelevant; they may be non-translated, expressed by morphemes, or added. Also, a target phrase is structurally similar, or *C-similar*, to a source phrase iff (i) it is equally sized, (ii) two content items that form a dependency are aligned with two content items that form a dependency of the same direction, and (iii) the order of content words is maintained in the translation. Again, the fate of function words in the translation has no bearing on the concept.

Some of the types in table 2 have a close affinity to translation relations, as is evident by their names. Other translation relations can be captured fairly well. For instance, a word-for-word translation with identical source and target parts is likely to be a loan translation. A literal translation can be defined as a C-similar translation, or a translation which is C-similar except for the occurrence of required reorderings. A transposition minimally involves a class shift, while many modulations can be covered by special combinations of the types listed in table 2, e.g., the combination of a property shift regarding diathesis with a subject shift.

Descriptive term	Numerical type	Definition
Word-for-word	$n * (1\text{-}1)$	C-similar translation where all words have been translated one-by-one
Omission	n-0	No part of the phrase is translated
Concentration	n-m, $n > m$	Translation is C-similar with fewer function words
Dilution	n-m, $n < m$	Translation is C-similar with added function words
Reduction	n-m, $n > m$	At least one content item untranslated
Augmentation	n-m, $n < m$	At least one content item introduced in the translation
Downward level shift	n-m, $n > m$	Function word in source corresponds to morpheme in target
Upward level shift	n-m, $n < m$	Function word in target corresponds to morpheme in source
Required reordering	-	At least one change in linear order due to grammatical constraints
Optional reordering	-	At least one change in linear order not required by grammatical constraints
Function shift	-	At least one change in function label
Subject shift	-	The translation of a subject is not a subject
Property shift	-	Some content item shows a change in a property
Class shift	-	C-similar translation with a change in part-of-speech of the head word
Head shift	n-n	Translation is equally sized but some dependency has reveresed its direction
Other	-	Any translation not covered by the defined types

Table 2: A taxonomy of phrase alignment types with definitions.

7 Conclusion

The literature on translation relations provides a rich source of concepts and terms for describing micro-level phenomena in translations, which have been little explored in computational linguistics. While some of them apply at the semantic or pragmatic level, where (current) parallel treebanks contain no relevant data, the structurally oriented ones can be applied for querying parallel data and sorting results. The taxonomies and definitions proposed in this paper, while tentative and subject to further scrutiny, should be seen as a step in that direction.

References

Ahrenberg, L. (2007a). Annotation guidelines for the LinES parallel treebank. URL http://www.ida.liu.se/ lah/transmap/Corpus/guidelines.pdf.

Ahrenberg, L. (2007b). Lines: An english-swedish parallel treebank. In *Proceedings of the 16th Nordic Conference of Computational Linguistics*

(NODALIDA 2007).

Brown, P., J. Cocke, S. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, , and P. S. Roossin (1990). A statistical approach to machine translation. *Computational Linguistics 16(2)*, 79–85.

Catford, J. C. (1965). *A Linguistic Theory of Translation: An Essay in Applied Linguistics*. London, Oxford University Press.

Ingo, R. (2007). *Konsten att översätta: översättningens praktik och didaktik*. Studentlitteratur.

Sågvall Hein, A. (2002). The PLUG project: parallel corpora in Linköping, Uppsala and Göteborg. In L. Borin (Ed.), *Parallel corpora, parallel worlds*, pp. 61–78. Rodopi.

Snell-Hornby, M. (2006). *The Turns of Translation Studies*. John Benjamins.

van Leuven-Zwart, K. (1989). The relation of translation to original. *Target 1(2)*, 100–101.

Vinay, J.-P. and J. Darbelnet (1977). *Stylistique comparée du français et de l'anglais*. Paris, Didier.

The Hunting of the BLARK – SALDO, a Freely Available Lexical Database for Swedish Language Technology

Lars Borin, Markus Forsberg
University of Gothenburg
Department of Swedish

Lennart Lönngren
University of Tromsø
Department of Russian

For the Snark's a peculiar creature, that won't
Be caught in a commonplace way.
Do all that you know, and try all that you don't:
Not a chance must be wasted to-day!
(Lewis Carroll: *The Hunting of the Snark*)

1 Introduction

Among Anna Sågvall Hein's numerous professional interests, computational morphology and the lexicon were among the very earliest. Her implementation of a full computational inflectional morphological processor for Russian was the topic of her PhD thesis (Sågvall, 1973). In the 1980s, she headed the LPS project[1] at the University of Gothenburg. A summary description of her work in the LPS project, on a computational lexicon for parsing is found in Sågvall Hein 1988.

Among her most recent research interests, on the other hand, is the creation of a research infrastructure for Swedish language technology, more specifically a Swedish BLARK[2] (see below).

Against this background, what would be more appropriate than to dedicate this article about a lexicon component of a Swedish BLARK to Anna?

The need for a basic research infrastructure for Swedish language technology is increasingly recognized by the language technology research community and research funding agencies alike. At the core of such an infrastructure is the so-called BLARK – Basic LAnguage Resource Kit. A planning project funded by the Swedish Research Council has been set the tasks of defining

[1] A Lexicon-oriented Parser for Swedish.

[2] See, e.g., Anna Sågvall Hein and Eva Forsbom: *A Swedish BLARK*, presentation at BLARK workshop, 29th January 2006, GSLT retreat, Gullmarsstrand, Sweden ⟨http://stp.lingfil.uu.se/~bea/blark/sveblark060129.pdf⟩.

Swedish BLARK components, surveying the availability and state of the art of these components, and determining the needs for the immediate future. One BLARK component identified during this work as lacking at present is a freely available basic Swedish lexicon with (inflectional) morphological information, containing at least 50,000 lemmas (Andréasson et al., 2007).

Here we describe our work on creating such a lexical resource, which will also contain additional useful linguistic information. This resource is SALDO, or SAL version 2 (see section 2). SALDO currently comprises a semantic lexicon (described in section 3), a morphological lexicon (discussed in section 4), and a computational morphological description written in FM (Forsberg, 2007), including a full-form lexicon generated with FM (treated in section 5). All components will be released under a Creative Commons Attribute-Share Alike license.[3]

2 SALDO: Background and Scope

Svenskt associationslexikon (SAL; Lönngren, 1992) – 'The Swedish Associative Thesaurus' – which formed the basis for SALDO, is a so far relatively little known Swedish thesaurus with an unusual semantic organization.

SAL was created during the years 1987–1992 by Lennart Lönngren and his co-workers in the Center for Computational Linguistics at Uppsala University.[4] SAL has been published in paper form in two reports, from the Center for Computational Linguistics (Lönngren, 1998), and the Department of Linguistics (Lönngren, 1992), both at Uppsala University. Additionally, the headwords and their basic semantic characterizations have been available electronically, in the form of text files, from the very beginning.

The history of SAL has been documented by Lönngren (1989) and Borin (2005). Initially, text corpora were used as sources of the vocabulary which went into SAL, e.g., a Swedish textbook for foreigners and a corpus of popular-scientific articles. A small encyclopedia and some other sources provided the large number (over 3000) of proper nouns found in SAL. Eventually, a list of the headwords from *Svensk ordbok* (SO, 1986) was acquired from the NLP and Lexicology Unit at the University of Gothenburg, and the second paper edition of SAL (Lönngren, 1992) contained 71,750 entries. At the time of writing, SALDO contains 72,557 entries, the increased number being due to some new

[3] ⟨http://creativecommons.org/licenses/by-sa/3.0/⟩

[4] Gunilla Fredriksson worked together with Lennart Lönngren on the lexeme definitions and Ágnes Kilár did most of the programming in the original project. Incidentally, the Center for Computational Linguistics was a research unit created in 1980 on the initiative of Anna Sågvall Hein who was its director until she accepted the chair in Computational Linguistics at Uppsala University in 1990 and the Center was merged with the Linguistics Department. Lennart Lönngren – a Slavist by training, like Anna – acted as her replacement for some years while Anna was acting professor of Natural Language Processing at the University of Gothenburg, which was when she did a great part of her work on the LPS lexicon.

words having been added, but mainly because a number of entries belong to more than one part of speech or more than one inflectional pattern.

The work described here first started in late 2003, when Lars Borin and Lennart Lönngren initiated a collaboration aiming at making SAL available for online browsing through Språkbanken (the Swedish Language Bank at the University of Gothenburg). Part of this work consisted in a formal check of the lexicon, which revealed some circular definitions, that were subsequently removed. In 2005, a computational linguistics student made a prototype graphical interface to SAL (SLV – Språkbanken Lexicon Visualization; Cabrera, 2005). Using this interface, Lennart Lönngren was able to revise a considerable number of entries with respect to their semantic characterization, so that SALDO is in this respect no doubt a new edition of SAL, i.e., also as a semantic lexicon.

We soon realized, however, that in order to be really useful in language technology applications, SAL would have to be provided at least with inflectional morphological information. Thus the work on SALDO began.

3 SALDO: A Semantic Lexicon

As a semantic lexicon, SALDO is a kind of lexical-semantic network, superficially similar to WordNet (Fellbaum, 1998), but quite different from it in the principles by which it is structured.

The organizational principles of SALDO are quite simple – at least superficially – as there are only two primitive semantic relations, one of which is obligatory and the other optional. Every entry in SALDO must have a *mother* (or *main descriptor*), a semantically closely related entry which is more central, i.e., semantically less complex, probably more frequent and acquired earlier in first and second language acquisition, etc. The mother will in practice often be either a hyperonym (superordinate concept) or synonym of the headword (but it need not be either). In order to make SALDO into a single hierarchy, an artificial most central entry, called PRIM, is used as the mother of 50 semantically unrelated entries at the top of the hierarchy, making all of SALDO into a single rooted tree.

Some entries have in addition to the mother an optional *father* (or *determinative descriptor*), which is sometimes used to differentiate lexemes having the same mother.

SALDO (or rather: the underlying SAL) is unusual in several respects:

- it contains a number of proper nouns and multi-word units, not normally found in conventional print or electronic dictionaries;
- it is strictly semantic in its organization; all entries are *lexemes* – i.e., semantic units – and there is no information whatsoever about part of speech or inflection; homonymous entries representing more than one

part of speech are often treated as different, but always because of their semantics and never for inflectional reasons;[5]

- the organizational principles of SALDO are different from those of lexical-semantic networks such as WordNet, in that the semantic relations are more loosely characterized in SALDO. They also differ from those of more conventional thesauruses, however, but in this case by having more, as well as more structured, sense relations among lexemes.

Below, we give a few examples of entries with their mother and father lexemes, randomly selected under the letter "B" of Lönngren (1992):

balkong : hus ('balcony' : 'house')
bröd : mat + mjöl ('bread' : 'food' + 'flour')
brödföda : uppehälle ('daily bread' : 'subsistence')
bröllop : gifta sig ('wedding' : 'get married')
Bulgakov : författare + rysk ('Bulgakov' : 'author' + 'Russian')

How SALDO is different from typical thesauruses becomes apparent when we consider that the two primitive lexical-semantic relations (*mother* and *father*) can form the basis of any number of derived relations. Thus the relation 'sibling having the same mother (as myself)' is very interesting, as such sibling groups tend to correspond to natural semantic groupings.

4 SALDO: A Morphological Lexicon

SAL did not contain any formal information about entries, not even an indication of part of speech. Thus, one important difference between SALDO and SAL is that SALDO now has full information about the part of speech and inflectional pattern of each entry.

The morphological description in SALDO was inspired by that of Sågvall Hein's LPS lexicon, even if the two descriptions differ in many details. The main differences are due to slightly different conceptions of what can constitute an inflectional pattern. In our view the inflectional pattern should be characterized as a set of bundles of morphosyntactic features conventionally expressed inflectionally in a language. We do not make a distinction between affixing and other means of morphological exponence. For instance, Swedish nouns conventionally (maximally) express the following bundles – or combinations – of morphosyntactic features:

singular indefinite nominative
singular definite nominative
singular indefinite genitive

[5]For example, the entry *alternativ* 'alternative' is considered to represent only one lexeme, although in Swedish as in English it has both a noun and an adjective reading.

singular definite genitive
plural indefinite nominative
plural definite nominative
plural indefinite genitive
plural definite genitive

There may be subclasses or individual cases of the main parts of speech which express fewer – or, in rare individual cases, more – such combinations, e.g., nouns appearing only in the singular or only in the plural.

In a non-agglutinating language, words will fall into inflectional patterns according to how the combinations are expressed.[6] Such patterns are conventionally known as declensions in nominal parts of speech and conjugations in verbs.

At the time of writing, with a few hundred multi-word units still to be assigned inflectional patterns and before the final checking of the lexicon, there are more than one thousand different inflectional patterns represented in the lexicon. This does not differ radically from what Sågvall Hein (1988) found when working on the LPS lexicon, although the large number of inflectional patterns in our description is there partly for different reasons than in her case.

For example, there are no artificial "half-paradigms" in SALDO, due to e.g., umlaut plural or ablaut tense formation; these are considered full inflectional patterns in their own right, just as those expressing the same categories by suffixation.

Still, there are many singleton inflectional patterns. In many cases, these are the irregular words of traditional grammar, although there are also subregularities (e.g., among the strong verbs). Surprisingly often, however, the source of plenty is another, viz. variation. We often find that a particular combination of morphosyntactic features – a particular cell in a paradigm – for a word or small group of words can be filled by more than one form, i.e., realized in more than one way. Such cases are legion, e.g., *himmeln*, *himlen*, *himmelen* 'heaven sg def nom' (citation form *himmel*), which in all other respects follows the inflectional pattern designated in SALDO as nn_2u_nyckel, which includes words like *nyckel* 'key', *åker* 'field', *öken* 'desert', *hummer* 'lobster'. This pattern allows only for the first of the three variants shown above for the singular nominative definite form of *himmel*, i.e., the form made by affixing an *-n* to the citation form.

There is an interesting theoretical issue lurking here. For Wurzel (1989, 57), an inflectional class (which he uses as a technical term) must have more than one or even just a few members, although he is not prepared to commit himself to a specific lower limit. A practically useful computational lexicon should in any case specify the morphological behavior of individual words as accurately

[6] Agglutination is – ideally – the case where each morphosyntactic feature is expressed separately and uniformly, so that there will in principle be no inflectional patterns beyond those determined by part of speech.

as possible. In SALDO, this behavior is encoded uniformly for all words – in the form of a unique identifier for each inflectional pattern[7] – i.e., in the lexicon we do not make a distinction between inflectional classes and individual cases in Wurzel's sense. This task is relegated to the computational morphological component, where a mapping is made between SALDO's inflectional patterns and regular, subregular and idiosyncratic inflectional descriptions. However, it is not difficult to get a picture of which inflectional patterns are general and which idiosyncratic. It suffices to extract and count all inflectional pattern designators from SALDO. The following is a list of the inflectional patterns with more than 1000 members in (the still not final version of) SALDO:

9507	nn_3u_film	4278	nn_0u_frid	1216	nn_0n_kaos
7787	nn_2u_stol	3945	nn_1u_flicka	1084	ab_i_inte
5901	av_1_gul	1826	nn_6u_kikare	1046	av_0_korkad
5700	nn_6n_blad	1319	nn_0u_tro	1018	nn_2u_nyckel
5528	vb_1a_laga	1305	nn_5n_ansikte		

These 14 inflectional patterns account for over 50,000 entries or more than 70% of SALDO.

Something which adds to the number of inflectional patterns is that we also encode some inherent features of words in the inflectional pattern designators, features which do not bear directly on the inflectional behavior of the word itself. However, they are potentially useful and comparatively easy to add simultaneously with the morphological information proper, but can be quite difficult to add later, e.g., the gender of nouns, agreement and anaphorical gender in proper names, etc.

In adding the morphological information to SALDO, we have used existing grammatical descriptions of Swedish inflectional morphology – above all *Svenska Akademiens grammatik* (Teleman et al., 1999), but also Hellberg (1978) – as well as the inflectional information provided in existing Swedish dictionaries, primarily *Nationalencyklopedins ordbok* (NEO, 1995; in the form of the lexical database which was used for producing NEO), but also its predecessor *Svensk ordbok* (SO, 1986), and *Svenska Akademiens ordlista* (SAOL, 2006).

For practical and sometimes theoretical reasons we have deviated from these descriptions (which – we should add at this point – are not always consis-

[7]The identifiers were designed to have some internal structure for the benefit of humans working with the lexicon. We cannot for resons of space go into any details here – they will be given in full in the documentation to accompany SALDO – but just to give the reader a flavor of how identifiers are built up: The identifier nn_3u_film below conveys the information that this is a third declension ("3") uter gender ("u") common noun ("nn") inflected like the noun *film* 'film'.

tent among themselves). For example, we have not found Hellberg's "technical stem" a useful concept.[8]

The inflectional information in SALDO deviates from that found in conventional dictionaries at least on two counts:

1. Our inflectional patterns are quite generous as to which forms are supposed to exist for a lemma. We thus subscribe to the notion of "potential form" which is inherent in the concept of inflectional paradigm, the general principle being that there should be a clear(ly statable) grammatical, semantic or pragmatic reason for us to postulate the absence of some form or forms in the paradigm of a lexical item. In practice, this is often the case with number in nouns, comparison in adjectives and certain adverbs, and past (passive) participles in verbs. This principle means that SALDO recognizes fewer uninflected items than traditional dictionaries (e.g., *dna*/*DNA*, which in SALDO also has indeterminate gender, to boot; both *dna:t* and *dna:n* are valid definite forms, according to SALDO). In general, then, if SALDO differs from modern Swedish reference dictionaries, SALDO will accept more forms for a lemma. The only systematic exception to this is that we recognize a class of verbs that do not form past participles, an inflectional pattern that the dictionaries do not seem to recognize at all.[9] Thus, in SALDO, we distinguish between two second-conjugation verb lemmas *väga* 'weigh', where traditional dictionaries recognize only one verb lemma with two meanings, only one of which correlates with the ability to form the past participle:

   väga vb_2a_leva (i.e., *väger*, *vägde*, *vägt*, *vägd*)
   väga vb_2m_väga (i.e., *väger*, *vägde*, *vägt*, –)

2. A lexicon for language technology must lend itself to the analysis of arbitrary free text, e.g., on the internet, where we will find many word forms which are not accepted by normative dictionaries of the written language, but still recognizeable as possible variant inflected forms of some existing lemma. Hence, the SALDO morphology recognizes many attested (but not normative) forms, e.g., the lemma *datum* with utral gender and corresponding double indefinite plural forms in *-ar*/*-er*). We also recognize some variant spellings which are not in the norm, but not "prototypical" misspellings (although the borderline between the two is far from sharp). For example, *microvågsugn* is a quite common spelling for

[8] Hellberg himself says that the technical stem is something that he has had to resort to because of technical shortcomings (Hellberg, 1978, 15f), and which have since then been eliminated.

[9] We are still somewhat more conservative in this respect than Sågvall Hein (1988, 285ff), who distinguishes four classes of (non-phrasal) verbs, according to their ability to form s-forms and the past participle.

mikrovågsugn (prescribed by the orthographic norm). How should we treat this spelling (67,000 Google hits as opposed to 182,000 for the 'correct' spelling, in early December 2007)? In the SALDO morphology it is considered a variant spelling.

SALDO is thus not a normative lexicon, but rather strives to be maximally descriptive. At the same time the notion of inflectional patterns (inflectional classes, paradigms) contains a kind of normativity, namely that which is an irreducible element of linguistics itself, i.e., the formulation of lawlike generalizations about our languages. It is also a recognition of the fact that, however large a corpus we collect, we will never see all the inflected forms of all the entries in our lexicon, not even in a morphologically challenged language like Swedish.[10]

At the same time we know as language users that some forms of some words are not only not attested, but actually non-attestable, e.g., the past participle forms of some verbs, or comparative and superlative forms of (participle-like) Swedish adjectives in *-ad* (e.g., *långfingrad*). The reasons for the lack of some forms in a paradigm can be various, semantic or formal (the latter seems to be the case for the adjectives in *-ad*), but paradigms can also have "holes" in them for completely idiosyncratic reasons (Hetzron, 1975). In the SALDO morphology, we take this into account to some extent, but we have preferred to err on the side of generosity in unclear cases, which means that our morphological description probably overgenerates. If the lexicon is used in language technology applications for analysis, this is not a problem, as long as a potential but impossible form does not coincide with an actual other form. The problem of dealing with this if the lexicon is to be used in natural language generation applications is left for future work.

5 FM for Swedish

Functional Morphology (FM; Forsberg, 2007) is a tool for implementations of computational morphological descriptions. It uses the typed functional programming language Haskell (Jones, 2003) as the description language and supports (compound) analysis, synthesis and compilation to a large number of other formats, including full form lists, XFST (Beesley and Karttunen, 2003) source code, and GF (Ranta, 2004) source code.

FM has been around for a couple of years now and has been successfully applied to a number of languages: Urdu (Humayoun, 2006), Arabic (Smrz, 2007), old and modern Swedish, Italian, Spanish (Andersson and Söderberg, 2003), Russian (Bogavac, 2004), Estonian, and Latin.

The starting point of SALDO's morphological description is the FM implementation of modern Swedish developed by Markus Forsberg and Aarne

[10] On the other hand, this is no more to be expected than you would expect at some point to have seen "all the sentences of the language" as you collect more and more text.

Ranta, which consists of an inflection engine covering the closed word classes and the most frequent paradigms in the open word classes. All in all, the existing implementation covered approximately 80% of the headwords of SALDO, but only less than a tenth of the inflectional patterns, or paradigms.

The remaining paradigms in SALDO, however, are typically variants of already existing paradigms rather than being fundamentally different.

For example, consider the paradigm of *vers*, which is ambivalent in being a second or a third declension noun. It may be implemented as a combination of the inflection functions of second (`nn2`) and third (`nn3`) declension, exemplified in the function `nnvvers`.

```
nnvvers :: String -> Noun
nnvvers vers =  combine (nn2 vers) (nn3 vers)
```

Another example is paradigms with one or more additional word forms, such as the paradigm of *kyrka*, a first declension noun with the additional compound form *kyrko*, as defined in the function `nn1kyrka`.

```
nn1kyrka :: String -> Noun
nn1kyrka kyrka =
  compvariants [tk 1 kyrka ++ "o"] (nn1 kyrka)
```

A complete implementation of a paradigm requires two more things, the interface function of the paradigm and the connection of the interface function to a paradigm identifier (see Forsberg 2007 for details).

The required time to implement a paradigm may vary, but a mean time is around 15 minutes per paradigm. This is a reasonable effort to spend on productive paradigms, but for the paradigms containing just one member it becomes prohibitively large. One possibility would be to resort to a simple enumeration of the worst-case word forms, i.e., the non-derivable word forms. However, we still need these paradigms to be productive, not only for aesthetic reasons, but to be able to treat compounds where these irregular words appear as the head.

We chose to create worst-case paradigms where word forms are described as pairs of suffixes and stem operations. A stem operation is a function from a string to a string, which allows arbitrary computation to be done to the headword before concatenated with the suffix, i.e., we have the same expressiveness as before.

This is best explained with an example, here with the paradigm of *man*, which is one of the more complicated noun paradigms with gemination and umlaut. The stem operation `id` leaves the stem intact, `ge` is a gemination function, and `um` performs umlaut on the stem vowel. We only give the four nominative word forms (variation enclosed in squared brackets), since the genitive is derivable.

```
paradigm "nn_6u_man" ["man"] $
 noun_f Utr
```

```
([(id,"")],                        -- man
 [(ge,"en")],                      -- mannen
 [(um,""),(id,""), (ge,"ar")],     -- män, man, mannar
 [(ge.um,"en"), (ge,"arna")])      -- männen, mannarna
```

The reason why it is considerably faster to implement paradigms with worst-case functions is partly because the whole definition is at a single place, but also because an implementation of a paradigm is decoupled from the rest of the implementation, which makes it more of a mechanical task. The drawback is that the paradigm implementations become less declarative.

The FM implementation of the SALDO morphology is now very close to complete, and the next step will be to ensure the correctness of the implementation.

All paradigms have example headwords in their definition, and FM can compute the inflection tables of these words. The first step will be to proofread these tables. The next step will be to run spellcheckers on the full form word list, which will not only allow us to spot errors in the inflection engine but also in the annotation.

6 Conclusion

So far, there has not been a freely available full-sized computational lexicon for Swedish providing inflectional morphological information for all entries. There are certainly several Swedish computational lexicons in existence, including lexicons with inflectional information, some of them quite large – including Anna Sågvall Hein's LPS lexicon – but none freely available, which SALDO will be. In addition, it will put a new kind of thesaurus at the hands of researchers in Swedish Language Technology, a field pioneered by Anna.

References

Andersson, I. and T. Söderberg (2003). Spanish morphology – implemented in a functional programming language. Master's Thesis, Master's Thesis in Computational Linguistics, Göteborg University.

Andréasson, M., L. Borin, R. Carlson, K. Elenius, E. Forsbom, B. Megyesi, and M. Magnus (2007). An infrastructure for Swedish language technology. Report to the Swedish Research Council Database Infrastructure Committee.

Beesley, K. R. and L. Karttunen (2003). *Finite State Morphology*. Stanford University, United States: CSLI Publications.

Bogavac, L. (2004). Functional Morphology for Russian. Master's Thesis, Department of Computing Science, Chalmers University of Technology.

Borin, L. (2005). Mannen är faderns mormor: *Svenskt associationslexikon* reinkarnerat. *LexicoNordica 12*, 39–54.

Cabrera, I. (2005). Språkbanken lexicon visualization. Rapport de stage. Projet réalisé au Département de Langue Suédoise, Université de Göteborg, Suède.

Fellbaum, C. (Ed.) (1998). *WordNet: An Electronic Lexical Database*. Cambridge, Mass.: MIT Press.

Forsberg, M. (2007). *Three Tools for Language Processing: BNF Converter, Functional Morphology, and Extract*. Ph. D. thesis, Göteborg University and Chalmers University of Technology.

Hellberg, S. (1978). *The Morphology of Present-Day Swedish*. Number 13 in Data linguistica. Stockholm: Almqvist & Wiksell International.

Hetzron, R. (1975). Where the grammar fails. *Language 51*, 859–872.

Humayoun, M. (2006). Urdu language morphology in Functional Morphology toolkit. Master's Thesis, Department of Computing Science, Chalmers University of Technology.

Jones, S. P. (2003, May). *Haskell 98 Language and Libraries: The Revised Report*. Cambridge: Cambridge University Press.

Lönngren, L. (1989). *Svenskt associationslexikon: Rapport från ett projekt inom datorstödd lexikografi*. Centrum för datorlingvistik. Uppsala universitet. Rapport UCDL-R-89-1.

Lönngren, L. (1992). *Svenskt associationslexikon. Del I-IV*. Institutionen för lingvistik. Uppsala universitet.

Lönngren, L. (1998). A Swedish associative thesaurus. In *Euralex '98 proceedings, Vol. 2*, pp. 467–474.

NEO (1995). *Nationalencyklopedins ordbok*. Höganäs: Bra Böcker.

Ranta, A. (2004). Grammatical Framework: A type-theoretical grammar formalism. *The Journal of Functional Programming 14(2)*, 145–189.

Sågvall, A.-L. (1973). *A System for Automatic Inflectional Analysis Implemented for Russian*. Number 8 in Data linguistica. Stockholm: Almqvist & Wiksell International.

Sågvall Hein, A. (1988). Towards a comprehensive Swedish parsing dictionary. In *Studies in Computer-Aided Lexicology*, Number 18 in Data linguistica, pp. 268–298. Stockholm: Almqvist & Wiksell International.

SAOL (2006). *Svenska Akademiens ordlista över svenska språket*. Stockholm: Norstedts Akademiska Förlag.

Smrz, O. (2007). *Functional Arabic Morphology. Formal System and Implementation.* Ph. D. thesis, Charles University in Prague.

SO (1986). *Svensk ordbok*. Stockholm: Esselte Studium.

Teleman, U., S. Hellberg, and E. Andersson (1999). *Svenska Akademiens grammatik, 1–4*. Stockholm: NorstedtsOrdbok.

Wurzel, W. U. (1989). *Inflectional Morphology and Naturalness*. Dordrecht: Kluwer.

The Tectogrammatics of English: on Some Problematic Issues from the Viewpoint of the Prague Dependency Treebank

Silvie Cinková
Eva Hajičová
Jarmila Panevová
Petr Sgall

Charles University in Prague
Institute of Formal and Applied Linguistics

1 Introductory Remarks

The present paper is aimed to illustrate how the description of underlying structures carried out in annotating Czech texts may be used as a basis for comparison with a more or less parallel description of English. Specific attention is given to several points in which there are differences between the two languages that concern not only their surface or outer form, but (possibly) also their underlying structures, first of all the so-called secondary predication (section 3.2). In section 4, we discuss the representations of these constructions in the PDT of Czech as compared with the corresponding annotation in the scenario of a treebank of English (PEDT), being developed in Prague as an English counterpart of PDT (Šindlerová et al., 2007, Bojar et al., 2007).

2 Tectogrammatics

In the Functional Generative Description (see Sgall et al., 1986, Hajičová et al., 1998), tectogrammatics is the interface level connecting the system of language (cf. the notions of *langue,* linguistic competence, I-language) with the cognitive layer, which is not directly mirrored by natural languages. Language is understood as a system of oppositions, with the distinction between their prototypical (primary) and peripheral (secondary, marked) members. We assume that the tectogrammatical representations (TRs) of sentences can be captured as dependency based structures the core of which is determined by the valency of the verb and of other parts of speech. Syntactic dependency is handled as a set of relations between head words and their modifications (arguments and adjuncts). However, there are also the relations of coordination (conjunction, disjunction and other) and of apposition, which we understand as relations of a further dimension. Thus, the TRs are more complex than mere dependency trees.

The TRs also reflect the topic-focus articulation (information structure) of sentences with a scale of communicative dynamism (underlying word order) and the dichotomy of contextually bound (CB) and non-bound (NB) items, which belong primarily to the topic and the focus, respectively. The scale is rendered in the TRs by the left-to-right order of the nodes, although in the surface the most dynamic item, i.e., focus proper, is indicated by a specific (falling) pitch.

In a theoretical description of language, the TRs are seen in a direct relationship to morphemic (surface) structures. This relationship is complicated by many cases of asymmetry – ambiguity, synonymy, irregularities, including the differences between communicative dynamism and surface word order (the latter belonging to the level of morphemics).

The core of a TR is a dependency tree the root of which is the main verb. Its direct dependents are arguments, i.e., Actor, Objective (Patient), Addressee, Origin and Effect, and adjuncts (of location and direction, time, cause, manner, and so on). Actor primarily corresponds to a cognitive (intentional) Agentive, in other cases to an Experiencer (Bearer) of a state or process. If the valency frame of a verb contains only a single participant, then this participant is its Actor, even though (in marked cases) it corresponds to a cognitive item that primarily is expressed by Objective (see (1)).

(1) The book (Actor) appeared.

If the the valency frame of a verb contains just two participants, these are Actor and Objective, which primarily correspond to Agentive and Objective, although the Objective may also express a cognitive item that primarily corresponds to another argument (see (2)).

(2) The chairman (Actor) addressed the audience (Objective)

If the frame contains more than two items, then it is to be distinguished whether the "third" of them is Addressee, Origin, or Effect (cf. the difference between e.g., (3) and (4).

(3) Jim (Actor) gave Mary (Addressee) a book (Objective).
(4) Jim (Actor) changed the firm (Objective) from a small shop (Origin) into a big company (Effect).

In a TR, there are no nodes corresponding to the function words (or to grammatical morphs). Correlates of these items (especially of prepositions and function verbs) are present in the TRs only as indices of node labels: the syntactic functions of the nodes (arguments and adjuncts) are rendered here as functors, and the values of their morphological categories (tense, number, and so on) have the forms of grammatemes. Functors and grammatemes can be understood as indices of lexical items.

In annotating texts from the Czech National Corpus in the frame of the project of the Prague Dependency Treebank (PDT) (Hajič et al., 2006), we work with several specific deviations from theoretically conceived TRs described above. The most important of these deviations is that the tectogrammatical tree structures (TGTSs) we work with in PDT differ from TRs in that they have the form of trees even in cases of coordination; this is made possible by the coordinating conjunctions being handled as specific nodes (with a specific index, here the subscript *coord*, distinguishing between the coordinated items and an item depending on the coordination construction as a whole). Thus, the (primary) TGTS of the sentence (5), with many simplifications, is the tree presented in figure 1:

(5) Mary and Tom, who are our neighbours, have two children.

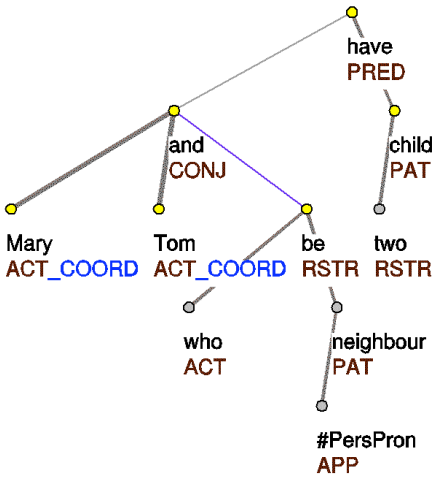

Figure 1

More details are presented in a linearized form of the corresponding TR in (5'); note that (i) every dependent item (or a string of coordinated items) is embedded in its own pair of parentheses, and the functors are present here as subscripts of the parenthesis oriented towards the head, and (ii) the left-to-right order of the nodes, corresponding to the communicative dynamism, differs from the surface word order of the numeral *two*, which is contextually non-bound and is more dynamic than its head noun. Most of the grammatemes are left out.

(5') $((\text{Mary Tom})_{\text{Conj}}\ (_{\text{Rstr}}\ \text{be}\ (_{\text{Obj}}\ \text{neighbour.Plur}\ (_{\text{App}}\ \text{we}))))_{\text{Actor}}\ \text{have}\ (_{\text{Obj}}\ \text{child.Plur}\ (_{\text{Rstr}}\ \text{two}))$

Rstr indicates here a restrictive adjunct, *App* one of Appurtenance (broader than possession), the other abbreviations being self-explaining.

Dependency trees are projective, i.e., for every pair of nodes in which *a* is a rightside (leftside) daughter of *b*, every node *c* that is less (more) dynamic than *a* and more (less) dynamic than *b* depends directly or indirectly on *b* (where *indirectly* refers to the transitive closure of *depend*). This strong condition together with similar conditions holding for the relationship between dependency, coordination and apposition, makes it possible to represent the TRs in a linearized way, as illustrated by (5‘) above. Projective trees thus come relatively close to linear strings; they belong to the most simple kinds of patterning.

3 Selected English Syntactic Constructions for Comparison

3.1 Introduction

A general assumption common to any postulation of a deep (underlying) layer of syntactic description is the belief that languages are closer to each other on that level than in their surface shapes. This idea is very attractive both from the theoretical aspects as well as from the point of view of possible applications in the domain of natural language processing: for example, a level of language description considered to be “common” (at least in some basic features) to several (even if typologically different) languages might serve as a kind of a “pivot” language in which the analysis of the source and the synthesis of the target languages of an automatic translation system may meet (see Vauquois’ known “triangle” of analysis – pivot language – synthesis, Vauquois, 1975).

With this idea in mind, it is then interesting (again, both from the theoretical and the applied points of view) to design an annotation scheme by means of which parallel text corpora can be annotated in an identical or at least easily comparable way. It goes without saying, of course, that the question to which extent a certain annotation scenario designed originally for one language is transferrable to annotation of texts of another language is interesting in general, not just for parallel corpora.

It is well known from classical linguistic studies (let us mention here – from the context of English-Czech contrastive studies – the writings of Czech anglicists Vilém Mathesius, Josef Vachek and Libuše Dušková) that one of the main differences between English and Czech concerns the degree of condensation of the sentence structure following from the differences in the repertoire of means of expression in these languages: while in English this system is richer (including also the forms of gerund) and more developed (the English nominal forms may express not only verbal voice but temporal relations as well), in Czech, the more frequent (and sometimes the

only possible) means expressing the so called second predication is a dependent clause (see Dušková et al., 1994, p. 542 ff.).

It is no wonder then that in our project, secondary predication has appeared as one of the most troublesome issues. In the present section, we devote our attention to one typical nominal form serving for the expression of secondary predication in English, namely infinitive (section 3.2), and look for its adequate representation on the tectogrammatical layer of PDT. The leading idea of our analysis is that we should aim at a representation that would make it possible to capture synonymous constructions in a unified way (i.e., to assign to them the same TGTS, both in the same language and across languages) and to appropriately distinguish different meanings by the assignment of different TGTSs.

The considerations included in the present section of our contribution resulted from our work on a project in which the PDT scenario (characterized above in section 2) was applied to English texts in order to find out if such a task is feasible and if the results may be used for a build-up of a machine translation system (or other multilingual systems); see Šindlerová et al. (2007) and Bojar et al. (2007). This English counterpart of PDT (PEDT) comprises approx. 50,000 dependency trees, which have been obtained by an automatic conversion of the original Penn Treebank II constituency trees into the PDT-compliant a-layer trees (i.e., trees representing the surface shape of sentences). These a-layer trees have been automatically converted into t-layer trees.

3.2 Secondary Predication Expressed by Infinitive

Two classes of constructions are often distinguished: equi-NP deletion and raising. The distinction between the two classes of verbs was already mentioned by Chomsky (1965, pp. 22-23) who illustrated it on the examples (6) and (7):

(6) They expected the doctor to examine John.
(7) They persuaded the doctor to examine John.

Referring to Rosenbaum (1967), Stockwell et al. (1973), p. 521ff., discuss the distinction between *expect* and *require* (which is even clearer than Rosenbaum's distinction between *expect* and *persuade*) and point out that a test involving passivization may help to distinguish the two classes: while (8) and (9) with an equi-verb are synonymous (if their information structure is not considered), (10) and (11) with a raising verb are not:

(8) They expected the doctor to examine John.
(9) They expected John to be examined by the doctor.
(10) They required the doctor to examine John.
(11) They required John to be examined by the doctor.

The authors propose a deep structure indicated by (12) for *expect* (*hate* or *prefer*) and a deep structure that includes an animate object in addition to a sentential object for *require* and *persuade* (see (13)) while it is not important that this NP is then rewritten as S)

(12) They – AUX – VP [V(expect) NP (the doctor examine John)]
(13) They – AUX – VP [V(require) – NP (the doctor) –
NP (the doctor examine John)]

Such a treatment of structures with equi verbs implies that there must be a position in the deep structure which is phonologically null (empty category PRO) and which is coreferential with one of the complementations of the equi verb; in our examples above, it is the object in (6). In theoretical linguistics, this issue is referred to as the relation of control (Chomsky, 1981; see also a detailed cross-linguistic study by Růžička, 1999; for Czech, see Panevová, 1986).

The different behaviour of verbs in the structures verb plus infinitive is discussed also in traditional grammars of English. Quirk et al. (2004) observe a certain gradience in the analysis of three superficially identical structures, namely N_1 V N_2 to-V N_3 (see their Table 16.64a, p. 1216) illustrated by sentences (14), (15) and (16) (their A, B, and C, respectively), each of which conforms to this pattern:

(14) We like all parents to visit the school.
(15) They expected James to win the race.
(16) We asked the students to attend a lecture.
(17) James was expected to win the race.

The authors claim that there is a strong reason to see a clear distinction between (14) and (16): in (14) they postulate a structure in which N_2 functions as the subject of the infinitival clause while in (16) the N_2 should be analyzed as the object of the main clause. However, according to the authors, (15) partakes in both these descriptions: from the semantic point of view, the same analysis as that of (14) would be appropriate; from the structural viewpoint, the analysis similar to that of (16) is preferable. This is supported by the fact that N_2 may become the subject of the passive sentence (17). With this analysis, N_2 behaves like an object in relation to the verb of the main clause and like a subject in relation to the infinitival clause. The authors use the term raised object to characterize this situation and they support their analysis by several criteria.

4 Solutions Proposed

4.1 Subject Raising

In the scenario of PEDT (the Prague English Dependency Treebank), the distinction between the structures with the so-called raising verbs and control verbs is preserved. The sentence (18) (see figure 2) is a typical example for the subject raising construction in English, see also a possibility of (18a) in English:

(18) John seems to understand everything.
(18a) It seems that John understands everything.

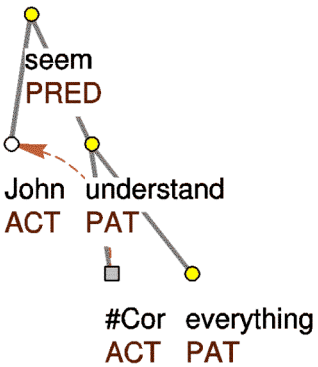

Figure 2

However, its Czech counterpart *zdát se* is connected with certain constraints: this verb must be determined by verbo-nominal (or only nominal) complement, see ex. (19). With verbo-nominal complement it has an analogical structure to the English example in figure 2, see figure 3. These constraints, however, eliminate this verb from the "pure" raising constructions; see also the unacceptability of (20) in Czech:

(19) Jan se zdá (být) smutný.
Lit. John Refl. he-seems (to-be) sad.
(20) *Jan se zdá rozumět.
Lit. John Refl. he-seems to-understand

In English, the modal and phase verbs are considered as belonging to the class of subject raising verbs. In the PDT scenario (as well as in the theoretical framework for it, FGD) most of these verbs are treated as auxiliaries, and their modal meanings are described by morphological grammatemes assigned to the autosemantic verb. As for modal verbs, this

approach is adopted for PEDT as well (see Cinková et al., 2006, p. 88f.). This approach is planned for the treatment of phase verbs, too (*Jan začal pracovat* [John started to work], *Jan začínal pracovat* [John was going to start to work]) could be described as multi verbal predicates).

The underlying structure proposed for subject raising constructions in Czech as well as in English is, however, identical to the control verb constructions, where ACT (i.e., the first argument of the control verb) controls Sb (subject) of the infinitive clause (see section 4.3).

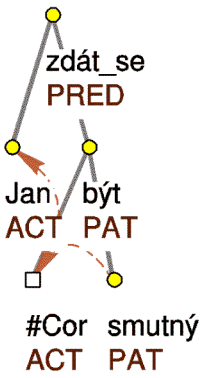

Figure 3

4.2 Object Raising

The English verbs used as clear examples of object raising verbs have no Czech counterparts with infinitive constructions; cf. (21) and figure 4 for English:

(21) John expects Mary to leave.

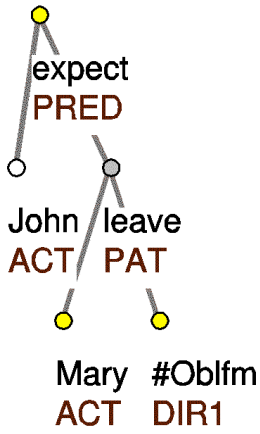

Figure 4

However, the subclass of verbs displaying this operation, called sometimes ECM (exceptional case marking), share this behavior with Czech constructions of *accusativus cum infinitivo* (AccI in sequel). It concerns the verbs of perception (see (22a) and figure 5 for English and (22b) and figure 6 for Czech):

(22a) John hears Mary cry/crying.
(22b) Jan slyší Marii plakat.

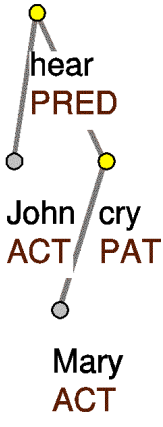

Figure 5

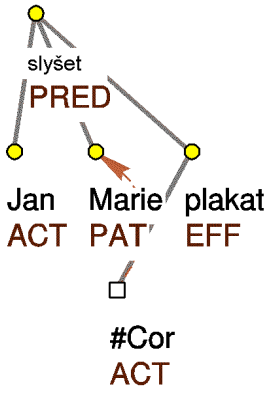

Figure 6

There are two possible ways to reflect the underlying structures of these sentences. The approach (A) is influenced by the English tradition: The verbs of perception proper (such as *to see, to hear*) are understood in English as two-argument structures; if their second argument is expressed by secondary predication, the first argument of the secondary predication is raised up and it receives ("exceptionally") the Accusative form. The structure given in figure 5 would yield the surface structure (22a) as well as the surface structure (22c):

(22c) John hears that Mary cries.
(22d) Jan slyší, že Marie pláče.

However, the synonymy illustrated by (22a) and (22c) does not hold in all contexts, see (23a), (23b), (23c) and (23d), and also (24a) and (24b):

(23a) Jan slyšel, že Carmen zpívá Dagmar Pecková.
Lit. Jan heard that Carmen-Acc sings Dagmar Pecková
(23b) Jan slyšel, že Dagmar Pecková zpívá Carmen.
Lit. Jan heard that Dagmar Pecková sings Carmen
(23c) Jan slyšel Dagmar Peckovou zpívat Carmen.
Lit. Jan heard Dagmar Pecková to-sing Carmen
(23d) ?Jan slyšel Carmen zpívat Dagmar Peckovou.
Lit. Jan heard Carmen-Acc to-sing Dagmar Peckova-Acc
(24a) Jan slyšel tu skladbu hrát kapelu Olympic.
Lit. Jan heard the piece-Acc to-play the band Olympic-Acc
(24b) Jan slyšel, že/jak tu skladbu hraje kapela Olympic.
Lit. Jan heard that/how the piece-Acc plays the band Olympic-Nom

In the pairs (23a), (23b) vs. (23c), (23d) the difference between the meanings of the polysemic verb *slyšet* [to hear] is reflected: while in (23a) and (23b) Jan is either the direct hearer of the singing or he may be only told about the singing, in (23c) and (23d), if it is possible at all, he must be a direct listener.

Moreover, the possible pre-posing of the object of the dependent clause (see (23a) and (24a) for Czech) has no counterpart in English.

In the approach (B) reflecting the situation in Czech the verbs of pereception are understood as three-argument structures with the underlying structure given in figure 6 corresponding to the sentence (22d), which differs from the underlying structure of ex. (22c) given in figure 5.

Under the approach (A), the formulation of the conditions under which the secondary predication could be nominalized by an infinitive clause seems to be very complicated while with the approach (B) the raised object is understood as a part of a cognitive operation, the result of which is manifested on the level of underlying structure.

4.3 Control (Equi) Verbs

As for the control verbs, the underlying structure proposed for Czech seems to be suitable for the PEDT scenario as well, see (25), (26) and figure 7, 8. A special node with lemma *Cor* is used for the controllee and an arrow leads from this node to its controller. The list of the verbs sharing the attribute of control will be nearly identical for both languages.

(25) John refused to cooperate.
(26) The editor recommended the author to correct the errors immediately.

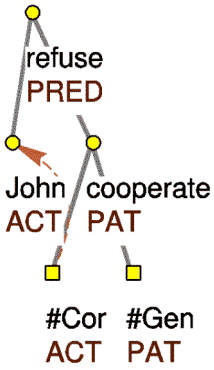

Figure 7

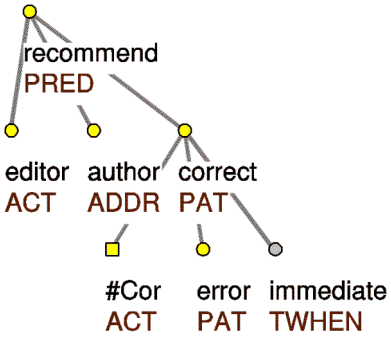

Figure 8

We have concluded that though the notions of raising and control are assumed not to be theory dependent and therefore applicable in both scenarios (for PDT as well as for PEDT), the differences between these two classes are not substantial (and they seem to be overestimated in the theoretical works).

4.4 Nominal Predicates

Analogical control constructions appear with some adjectives in the position of the nominal predicates in sentences with copula, see (27), (28) and figure 9 for English:

(27) John is eager to please.
(28) John is eager to be pleased.

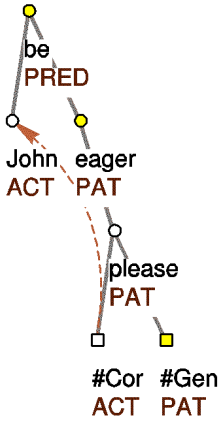

Figure 9

The corresponding underlying structures for Czech sentences (29a), (30a) are similar to those for English (29b), (30b):

(29a) Jan je schopen to udělat.
(29b) John is able to do it.
(30a) Jan je ochoten být očkován.
(30b) John is willing to be vaccinated.

However, the list of English adjectives complemented by an infinitive clause is wider than in Czech. In (31), (32) and figure 10 a control between ACT and the Sb of infinitive clause could be seen:

(31) She was quick to shut the door.
(32) Bob was reluctant to respond.

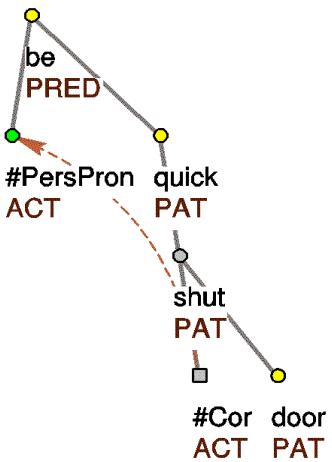

Figure 10

4.5 Tough Movement

The object-to-subject raising (sometimes called tough movement) takes place with some evaluative adjectives in complex predicates, see (33a) and its transformed version after the raising operation (33b, figure 11). This type of raising has no counterpart in Czech.

(33a) It is difficult to please John.
(33b) John is difficult to please.

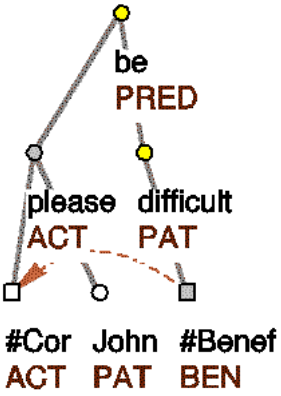

Figure 11

4.6 Causative Constructions

Causativity of constructions such as (34) and (35) is expressed by the lexical meanings of the "semiauxiliaries" *to make, to get, to have* and by the secondary predication denoting the caused event filling the position of the PAT(ient) of the semiauxiliary causative verb.

(34) John made Mary stay.
(35) John had Mary clean the window.

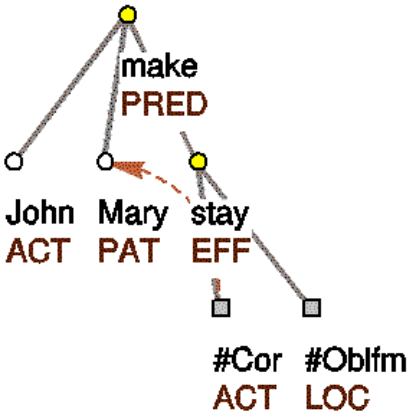

Figure 12

The constructions with the Czech verb *nechat* [to let] and the analogical underlying structure (with raised subject-to-object position) correspond to this type of causativity.

5 Conclusions

In our contribution, we have briefly discussed certain issues of secondary predication in which English differs from Czech with the result that most of them probably can be handled without differences in underlying structures of the two languages.

There are, of course, other cases in which the TRs of the two languages certainly differ. We want only to note here that not all such differences concern syntactic relations (functors). Thus in the case of such grammatical categories as definiteness or as tense and verbal aspect the differences can be captured by distinctions in the repertoires and values of grammatemes (representing morphological values).

Acknowledgements

This work was funded by GA-CR 405/06/0589, MSM 0021620838, and in part by the Companions project (www.companions-project.org) sponsored by the European Commission as part of the Information Society Technologies (IST) programme under EC grant number IST-FP6-034434.

References

Bojar, O., S. Cinková and J. Ptáček (2007). Towards English-to-Czech MT via tectogrammatical layer. In *Proceedings of the 6th International Workshop on Treebanks and Linguistic Theories (TLT 2007)*, pp. 7–18. Bergen, Norway: Northern European Association for Language Technology.

Cinková, S., J. Hajič, M. Mikulová, L. Mladová, A. Nedolužko, P. Pajas, J. Semecký, J. Šindlerová, J. Toman, Z. Urešová, and Z. Žabokrtský (2006). Annotation of English on the Tectogrammatical Level. Technical report UFAL TR 2006-35. Prague.

Čmejrek, M., J. Cuřín, J. Havelka, J. Hajič and V. Kuboň (2005). Prague Czech-English Dependency Treebank Version 1.0. *EAMT 2005 Conference Proceedings,* pp. 73–78.

Chomsky, N. (1981). Lectures on Government and Binding. Dordrecht: Foris.

L. Dušková et al. (1994). Mluvnice současné angličtiny na pozadí češtiny [Grammar of Present-Day English on the Background of Czech], Academia, Prague.

Hajič, J., J. Panevová, E. Hajičová, P. Sgall, P. Pajas, J. Štěpánek, J. Havelka, M. Mikulová, Z. Žabokrtský, and M. Ševčíková-Razímová (2006). Prague Dependency Treebank 2.0. CD-ROM. Linguistic Data Consortium, Philadelphia, PA, USA. LDC Catalog No. LDC2006T01 URL <http://ufal.mff.cuni.cz/pdt2.0/>, quoted 2008-12-02.

Hajičová, E., B. H. Partee and P. Sgall (1998). *Topic-Focus Articulation, Tripartite Structures and Semantic Content*. Dordrecht: Kluwer.

Mikulová, M., A. Bémová, J. Hajič, E. Hajičová, J. Havelka, V. Kolářová, L. Kučová, M. Lopatková, P. Pajas, J. Panevová, M. Razímová, P. Sgall, J. Štěpánek, Z. Urešová, K. Veselá and Z. Žabokrtský (2006). Annotation on the tectogrammatical level in the Prague Dependency Treebank. Annotation manual. Tech. Report 30 ÚFAL MFF UK. Prague.

Panevová, J. (1986). The Czech infinitive in the function of objective and the rules of coreference. In: J. L. Mey (ed.) *Language and Discourse: Test and Protest*, pp. 123–142. Amsterdam: Benjamins.

Quirk, R., S. Greenbaum, G. Leech and J. Svartvik (2004). *A Comprehensive Grammar of the English Language*. Longman. First published 1985.

Rosenbaum, P. S. (1967). *The Grammar of English Predicate Complement Constructions*. The MIT Press, Cambridge, Mass.

Růžička, R. (1999). *Control in Grammar and Pragmatics*. Amsterdam/ Philadelphia: Benjamins.

Sgall, P., E. Hajičová and J. Panevová (1986). *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. Dordrecht: Reidel Publishing Company and Prague: Academia.

Stockwell, R. P., P. Schachter and B. Hall Partee (1973). *The Major Syntactic Structures of English*. Holt, Winehart and Winston, New York.

Šindlerová, J., L. Mladová, J. Toman and S. Cinková (2007). An application of the PDT scheme to a parallel treebank. In *Proceedings of the 6th International Workshop on Treebanks and Linguistic Theories (TLT 2007)*, pp. 163–174. Bergen, Norway: North European Association for Language Technology.

Vauquois, B. (1975). Some problems of optimization in multilingual automatic translation. First National Conference on the Application of Mathematical Models and Computers in Linguistics. Varna, May 1975.

The Abstract-Concrete Syntax Distinction and Unification in Multilingual Grammar*

Robin Cooper

University of Gothenburg
Department of Linguistics

1 Introduction

We will characterize a notion of multilingual grammar based on work conducted in the Grammatical Framework (GF), developed by Aarne Ranta (2004, 2007), Perera and Ranta (2007) and Regulus, developed by Manny Rayner and others (Rayner, Hockey, and Bouillon, 2006; Santaholma, 2007). We will suggest a way of constructing multilingual grammars using TTR (type theory with records, Cooper 2005a, 2005b, in preparation) which combines the GF and Regulus approaches to multilingual grammar with aspects of unification-based grammar such as HPSG (Sag et al., 2003).

2 Multilingual Grammar

One approach to multilingual grammar development is the porting of one grammar to a grammar for another (related) language as discussed in Kim et al. (2003) and Bender et al. (2005). This approach is useful when you have a large coverage grammar for one language and want to create a similar large coverage grammar for another language. The approach we will explore here is the creation of grammars that are simultaneously grammars for more than one language from the outset. This is useful when you are concerned with small coverage grammars for a particular application (for example, speech recognition in a dialogue system) and wish to maintain exactly comparable linguistic functionality in more than one language. It is this second approach to multilingual grammar that both GF and Regulus are concerned with. In GF the emphasis is on *abstract* and *concrete syntax* (several different concrete syntaxes

*I would like to thank Anna Sågvall Hein for many years of happy and productive collaboration since I joined the Swedish language technology community in 1995. Her support, help and plain hard work accompanied by unfailing good cheer have meant a great deal. This work was supported by Vetenskapsrådet project 2005-4211 (Library-based Grammar Engineering) and Riksbankens Jubileumsfond project 2008-2010 (Semantic Coordination in Dialogue).

corresponding to a single abstract syntax) and the use of resource grammars to express specific characteristics of particular languages. In Regulus, as described by Santaholma (2007), the emphasis is on *grammar sharing* and the use of macros to give abstract representations of grammatical phenomena that can be realized in different ways in different languages. Here we will concentrate on three important ways in which the GF and Regulus approaches are similar:

abstract representation use of abstract representations of natural language which can be shared between different languages to a larger extent than the concrete or surface representations of the languages and which can serve to represent meanings

interlingua and transfer provision of facilities for both an interlingual and a transfer approach to translation

resource grammars abstract representation of grammatical information allowing the concrete or surface form of the language to be specified by an API-like formalism which can be interpreted in different ways depending on which language's grammatical resource is being used

I shall illustrate how each of these three aspects are realized in TTR using the grammar TOY1 from Rayner et al. (2006). I shall first give an intuitive characterization of the general grammar architecture that GF employs and how this is realized using TTR (section 3). The architecture presents a pretty picture of how languages can be related in the framework using an interlingua architecture for limited domain multilingual grammars. However, mismatches between languages arise in even the smallest of domains (such as TOY1) and I will illustrate some of the simple problems that appear to disturb the picture even for closely related languages such as English and Swedish (section 4). In section 5 I will suggest that TTR provides us with reasonably elegant techniques for treating these mismatches without unduly disturbing the elegant interlingual picture.

3 The GF Multilingual Grammar Architecture

At the core of the GF approach is the relation between abstract and concrete syntax as illustrated in Fig. 1. Abstract syntax, like Curry's (1961) tectogrammar, provides a representation of the function argument structure underlying the natural language expressions presented in concrete syntax, corresponding to Curry's phenogrammar, where individual language aspects such as word order and morphology are represented. Thus abstract syntax can be shared by languages which might have radically different realizations in concrete syntax. It is for this reason that abstract syntax seems to be a good candidate for an interlingua in at least restricted domain grammars and, since it is expressed in

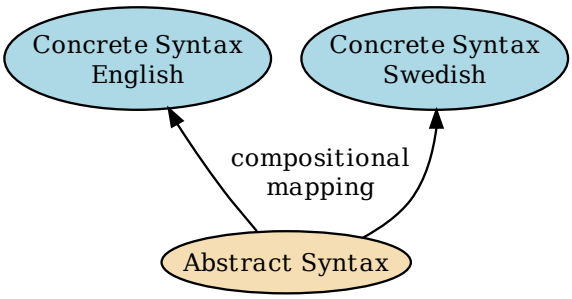

Figure 1: Abstract and concrete syntax

terms of function argument structure, can also do duty as semantic representation of the particular domain, representing what kinds of objects are available, what properties they have and what relations may hold between them, i.e., *ontologies* in the sense that this term is employed in work on knowledge representation and exploited in OWL (Web Ontology Language).[1] The concrete syntax is defined compositionally on the abstract syntax. That is, for each object, class of objects, property or relation in the abstract syntax there will be an expression in the concrete syntax and for each way of applying or constructing objects in the abstract syntax there will be a corresponding combination rule of expressions in the concrete syntax. For example, in the domain discussed in this paper, there is a class called `Light` and another called `Fan` and a function `SwitchOn` which maps objects of these classes to an object which is an `Action`. In the concrete syntax the phrases *(the) light(s)* and *(the) fan(s)* are related to the class `Light` and `Fan` respectively and the phrase *switch on* to the function `SwitchOn`. The phrase *switch on the light* is related to the action which results from applying the function `SwitchOn` to a particular member of the class `Light`.

An important aspect of our particular realization of the GF architecture is that we exploit the similarity of record types to feature structures. We construct grammars in which information from the abstract and concrete levels are added together. That is, grammars are defined in terms of structures which simultaneously represent abstract and concrete syntax in a manner which is similar to the use of signs in HPSG. This will play an important role in our strategy for dealing with the kind of mismatches discussed in section 4. Thus, for example, *light* in abstract syntax will be:[2]

[1] See for example `http://www.w3.org/TR/owl-guide`.

[2] Record types are represented as records with the label `rectype` in the Oz programming language (`http://www.mozart-oz.org`). `sintype(T a)` represents a singleton subtype of the type `T` whose unique element is `a`. Dependent types such as `light([x])` are represented in an abbreviated form which is slightly imprecise but more readable than the official form which need not concern us in this paper.

```
rectype(cat:sintype('Cat' n)
        domain:sintype('RecType'
                       rectype(o_device:device([x])
                               o_light:light([x])
                               o_switchable:switchable([x])
                               x:'Ind'))
        id:sintype('Id' 'Light'))
```

This is a singleton type whose unique element is the record:

```
rec(cat:n
    domain:rectype(o_device:device([x])
                   o_light:light([x])
                   o_switchable:switchable([x])
                   x:'Ind')
    id:'Light')
```

Thus the abstract grammar provides category information, the domain type associated with the item and an id which in complex elements will indicate the construction tree of the item according to the abstract syntax. For complex items it will also provide information about the item's daughters in a 'daughters' field (corresponding to the daughters-feature in HPSG grammars). In the implementation we are discussing here we have actually derived the type by combining two separate resources, one which provides the domain information and one which provides a way of relating the domain information to syntactic information, presented below.

```
rectype(o_device:device([x])
        o_light:light([x])
        o_switchable:switchable([x])
        x:'Ind')

rectype(cat:sintype('Cat' n)
        id:sintype('Id' 'Light'))
```

The grammar for concrete syntax imports information from the abstract syntax and adds to it. Thus according to the grammar for the English concrete syntax the type associated with *light* is

```
rectype(agr:rectype(num:sintype('Number' sg))
        cat:sintype('Cat' n)
        domain:sintype('RecType'
                       rectype(o_device:device([x])
                               o_light:light([x])
                               o_switchable:switchable([x])
                               x:'Ind'))
        id:sintype('Id' 'Light')
        phon:sintype('Phon' [light]))
```

Here agreement and phonology[3] information have been added to the type provided by the abstract syntax. The concrete syntax for Swedish is derived by adding different information to the same abstract representation:

```
rectype(agr:rectype(def:sintype('Definiteness' yes)
                    gen:sintype('Gender' utr)
                    num:sintype('Number' sg))
        cat:sintype('Cat' n)
        domain:sintype('RecType'
                       rectype(o_device:device([x])
                               o_light:light([x])
                               o_switchable:switchable([x])
                               x:'Ind'))
        id:sintype('Id' 'Light')
        phon:sintype('Phon' [lampan]))
```

This view of an approach to limited domain multilingual grammars in which the different languages share a common interlingual representation is very attractive for a number of reasons. It appears not to have the central problem that the interlingual approach to general translation has, that is, that the interlingua has to make all the distinctions that can be relevant for any language that is connected to it. Here the problem is not to construct adequate interlingua for all languages. Rather it is to take a given characterization of a domain and relate it to different natural languages. The engineering advantage for building multilingual applications is great. We need to make sure that the linguistic coverage is exactly similar in each language if we are, for example, building a dialogue system to interact with an MP3 in a car. By ensuring that we cover the abstract syntax in each language we can be sure that we have exactly corresponding coverage in each of the languages. The work on the domain and abstract syntax can be carried out by different people and the work on relating the abstract syntax to the concrete syntax need not be carried out by the designer of the domain representation. In addition the definition of the concrete syntaxes uses general definitions given in a resource grammar which could be implemented by a linguist who is only concerned with a general characterization of the language and not with any of the work on specific domains. The line in the code for the English concrete syntax which defines *light* is[4]

```
Light = {ResGram.light AbsDom.light}.sg
```

The English resource grammar provides a function `ResGram.light`[5] which applies to the abstract syntax representation `AbsDom.light` (on page 4) to

[3] We follow the honourable HPSG tradition of using lists of words as phonological representations.

[4] This uses the Oz notation `{F A}` which represents the application of function `F` to argument `A`.

[5] That is, the value of the variable `Light` in the Oz module to which the variable `ResGram` has been set, i.e., the module containing the resource grammar for English. Similar remarks hold for `AbsDom.light`.

produce a paradigm for the English noun *light*. This line of code sets the variable `Light` to the entry for singular in that paradigm. The corresponding line in the Swedish concrete syntax is[6]

```
Lampan = {ResGram.lampa AbsDom.light}.sg.def
```

Here the singular definite entry in the paradigm is chosen.

The approach represented by the GF architecture offers the opportunity for highly structured modular code and the construction of very high level and powerful formalisms for manipulating grammars. However, translating between natural languages is a messy business and there are many pitfalls which appear initially to disturb the pretty picture of a common abstract syntax related to several concrete syntaxes, even if we are only looking at tiny domains. We shall address some of these problems in section 4.

4 Mismatches between Languages

The coverage of the TOY1 example grammar that is introduced in Rayner et al. (2006) is represented by:

turn/switch on/off the light(s)/fan(s) (in the kitchen/living-room)
dim the light(s) (in the kitchen/living-room)
is the light/fan (in the kitchen/living-room) (switched) on/off
are the lights/fans (in the kitchen/living-room) (switched) on/off[7]

The coverage for the corresponding Swedish grammar is represented by:

tänd/släck lampan/lamporna (i köket/vardagsrummet)
light/extinguish light[def]/lights[def] (in kitchen[def]/livingroom[def])
vrid ner lampan/lamporna (i köket/vardagsrummet)
turn down light[def]/lights[def] (in kitchen[def]/livingroom[def])
sätt på/stäng av fläkten/fläktarna (i köket/vardagsrummet)
put on/close off fan[def]/fans[def] (in kitchen[def]/livingroom[def])
är lampan (i köket/vardagsrummet) tänd/släckt
is light[def] (in kitchen[def]/livingroom[def]) lit[sg]/extinguished[sg]
är lamporna (i köket/vardagsrummet) tända/släckta
are lights[def] (in kitchen[def]/livingroom[def]) lit[pl]/extinguished[pl]
är fläkten/fläkterna (i köket/vardagsrummet) på/av
is/are fan[def]/fans[def] (in kitchen[def]/livingroom[def]) on/off

Despite the extremely small size of this coverage it contains a number of challenges to the idea that both languages should be related to the same abstract syntax. The core of the definition of the abstract syntax (prior to combination with domain information) is given in Fig. 2. We will now detail some

[6] Here `ResGram` is set to the module containing the Swedish resource grammar.

[7] The TTR grammar we are discussing in this paper does not account for the optional occurrences of *switched* in the third and fourth lines.

```
DefArt = {Lex 'DefArt' det}
Light = {Lex 'Light' n}
Fan = {Lex 'Fan' n}
Kitchen = {Lex 'Kitchen' n}
Livingroom = {Lex 'Livingroom' n}
In = {Lex 'In' prep}
SwitchOn = {Lex 'SwitchOn' v}
SwitchOff = {Lex 'SwitchOff' v}
On = {Lex 'On' part}
Off = {Lex 'Off' part}
Be = {Lex 'Be' cop}
Dim = {Lex 'Dim' v}

S_VP = {RuleUnary s_vp s vp}
S_Cop_SmallCl = {RuleBinary s_cop_smallcl s cop smallcl}
SmallCl_NP_Part = {RuleBinary smallcl_np_part smallcl np part}
NP_Det_N = {RuleBinary np_det_n np det n}
N_N_PP = {RuleBinary n_n_pp n n pp}
PP_Prep_NP = {RuleBinary pp_prep_np pp prep np}
VP_V_NP = {RuleBinary vp_v_np vp v np}
```

Figure 2: Abstract syntax (without domain information)

challenges that arise in the relating both coverages to this abstract syntax.

1. **Phrases in concrete syntax can correspond to single items in abstract syntax.** In the abstract syntax there are single verb elements corresponding to the single domain function `SwitchOn`, `SwitchOff` and `Dim`. However, in English `SwitchOn` and `SwitchOff` correspond to phrases such as *switch on* and *turn off*. In Swedish these items correspond sometimes to a single word (*tänd* or *släck*) and sometimes to a phrase (*sätt på* or *stäng av*). `Dim` corresponds to a phrase in Swedish (*vrid ner*).

2. **Single words in concrete syntax can correspond to complex constructions in abstract syntax.** Swedish definite nouns can be used to form a complete noun-phrase corresponding to a tree of the form $[_{NP}$ *Det N]* in the abstract syntax.

3. **Single items in abstract syntax are not covered by a single item in concrete syntax.**. `SwitchOn` and `SwitchOff` are functions in our domain characterization which apply to both lights and fans. This corresponds well to the concrete syntax of English where both lights and fans can be switched (or turned) on and off. This is not so in Swedish, however. The words *tänd* and *släck* are used for turning lights on and off respectively but cannot be used for fans.[8] For fans *sätt på* and *stäng*

[8]The phrase *tänd fläkten* is grammatical in the language at large but means *set fire to the fan* which is not normally required in smart house applications, for example.

av are used, but these phrases are not normally used for lights.[9] Similar remarks hold for the Swedish for *on* and *off* in sentences like *Is the light/fan on/off?*.

4. **A single category in abstract syntax has to be split in concrete syntax.** The items *on* and *off* are classified as particles with category `part` in our abstract syntax. When you ask whether the light in the kitchen is on/off in Swedish it is natural to use the (deverbal) adjectives *tänd*/*släckt* in Swedish. Predicate adjectives need to agree in number and gender with their subject. If particles in this position were always represented by adjectives in Swedish, this would not cause a great problem. We could use the name of the category from the abstract syntax but make the syntax obey the agreement rules for adjectives specified in the Swedish resource grammar. Thus Swedish adjectives would just get a rather odd category name in this particular abstract syntax. However, the matter is not so simple since when we are asking whether fans are on and off we do use particles in Swedish (*på* and *av*). Particles do not inflect and thus a grammar that tried to make them agree with their subject would fail. Thus in the Swedish concrete syntax we need to split the category `part` from the abstract syntax, using `adj` when we are talking about lights and `part` when we are talking about fans.

These are the kinds of problems that lead to the use of transfer approaches to translation in place of interlingual approaches. In the Regulus system described in Rayner et al. (2006), facilities are provided for the development of both transfer and interlingual multilingual grammars but the approaches cannot be put together in a single grammar. In section 5 I want to suggest that such a combination is desirable and that the pretty picture provided by the GF interlingual architecture can be maintained by incorporating certain elements of transfer into it.

5 Integrating Elements of Transfer into an Interlingual Architecture

The kind of solution to these problems that I will suggest here is perhaps different from the discussion of transfer in the standard GF literature[10] in that I exploit the fact that in TTR we are constructing grammars by successive refinement, that is, adding information about concrete syntax to the abstract syntax. Since we are carrying along all the abstract syntax information in the concrete syntaxes we also have the opportunity to carry out refinements on the abstract

[9] The verb *sätt på* is more possible for lights than *stäng av* perhaps.

[10] See discussion of the GF transfer language by Björn Bringert on `http://www.cs.chalmers.se/~aarne/GF/doc/transfer.html`.

syntax information which is included in the refined grammar. We will take each of the challenges of section 4 in turn.

1. **Phrases in concrete syntax can correspond to single items in abstract syntax.** We allow for non-compositional compounding of words, e.g., in the English concrete syntax we define *switch on* by

```
V_Switch_On =
{UResSyn.nonCompositionalCompoundWords
 Switch On
 {UResSyn.imperativeLex AbsDom.switchOn}}
```

   This provides us with a rule which will combine only the specific items `Switch` and `On` to form a verb with the abstract information from `SwitchOn`. The tree will correspond to the syntactic form *[$_V$ V Part]* but abstract information associated with the mother node will correspond to that associated with a single item in the abstract syntax and any abstract information associated with the daughters will be disregarded. The function `NonCompositionalCompoundWords` which does this is provided in the universal syntax resource module (represented here by `UResSyn`) where general syntactic resources for the construction of grammars are made available.

2. **Single words in concrete syntax can correspond to complex constructions in abstract syntax.** We allow for non-branching rules in concrete syntax to associate abstract information with the mother which results in the application of an abstract syntax rule to an argument. For example, the Swedish syntax rule which allows a single noun to be a noun-phrase in this domain grammar[11] is defined by

```
NP_N = {ResGram.nP_N
        {UResAbs.applyRule
         AbsDom.nP_Det_N
         AbsDom.defArt}}
```

   Recall that what the resource grammar provides are functions which will apply to abstract information and return a rule that incorporates this abstract information alongside the concrete information about syntax and morphology. Here the Swedish grammar resource for the rule which constructs an NP from a single noun is provided with an argument which is the result of applying an abstract rule for NPs containing a determiner to an abstract argument representing a definite determiner. This is facilitated by the fact that rules are represented as functions which take

[11] Note that there are only definite NPs in this small grammar so we do not have to worry about issues of definiteness here.

one argument at a time, similar to the style of much categorial grammar. Thus, in much simplified form, a rule corresponding to

$$\text{NP} \rightarrow \text{Det N}$$

is a function

$$\lambda x : \text{Det}\ \lambda y : \text{N}\ [_{\text{NP}}\ x\ y]$$

which takes an element of type *Det* and returns a function which will map an element of type *N* to an NP-structure. This means that applying this function to a determiner will give us in effect a rule that constructs NPs out of nouns, as desired. Notice that this is exploiting the fact that TTR has functions in addition to record types. Thus we have both functional and unification-like behaviour.

3. **Single items in abstract syntax are not covered by a single item in concrete syntax.** In the Swedish concrete syntax the word *tänd* 'switch on (for lights)' is characterized by

```
Tänd = {ResGram.tända
        {RestrictFunction AbsDom.switchOn
                          Dom.light}}.imp
```

Here the Swedish resource grammar function corresponding to *tända* is not provided with the abstract information corresponding to `SwitchOn` but rather a modification of this. In the abstract syntax `SwitchOn` corresponds to a function which applies to any device (i.e., both fans and lights). We restrict this function so that it only applies to lights (as characterized in the domain description, `Dom`). The operation that carries out the restriction (represented here as `RestrictFunction`) is provided by a module which is a universal resource for domain information definition. Note that this strategy again relies on the fact that we are carrying over abstract syntax information to the concrete syntax grammar and thus have an opportunity to refine it on the way. Since the domain of the function is characterized in terms of a record type as is also the domain class `Dom.light`, restricting the domain of the function involves applying the operation that corresponds to unification on the function domain type and `Dom.light`. Thus we are simultaneously enjoying the fruits of unification and functional approaches.

4. **A single category in abstract syntax has to be split in concrete syntax.** The adjective *tänd* is characterized in the Swedish concrete syntax by

```
TändAdj = {ResGram.tänd
           {SetCat
```

```
                    {Restrict AbsDom.on Dom.light}
                    adj}}.sg.indef.utr
```

Here the abstract syntax information builds on that provided by *on* but restricted to lights in a similar manner to the previous example. However, in addition here we need to set the category information with the operation `SetCat` provided by the universal domain resource module. This is because the category associated with *on* in the abstract syntax is `part` whereas what we need is `adj`. This is the only non-monotonic (or destructive) operation that we have suggested here in that it actually removes information that was present in abstract syntax before. But, of course, since we are incorporating the information into our grammar for concrete syntax we do not need to go back and change the information presented in the abstract syntax grammar and it is this that is important in language engineering terms since there may be other concrete syntaxes for other languages which rely on the abstract syntax grammar having precisely the form it has.

6 Conclusion

We have suggested a way of preserving the attractive and elegant picture of the GF interlingual architecture by building in elements of transfer into the construction of concrete syntaxes. The operations we have suggested appear to be linguistically quite intuitive corresponding to ways in which we think of the translation relationships that hold between languages. It should be emphasized, however, that at the time of writing this particular TTR implementation has not been tested on anything much larger than what is discussed here. We present this, therefore, as a suggestion for ideas to explore rather than established results. We do, however, think it is suggestive of the potential power of combining functional, type theoretic techniques with unification-based techniques.

References

Bender, E. M., D. Flickinger, F. Fouvry, and M. Siegel (2005). Shared representation in multilingual grammar engineering. *Research on Language and Computation 3*, 131–138.

Cooper, R. (2005a). Austinian truth, attitudes and type theory. *Research on Language and Computation 3*, 333–362.

Cooper, R. (2005b). Records and record types in semantic theory. *Journal of Logic and Computation 15*(2), 99–112.

Cooper, R. (in preparation). Type theory with records and unification-based grammar. Available from `http://www.ling.gu.se/~cooper/records/ttrhpsg.pdf`.

Curry, H. B. (1961). Some logical aspect of grammatical structure. In *Structure of Language and its Mathematical Aspects: Proceedings of the Twelfth Symposium in Applied Mathematics*, pp. 56–68. American Mathematical Society.

Kim, R., M. Dalrymple, R. Kaplan, T. King, H. Masuichi, and T. Ohkuma (2003). Multlingual grammar development via grammar porting. In *Proceedings of the Workshop on Ideas and Strategies for Multilingual Grammar Development, ESSLLI 2003*, pp. 49–56.

Perera, N. and A. Ranta (2007). Dialogue system localization with the dialogue system localization with the GF resource grammar library. In *Proceedings of the ACL Workshop on Grammar-Based Approaches to Spoken Language Processing (SPEECHGRAM)*.

Ranta, A. (2004). Grammatical Framework: A type-theoretical grammar formalism. *Journal of Functional Programming 14*(2), 145–189.

Ranta, A. (2007). Modular grammar engineering in GF. *Research on Language and Computation 5*(2), 133–158.

Rayner, M., B. A. Hockey, and P. Bouillon (2006). *Putting Linguistics into Speech Recognition: The Regulus Grammar Compiler*. CSLI Studies in Computational Linguistics. Stanford: CSLI Publications.

Sag, I. A., T. Wasow, and E. M. Bender (2003). *Syntactic Theory: A Formal Introduction* (2nd ed.). Stanford: CSLI Publications.

Santaholma, M. (2007). Grammar sharing techniques for rule-based multilingual NLP systems. In J. Nivre, H.-J. Kaalep, K. Muischnek, and M. Koit (Eds.), *Proceedings of the 16th Nordic Conference of Computational Linguistics NODALIDA-2007*. University of Tartu, Tartu.

Using the Text Processing Tool Textin to Examine Developmental Aspects of School Texts

Bengt Dahlqvist

Uppsala University
Department of Linguistics
and Philology

Mikael Nordenfors

University of Gothenburg
Department of Swedish
Language

1 Introduction

The purpose with this article is to first make a brief presentation of the functions in the web based text processing tool Textin 1.2, and then to illuminate these functions by putting the program to use within a research project in progress that concerns developmental aspects on texts written by Swedish pupils during school years 5 to 9.[1]

The text will begin with a brief description of Textins' main functions, and then move on to previous research on school texts where computer linguistic methods either *were* used or *could have been* used if the technology had been accessible at the time being. The article then continues with a presentation of the results that Textin delivers, and ends with a discussion on these findings.

2 Textin as a Tool

Computer analysis of language has a long history starting back in the early days of computational linguistics. Tools for processing texts have been developed for linguistic applications per se and also in many application areas where study of language is conducted. The basics of text and word analysis include the computation of different kinds of word lists and related statistics, valuable for the language researcher in his task. Barnbrook (1996) points out some of the main advantages with using computers for language analysis; the ease of which one can manipulate, select, sort and format data, the speed and accuracy and so on. This is of course true even more today, when the ubiquitousness of texts in electronic readable form is becoming reality.

Aside the last decade of massive software development for language engineering purposes, the need for classic, simple analysis tools remain.

[1] The Swedish school system contains years 0 to 12 (although the last three years are optional).

Whereas tools such as the well-known WordSmith system (Scott, 2004) are well established, still a need for quick, web-based tools pertains. This paper presents such a tool, Textin, in part based on an earlier standalone program TSSA (Dahlqvist, 1994). Textin offers some core tools, designed to be fast, for small to medium sized texts for immediate analysis over the Internet by a web interface.

2.1 Implementation

Textin is written in the programming language PHP, which is specially suited for web applications. Textin can be used for making basic word analyses and is able to produce simple word lists of various kinds from texts. The main window for Textin is shown in figure 1. The text to be analysed is pasted into the available text area and the analysis is started by clicking the submit button.

The present version 1.2 of Textin has the following URL:

```
http://www.lingfil.uu.se/personal/bengt/textin/
```

The program is protected by a login password, which can be given to users upon request.

2.2 Language Options

The languages supported by Textin are for the moment Swedish and English. All text in the interface is given in Swedish, and the pages are displayed in Latin-1 (8859-1) encoding. A drop-down list on the main window enables the user to choose between languages.

The choice between Swedish and English affects the sorting of the text constituents. Word lists processed as Swedish text will order words containing the Swedish letters ‘å’,’ä’ and ‘ö’ in the proper manner for Swedish, as well as jointly order words with ‘v’ and ‘w’, in contrast to the English sorting order.

Figure 1: The main interface for Textin, showing a school text to be analyzed.

2.3 Word Lists and N-grams

Textin allows for the creation of several types of word lists:

- Alphabetically sorted
- Final-alphabetically sorted (i.e., reverse order)
- Frequency sorted (descending order)

As an option, given as a check box on the main window, it is possible to filter away entries from the found tokens not containing any letters or numerals, i.e., entries not ordinarily considered constituting a word.

All computed word lists are listed in full, together with sequence and frequency numbers, on the resulting report page (see figure 2). Tokens in Textin are defined as strings separated by whitespace, i.e., blanks, tabs and linefeed, and delimiters such as semicolon, slash, exclamation and question mark as well as double quotes and parentheses. These delimiters together with period, comma, single quote and colon are trimmed away from the strings and cannot alone constitute a token. Words are for efficiency reasons case normalized, so that they contain only lower case letters.

In addition to the various common word lists, Textin also allows for the creation of some types of n-grams:

- Bigrams
- Trigrams
- Tetragrams

For technical reasons, no non-word filtering can be made on the n-grams. Further, the n-grams do not make any assumptions about sentence boundaries, of which Textin has no knowledge.

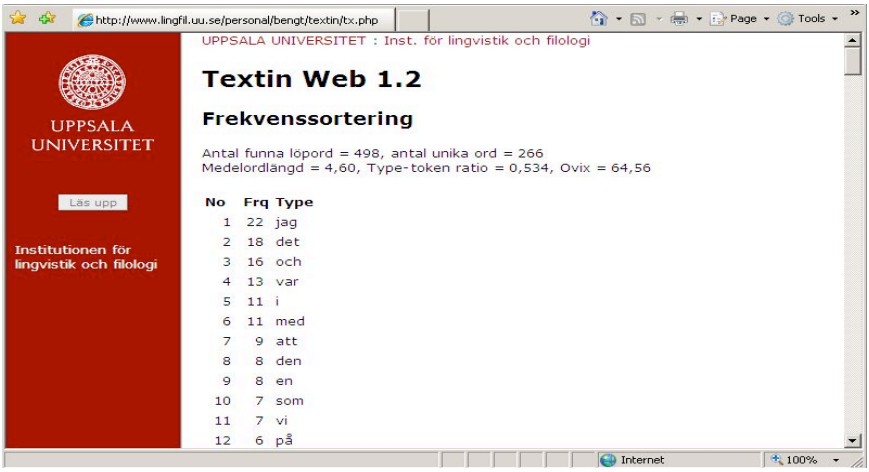

Figure 2: A frequency ordered word list computed from the text in figure 1.

2.4 Statistics

In addition to the produced word lists themselves, Textin also computes and reports some basic and relevant statistics for the text analysed. These statistics include:

- Number of tokens found
- Number of types found
- Mean length of tokens
- Type/token ratio (TTR)
- Word variation index (Ovix)

For n-grams, only the number of total n-grams and the number of unique n-grams in the text are given. Otherwise, for the common word lists, the

number of tokens is the total number of words found in the text, while types stand for the number of unique words. For the word lists, the mean length of the word tokens, not including the delimiters, is given next. The type/token ratio will be a number between zero and one, and hints at the word richness and readability of the text. Ovix (Hultman and Westman, 1977) is an established measure of word variation, which gives information on the level of complexity of the text. Note that Ovix cannot be computed, and is not meaningful, when the number of types and tokens are equal.

2.5 Restrictions

PHP and the server environment impose some restrictions on the text handling capability for Textin. First, a memory limit is set by the server which yields a text size of maximum about 150,000 words. The length and variation in each text to be analyzed determines this number in detail.

The recommendation is to keep the texts below 100,000 words, which corresponds to the word mass for one to two novels of standard length. Secondly, a time limit is set. The server utilized allows no execution time to exceed 60 seconds. This should however not be a grave problem, since the space limit should make itself known well before the time limit is reached.

3 Previous Research Findings on Writing Development

A quick overview will now follow concerning research that has mainly or partially focused two of the quantitative measures Textin can provide: *average word length* and *OVIX* (see figure 2). Textin offers more functions than merely these two, but when it comes to comparing texts over time, average word length and OVIX are perhaps the more reliable factors. Another measure, which traditionally has been used as a sign of productivity, is total text length. Such a measure is here considered being too sensitive to the different contexts of production and will therefore not be used. And while already addressing problems of such it must be mentioned that the two measures used in the article are far from liberated from issues of context.

3.1 Average Word Length

One point, made by earlier research, is that a texts quality is connected with its average word length. An extended word length can be considered a sign of a more mature language, e.g., the capability of economizing language in order to make it more packed with information. Björnsson (1968) showed that words that hold more than 6 letters could be considered being “long

words", and a higher relative frequency of such words would indicate a more mature vocabulary. In the "LÄSK-project"[2] (Grundin 1975) 2500 pupils, 7 to 19 years old, were tested for reading and writing ability in order to gain knowledge about developmental aspects. The measure average word length was not calculated in this project, but the project clearly shows that the frequency of long words does increase over that particular 12 year period (Grundin 1975, p. 53). This gives at least some initial support to the thought of using word length as a developmental parameter when characterizing writing ability.

Hultman and Westman's *Gymnasistsvenska* (1977) is a large survey project that describes different quantitative aspects of text production, with data from school years 10 to 12 (upper secondary school). One of their conclusions on word length was that it has a positive connection with higher grade (Hultman and Westman 1977, p. 78). The word length ranges from 4.83 to 5.52; all together giving these pupils an overall average of 5.07 letters/word. Nyström (2000) shows that texts[3] written in the same school stage, now in years 1996 and 1997, scored an average word length of 4.7. Nyström also points to the fact that type of text or genre (probably) correlates with average word length. Genres that are mostly built upon displaying facts gets an average of 5.26 – thus higher than in *Gymnasistsvenska* – and in a more informal text type, as a letter, the average is as low as 4.29 (Nyström 2000, p. 179). The difference in genre makes it difficult to speak of any decreasing (or increasing) trend within this stage of the school period, and Nyström declares that larger surveys are needed in order to compare results (Nyström 2000, p. 178).

Neither Hultman and Westman nor Nyström covers the earlier year's average word length, so others will have to complete the longitudinal picture. Pettersson (1977, p. 131) states that in between school years 5 to 9 the average word length increases from 4.1 to 4.4. Furthermore, Olevard (1999 and 2002) shows that the average word length in school year 9 has decreased over the years between test groups. The findings of that study shows an average word length of 4.18 in 1987 and 4.17 in 1996, based on those years national tests.[4] Lindell et. al (1978) and the project FRIS[5] concludes that the average word length in school year 6 is 4,2 and finally, in school year 4 it is 4,1: a figure perhaps close to the end of the measurable scale of development.[6]

---

[2] "Reading and writing competence and its' development through the school years". (Läs och skrivförmågans utveckling genom skolåren.)

[3] Based on Nyströms (2000) entire material, i.e., a material consisting of many different genres.

[4] Olevards' measures are provided by the above mentioned text processing program TSSA (Dahlqvist 1994).

[5] Fri Skrivning i mellanstadiet. "Free writing within intermediate levels".

[6] The figures mentioned are of course on a general level and individuals does of course deviate from these approximates as in Larssons (1984) sample where the lowest (of the listed) average word length turns out to be 3.66.

Putting it altogether, one could say – all reservations taken into account – that between school years 5 and 12 the development increases from 4,1 to 4,9 on a general basis. Then, there is obviously something with word length that corresponds to age, specifically between the writing in year 9 and years 10-12. Let us continue with the second aspect on text development: Ovix.

3.2 Ovix[7]

Another parameter, often referred to as a signal of development, is word variation. This is possible to measure with either a type/token ratio or Ovix ("Word variation index" – Hultman 1977 and 1993). Ovix is a measure that works with a logarithm that enables comparison between texts of different length. This is important because a text of greater length also contains a lot more of short, logical words, like conjunctions, which reduces the type/token ratio, thus making it difficult to use when comparing variation as achievement.

The first project to use Ovix was Hultman and Westman (1977). The total score in the 1977 material of Hultman and Westman showed that essays with the lowest grade held an Ovix of 58,7[8] and those with the highest grades had 67.6. As comparison, brochures [61], newspaper text [67.6] and text books [71] were listed. It was then possible to say that the writers in the upper scale of grades were closing in on, and overlapped some of the adult texts word variation. Hultman and Westman also displays a listing of essays with approximately the same amount of types ranging from an Ovix of 56.7 up to 72.9 – thus a quite striking difference between texts of the same length making it, on an individual level, impossible to say that *only* text length can guarantee text quality within school writing.[9]

Nyström (2000) uses the same measurement for word variation[10] and lists the values in relation to what educational program in upper secondary school the pupil attends. Nyström presents Ovix related to two different text types which are labelled A and B (both written within the national testing system), where A is a text that is rich of quotations – making the Ovix automatically higher when quoting more mature writers – and B that is closer to the pupils' own production and less "importing". Text type A renders an average of 64 and text type B, 60. The highest Ovix is performed on "theoretical" programs of the social and natural sciences [68], and the lowest on the more "practical, profession preparing" programs [48] (Nyström 2000, p. 176).

[7] The formula for Ovix 1 is defined with the help of the natural logarithm: ln.Ovix = 1/(ln(2-(ln(types)/ln(tokens)))/ln(tokens)).

[8] It is not easy to pin down what one point of Ovix means but the general meaning is that a higher value signals a better text.

[9] In Larssons (1984) sample the correlation between text length and grade is from 0.81 to 0.83, and in Hultman and Westmans 0.57. On an average basis then, longer texts gets higher grades. The correlation between word variation and grade has been showed to be: 0.61 (Hultman and Westman, 1977:56).

[10] Also extracted through TSSA (Dahlqvist 1994).

Another interesting result from Nyström, is the focus on Ovix within different genres. Ovix, in those texts that holds a large amount of facts and quotations, like for instance the traditional school report on a specific subject matter gets 66 on an average basis. Narratives [58] and letters [52] render lower values, thus the same phenomenon as with average word length.

Turning to the pupils in earlier school stages, Pettersson (1977), shows a rather depressing curve stating that Ovix stagnates between school year 6 and 9 at the level of 55. This is supported by Sjödoff (1989) that notes a minor increase in Ovix between school years 7 to 9 – from 47.6 to 51.1 but the final figure for school year 9 is even lower than it was in Petterssons sample.[11] There is perhaps not enough support from earlier research to display the word variation on a longitudinal basis, but we can at least state that it increases from year 4 with an average Ovix of 51 to 63.2 in school year 12.

4 Putting Textin to Work

4.1 Concerning Method

As mentioned above, programs like *TSSA* and the lightweight web version *Textin*, has successfully been used in order to produce quantitative measures on text quality. Some measures have proven to have strong connections with the teachers' assessment. It has therefore been argued that the two measures can be considered a) two signals of what defines longitudinal development and b) signals that in a general way can differentiate achievement within a small or large sample of pupils. These two arguments constitute a strong reason for redoing research in a new, longitudinal project. Another motivation for doing the calculations is to establish bonds between research projects over time on such a complex subject matter as written texts. It is thus necessary to reuse some of the methods of analyses used in the preceding field of research in order to establish a context to the more recent text material.

4.2 The Project[12] and Its Empirical Data

The project has a corpus of texts that originates from 31 pupils attending school years 5 to 9. The text types are of 12 different kinds, all together 331 texts containing approximately 500,000 words. It is quite a large material but this is where Textin has its' advantages. The program is both easily

[11] There is also a third result, concerning Ovix on the period school year 6-9, that states an increase from 55 to 60 (Josephson and Melin 1990:45).

[12] The project's working name is Developmental Aspects on Texts Written in School Years 5 to 9 Within the Subject Matter of Swedish. From now on, in this article; let us just call it *The Project*.

accessible through the web and it is also a program that works very fast with text masses of this particular size. The material as a whole can not be run through Textin due to the restrictions mentioned above, but the text types taken separately contains about 70,000 words which is processed in approximately 3 seconds with Textin. In the following sections the results on the texts in *The Project* will be presented and when possible related to earlier research.

In order to make the results more comparable, the texts from *The Project* will be divided into "narratives" and "other texts". The results on the national tests will also be compared to 30 texts from the online corpus *Skrivbanken*.[13] This gives us the (unique?) possibility to compare measures on national tests performed in school year 9 for the years 1987 and 1996 (Olevard 1999); 1992 and 2003 (Skrivbanken) and 2007 (*The Projects* 31 tests). We will also display the national model texts' (texts put forward by the national tests group, to aid teachers in their grading)[14] measures in order to visualize one demand on text quality on a national level. It is important to notice that the national test is constructed as a multiple choice of tasks, but a majority of the pupils gets pulled down by the so to speak "Bermuda triangle of narratives". The texts from *Skrivbanken* are all of a narrative kind.

Within the group of narratives we will also focus upon differences between the national tests' text products, and the texts written in a scaffolded writing environment. It is of course a hypothesis that the results on the latter texts become different, due to the helpful surroundings. We will return to this issue to see if any results could support such a hypothesis, and what the complications for pedagogy and testing system in such case might be. The texts mentioned as "other" will, in the present article, mainly serve as means of comparison to the narratives.

4.3 The Project's Results

4.3.1 Average Word Length in National Tests in School Year 9, Period 1987–2007

Is there something to learn from drawing a longitudinal line through the 20 years of national test results that are accessible to us? The test system has during this period been more or less thoroughly reformed, and the somewhat elusive variable context is always around to remind us that it affects the writing in different ways. Those reservations set aside we can sketch a curve in figure 3.

[13] www.skrivbanken.se. Acknowledgements to Växjö universitet, Daniel Bergman and Jan Einarsson, for making this material available!
[14] For the year 2007 at this adress:
http://www.nordiska.uu.se/natprov/skolar9/exempel9

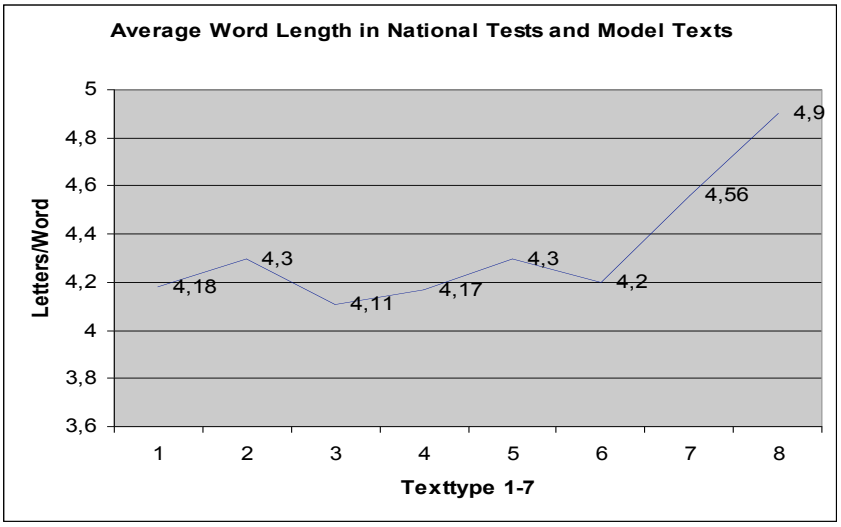

Figure 3: Average word length in national tests school year 9 and national model texts. Texttypes are: 1 – 1986 (Olevard 1999); 2 – 1992 (Skrivbanken.se); 3 – model texts 1992 (Skrivbanken.se); 4 – 1996 (Olevard 1999); 5 – 2003 (Skrivbanken.se); 6 – 2007 – (The Project's texts); 7 – Modeltext 2007 (see footnote no., 14) and finally 8, for comparison between different text types, the scientific essay from 2007.

Two main conclusions can be drawn from this curve. First it is possible to say that the capability measured up with average word length is revolving quite stable somewhere around 4.2. There should then not be any risk that the writing competency has decreased. Second, there are two values that are a lot higher than the rest; the value for the national model text type (2007) and the scientific essay.

We have already commented upon the fact that a text type like the essay, which heavily relies on an externally imported lexicon and/or quotations, must gain a higher average word length. It goes without saying that a text using the specifying lexicon of e.g., aerodynamics automatically increases in both maturity and complexity. What about the value on the national model text then? The three model texts used in this article are the ones that are accessible online, texts within task C4 which is much more less of a narrative than the text type that most pupils chose, task C3. On a national level only 4% chose the text type used as example for the model texts here, but a total of 66% chose a more pure narrative task (Skolverket 2008), and so did the *Text Projects* pupils. Thus, it is not at all strange that our model text of 2007 gains such a different score; again an evidence of differences between text types when it comes to measurements of this kind.

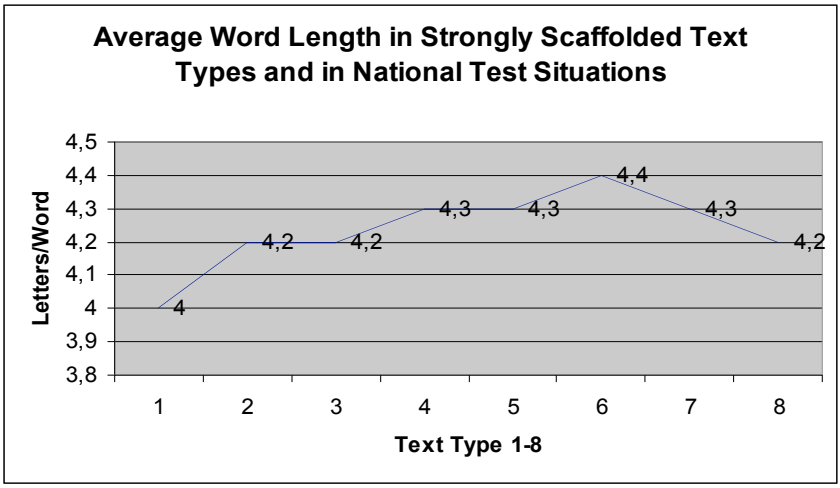

Figure 4: Average word length in different narrative text types within The Project. Text types are: 1 – National test school year 5; 2 – Fairy Tale school year 6; 3 – Teen age short story, year 7; 4 – Criminal short story, year 8; 5 – Robinsonade, year 9; 6 – Horror short story, year 9; 7 – rewriting the national test of school year 5, year 9 and finally, 8 – National test school year 9.

4.3.2 The Projects Narratives Versus the National Tests

Let us now make a comparison between the texts written in the scaffolded text productions in *The Project*, and the ones written within the national tests. We will start by displaying the differences in figure 4. We will also provide the value for average word length in the national test of school year 5.

This longitudinal curve, based on the same individuals' achievements over 5 years of school writing, shows a positive progression all the way until texts 7 and 8. The trend after that seems to be a downward bound curve, making the national test equal to the writing of the year 6 fairy tale. The texts 7–8 are texts written within situations where writing should be performed by hand during time pressure. In the other text situations, the pupils' text is developed during a long period of time with scaffolds like peer response and word processing present. It is not surprising that there is a difference; it would in fact be a lot more surprising if it was the other way around!

4.3.3 Ovix Within National Tests, 1992 to 2007

When it comes to Ovix there are no data accessible for some of the years listed under average word length. For instance, Olevard (1999), does not provide Ovix measures at all, which gives a curve that is some what weaker substantiated (figure 5).

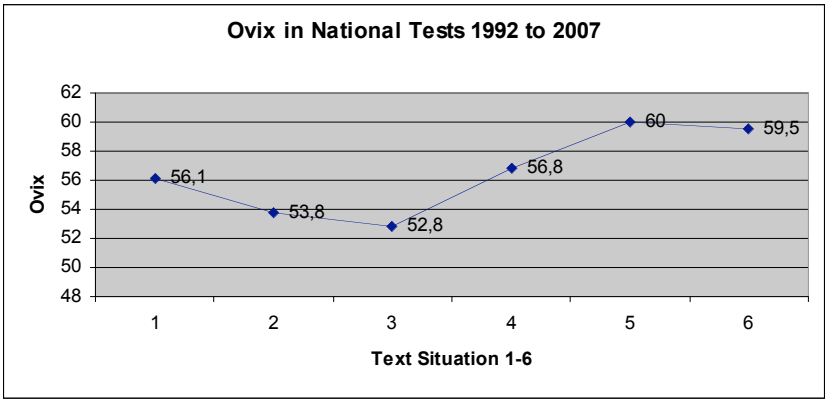

Figure 5: Ovix in national tests and national model texts. Texttypes are: 1- 1992 (Skrivbanken.se); 2 – 2003 (Skrivbanken.se); 3 – model texts 1992 (Skrivbanken.se); 4 – 2007 (The Project's texts); 5 – Modeltext 2007 (see footnote no., 14) and finally 6, again for comparison between different text types, the scientific essay from 2007.

The results are essentially similar to the ones accounted for under average word length. The national tests Ovix is approximately the same at the years 1992 and 2007. The year 2003 is 3 points lower and this would have to be looked into in a more extended way. Again, the model texts of 2007 are at a completely different level compared with the texts of the previous years. Not even the scientific essay, with its' importing language style, can outscore the chosen model text.

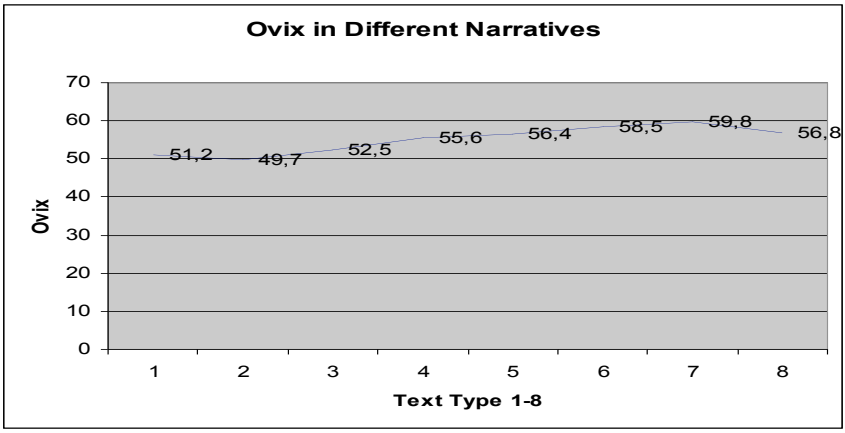

Figure 6: Ovix in Different Narrative Text Types Within The Project. Text Types are: 1 – National test school year 5; 2 – Fairy Tale school year 6; 3 – Teen age short story, year 7; 4 – Criminal short story, year 8; 5 – Robinsonade, year 9; 6 – Horror short story, year 9; 7 – rewriting the national test of school year 5, year 9 and finally, 8 – National test school year 9.

4.3.4 *The Project's* Narratives, versus the National Tests' Ovix

Figure 6 shows the tendency within narratives of different kinds over the 5 years. The curve is quite similar to the one displayed under average word length. There is a stable increase in Ovix from the narrative text type Fairy Tale to Horror Story, and then there is a decrease on the national test. It is interesting to note that text 7, rewriting the national test from year 5, receives such an increase in Ovix but a decrease in average word length. This is merely one of the "asymmetries" that has to be more consequently looked into by a more qualitative analyses on *The Project's* material. Finally, in this curve, we can *not* see that the national test is equal to the writing of the year 6 fairy tale.

5 Discussion

The results based on the material available in this article point to mainly three facts; first, and this is of course a "metafact" aiding the latter two, Textin turns out to be a effective tool when it comes to processing texts of the size of *The Project's*. Second; the results on national writing tests maintain a rather stable level of achievement and third; the measures displayed here shows that pupils, during school years 5 to 9, *do* develop towards a more mature language, although more visible in the scaffolded text types.

Furthermore, the article points out a sort of inconsequence between the text processes surrounding the national tests, and the ones dominating the class rooms visited within *The Project*. This gives an opportunity to raise the question of what exactly the purposes could be with these two different processes. We can either put it this way: The texts produced within the strongly scaffolded writing processes are not a fair and equal material to use for individual grading of pupils. One could say that the texts are more of a social and collective product, than an individual and stand alone achievement of one writer. Or, we could put it this way: The national tests are said to be guide lines to teachers who are going to grade pupils in order to make it possible to make a differentiation before sending them further up in the educational system. Then why is the national tests' writing process conducted in such a manner that it renders poorer results (seen to these measures)? Why is it conducted in such a different and with the writing process' ideal colliding way? As the tests are *not* said to have the absolute and statutory function to regulate the pupils final grades, their purpose is somewhat unclear and this will definitely lead to that they are used in different ways across the nation. And this is of course something that, without doubt, causes inequalities and variation between different classrooms when it comes to what the individual grades really are based on.

It seems as if the national level of testing and evaluating writing skills, and the locally chosen method for teaching and learning writing, is on a collision course which must cause confusion to the grading teachers involved, at least to those with less experience?

6 Conclusion

The main conclusions of this article are that Textin works very effectively thanks to its accessibility via the Web and when working with text masses of *The Project's*size. It has also proven to be able to build bridges between research on language development within school contexts of different kinds over different periods of time. The results from Textin have of course to be viewed alongside with qualitative approaches to textual analysis to help shed light on problems like: What words do the variation consist of? What makes a text of one type so much stronger than another? How come the connection to grade is so strong? What factors dwells within the teachers' habitus and what lies embedded in the school environments' hidden norms of value?

One final reflection will have to cover all other things concerning writing and texts; there is of course an abundance of more matters to tend to when it comes to text quality than merely numbers, but these numbers can serve as a generator of hypothesises to be investigated by qualitative approaches, and vice versa. There is nothing to do but to agree with Nyström's (2000) conclusion that more large scale surveys are needed and on a regular basis to further investigate and evaluate the quantitative aspects on achievement within writing development.

References

Barnbrook, G. (1996). *Language and Computer. A Practical Introduction to the Computer Analysis of Language*. Edinburgh, UK: Edinburgh University Press.

Björnsson C. H. (1968). *Läsbarhet*. Stockholm.

Dahlqvist, B. (1994). *TSSA 2.0, A PC Program for Text Segmentation and Sorting*. Uppsala: Institutionen för lingvistik, Uppsala universitet.

Grundin, H. (1975). *Läs- och skrivförmågans utveckling genom skolåren*. Utbildningsforskning rapport 20. Stockholm Liber/Skolöverstyrelsen.

Hultman, T. G. and M. Westman (1977). *Gymnasistsvenska*. Lund, Liber.

Hultman, T. G. (1994). Hur gick det med OVIX? I *Språkbruk, grammatik och språkförändring. En festskrift till Ulf Teleman. 13.1.1994*. Lund. S. 55-64.

Josephson, O. and L. Melin (1990). *Elevtext. Analyser av skoluppsatser från åk 1 till åk 9*. Lund, studentlitteratur.

Larsson, K. (1984). *Skrivförmåga. Studier i svenskt elevspråk*. Uppsala, Liber.

Lindell, E., B. Lundquist, A. Martinsson, B. Nordlund B., and I. Petterson (1978). *Om fri skrivning i skolan.* Utbildningsforskning FoU rapport 32. Stockholm Liber/Skolöverstyrelsen.

Nyström, C. (2000). G*ymnasisters skrivande. En studie av genre, textstruktur och sammanhang.* Skrifter utgivna av Institutionen för nordiska språk vid Uppsala universitet, 51. Uppsala, Textgruppen i Uppsala AB.

Olevard, H. (1999). *Tonårsliv*. En pilotstudie av 60 elevtexter från standardproven för skolor (sic) 9 åren 1987 och 1996. I *FUMS rapport nr 194*. Uppsala FUMS.

Olevard, H. (2002). Skrivuppgift i repris – presentation av ett forskningsprojekt. I Garme (red. 2002). *Språk på väg. Om elevers språk och skolans möjligheter*. Uppsala, Hallgren & Fallgren studieförlag AB.

Pettersson, Å. (1977). Barnspråk. I Pettersson/Badersten (1977) *Språk i utveckling*. Lund, Liber läromedel.

Scott, M. (2004). *WordSmith Tools version 4*. Oxford University Press, Oxford, England.

Sjödoff, I. (1989). *Med svenska som mål. Effekter av två undervisningsprogram på invandrarelevers svenska i skrift*. Skrifter utgivna av institutionen för nordiska språk vid Uppsala universitet 23. Uppsala, institutionen för nordiska språk.

Skolverket (2008). *Ämnesprovet 2007 i grundskolans årskurs 9. En resultatredovisning*.

Texts chosen from Skrivbanken.se

For the year 1992: 102151,102157, 102202, 112105, 112148, 102160, 115030, 115155, 150040, 150120, 162036, 186005, 302103 and 325101.

For the year 2003: 113309, 207111, 207115, 258315, 281121, 101213, 101214, 101309, 114116, 121115, 114224, 195325, 413202, 875115 and 881219.

Good Tag Hunting: Tagability of Granska Tags

Eva Forsbom

Uppsala University
Department of Linguistics and Philology

1 Introduction

Part-of-speech (or rather morphosyntactic) tagging is an important preprocessing step in most natural language processing applications, even in applications where full parsing is performed, such as rule-based machine translation.

For Swedish, the Stockholm-Umeå Corpus (SUC, Ejerhed et al., 2006) has become the *de facto* standard for training and evaluating part-of-speech taggers for Swedish. It is a balanced corpus, intended to be representative of general, written, published, Swedish. Two interchangeable tagsets are used: the SUC tagset (Ejerhed et al., 1992), and the PAROLE tagset. In this paper, only the SUC tagset is used, but as the two tagsets are interchangeable, any findings will apply also to the PAROLE tagset.

Although the SUC tagset covers most of the morphosyntactic features used in Swedish, it does not cover everything. In order to approve the situation, the SUC tagset was altered to fit the needs of the Granska grammar checker (Carlberger and Kann, 1999). In the Granska tagset, some features were added to the morphosyntactic features represented in the SUC tagset, and some low-frequent features were removed. When taggers using the same learning algorithm with the SUC and the Granska tagset, respectively, were trained and evaluated on SUC, the one with the Granska tagset performed better (Carlberger and Kann, 1999), and I have noticed the same effect in my own experiments. However, no analysis has been done on the effect of the individual changes to the tagset.

Based on my knowledge of Swedish, I suspected that some of changes were more influential than others, such as the differentiation of main and auxiliary verbs, and the conflation of some participle tags with the corresponding adjective tags. I also suspected that the removal of some features was actually harmful, such as the abbreviation feature. Furthermore, I doubted the usefulness of the addition of two semantic features: date (e.g., *januari* 'January') and set (e.g., *meter* in *10 meter*), as they are not morphosyntactic in nature.

In this paper, I have analysed the effect of groups of related changes in the Granska tagset, in the hope of hunting down a set of good tags that could improve the SUC tagset.

2 Background

2.1 SUC Tagset

SUC is a balanced corpus of modern Swedish prose covering approximately 1 million word tokens, with manually validated part-of-speech annotation. The manual validation, and the inclusion of several genres, makes it an excellent choice for tagger training and evaluation. The texts are from the years 1990 to 1994, and the basic idea of the compilation was that it should mirror what a Swedish person might read in the early nineties. It has been published in two versions. The second version contains the same text samples, but has corrected (and additional) annotations. Carlberger and Kann (1999) used version 1.0. For this paper, version 2.0 is used.

Of the two tagsets present in SUC, only the SUC tagset is used here. It has 22 main categories, such as noun and verb, and a set of morphosyntactic features to make further distinctions within the main categories, such as gender, number, voice, and form. A tag for a neuter, singular, indefinite, nominative, common noun, for example, would be `NN NEU SIN IND NOM` (or in Granska format `nn.neu.sin.ind.nom`).

All in all, the SUC tagset consists of 153 different tags[1], many of which occur only a few times (45 occur less than 100 times, and 23 less than 10 times). With such a large tagset, some of the trigram statistics for a trigram tagger will be unreliable (average trigram count is 16 times, and 17% of the trigrams occur only once). This data sparsity was one of the reasons for adapting the SUC tagset.

2.2 Granska Tagset

The Granska tagset is an automatic adaption of the SUC tagset to better suit the purposes of a grammar checker. Adaptation was done in two ways. Firstly, some of the least occurring tags were matched with similar tags and two or more tags were merged into one tag. In this way, the number of tags were reduced, and the most unreliable tags, from a learning perspective, were removed. Secondly, a number of new features were introduced, where the original tagset was not detailed enough, such as adding a distinction between main verbs, and auxiliaries, copulas and modal verbs, respectively, and a distinction between singular and plural cardinals. All in all, 14 tags were removed and 5 new tags added (Carlberger and Kann, 1999).

[1] 140 in version 1.0, according to Carlberger and Kann (1999)

Change group	Example
Copulas (kop)	vara, (för)bliva, heta, kallas
Past participles (pc)	pc.prf.utr/neu.plu.ind/def.nom → jj.pos.utr/neu.plu.ind/def.nom
Auxiliaries (aux)	ha
Modals (mod)	lära, måste, behöva, tänka, försöka, förefalla, söka, ska, böra, töras, kunna, börja, våga, bruka, sluta, låta, få, hinna, verka
Dates (dat)	måndag(en), tisdag(en), januari, februari
Singulars (sin)	1, i
Adjectives (jj)	jj.pos.utr/neu.plu.ind/def.gen → jj.gen
Genitive participles (pc.gen)	pc.prs.utr/neu.sin/plu.ind/def.gen → pc.gen, pc.prf.utr/neu.sin.def.gen → pc.gen
Masculines (mas)	dt.mas.sin.def → dt.mas.sin.ind/def, dt.mas.sin.ind → dt.mas.sin.ind/def
S-forms/Active forms (sfo)	vb.imp.sfo → vb.imp, vb.imp.akt → vb.imp
Sets (set)	kilo, meter, familj, armada
Abbreviations (an)	nn.an → nn, ps.an → ps.utr/neu.sin/plu.def

Table 1: Tagset groups.

No attempt was made to evaluate the effect of each individual adaptation, although the full adaptation was evaluated, and found to give slightly better performance results than the original tagset (2%).

The conversion of SUC tags to Granska tags was done automatically by a lexer. I did not have access to the original version of the lexer used in Carlberger and Kann (1999), but had a newer version, and a version of SUC with Granska tags converted with the original version. The newer lexer was ported to Perl, and harmonised with the Granska-versioned SUC. The lexer rules were divided into groups of similar changes, which could be turned off or on. An overview is given in table 1.

Three groups concern differentiation between main verbs and auxiliary verbs: copulas, (temporal) auxiliaries, and modal verbs, each with an extra feature added (kop, aux, mod). This differentiation was actually present in an earlier version of the SUC tagset (Ejerhed et al., 1992), but is removed in the current version. As the Granska conversion lexer assigns the additional features automatically, no distinction is made between, for example, the copula, passive-constructing, or existential reading of *vara* 'be', or the temporal auxiliary or main verb *ha* 'have'. Although most occurrences in SUC of those verbs could be classified as auxiliaries, some could not, thus introducing some noise in the corpus.

One group takes care of the removal of the voice feature (s-forms, sfo, and active forms, akt) of imperatives and conjunctives. S-forms are either passive or deponential forms of a verb, as in *hoppas* 'jumping' or *hoppas* 'hoping'.

Two groups concern past participles. The first group conflates past participles and the corresponding adjective (positive) form to the same tag, e.g., *handikappade* 'handicappeds' with tag `pc.prf.utr/neu.plu.ind/def.nom` would get the tag `jj.pos.utr/neu.plu.ind/def.nom`. The other group conflates all genitive forms of participles to one single tag, so *handikappades* 'handicappeds" with tag `pc.prs.utr/neu.sin/plu.ind/def.gen` would get the tag `pc.gen`.

Another group deals with the conflation of adjective tags. Genitive forms are merged into one tag, as with past participles. Three rules also conflates the infrequent tags of three word forms into a more frequent tag: *rätt* 'right', *flest* 'most', and the compound part *äldre-* 'elderly'.

Yet another group conflates tags for determiners with masculine forms into a common tag. The rules concern three word forms: *denne* 'this', *själve* 'himself', and *samme* 'the same'.

One group removes the abbreviation feature from tags. For example, the tag for the abbreviated form *proc* 'per cent' would change from `nn.an` to `nn`. There are also a couple of rules that adds features to the shortened tag to conflate it with an existent tag. The rules include the determiner *d* 'the', some adjectives (*f_d* 'former', *St* 'Saint'), two past participles (*tf* 'acting', *adj* 'adjoint'), and the possessive pronoun *h:s* 'her'.

The last morphosyntactic group adds the number feature value `sin` to singular cardinals or ordinals, i.e., *1* and *i*.

Finally, there are two semantic groups. The first deals with date expressions, and adds the feature `dat` to words representing months and days. The second with set expressions, and adds the feature `set` to words representing measurements (e.g., *kilo* and *meter*), indefinite numbers (e.g., *femtiotal* 'some fifty' and, more questionable, *femtiotalet* 'the 50's'), and group words (e.g., *familj* 'family' and *armada*).

3 Experiments

In order to try the goodness of each group of adaptations when it comes to tagging performance, I used the statistical TnT tagger (Brants, 2000) to train a number of models. The models were evaluated using 10-fold cross-validation[2] on SUC, with and without the current adaptation under scrutiny. Both training and testing was done with the tagger's default settings[3], i.e., no parameter optimisation.

For feature optimisation, I use a forward selection strategy. First, each group of adaptations is tried in isolation to see whether they are significant

[2]The cross-validation set is available at `http://stp.lingfil.uu.se/~evafo/software/cross_validation_sets.txt`.

[3]During training the -c flag, for internal case-labelling of tags, was used, as this generally gives better performance for Swedish.

or not. Then, all groups showing a significant change in performance are tried in pairs to see whether the two groups affect each other, and finally, the top-3 most promising groups are combined with groups with significant combination power.

The performance measure is accuracy, i.e., the percentage of correctly tagged words. It is measured with the `tnt-diff` tool, which gives overall statistics as well as separate statistics for known and unknown words[4]. As most of the adaptations aim at improving the performance for unknown words, this information is interesting.

Statistical significance is tested with McNemar's chi-squared test (R Development Core Team, 2007), but since the sample size is very large, even small changes will be significant, however useful they are in practice. Therefore, the effect size is also estimated with an exact binomial test (R Development Core Team, 2007).[5] The effect size is the probability of success of the tested adaptation in cases where the two taggers differ.

3.1 Single Groups

The performance for single groups is shown in table 2. As can be seen, the copula and past participle groups contribute most to the improved result. These groups explain most of the boost in performance for known words. Although the size of the tagset increases for copulas, the extra tags makes it easier to distinguish the rather stable, copula-specific, trigrams from other trigrams including other verbs.

The set and abbreviation groups slightly harm the performance, but the change is not significant. All other groups improve the accuracy, although the change for the voice group is not significant, and for the masculine and modal groups is only mildly significant. The improvement is particularly visible for unknown words.

For the modal group, the probability of success is almost fifty-fifty, which makes it a rather unsafe (automatic) adaptation. A probable explanation is that among the main verb–auxiliary distinction groups, this is the group which contains most borderline cases.

3.2 Pairwise Groups

The performance for pairwise groups is shown in table 3. The copula and past participle group pair did answer for almost all the improvement. No pairs harmed the performance, but two groups did not improve the performance in any pairing: masculines and genitive participles (except for `pc.gen + aux`), which makes these adaptations less useful for tagging purposes.

[4]The proportion of unknown words is 7.87 ± 0.20 for all models.

[5]Two-sided, at 95% confidence level, with a null hypothesis of 0.5% success.

Group	Tagset size	Accuracy (%)			Prob. of success
		Overall	Known	Unknown	
No change	153	95.52±0.15	96.31±0.13	82.26±0.99	
All***	152	95.68±0.14	96.42±0.13	87.10±0.91	61 (60-62)
kop***	160	95.60±0.14	96.39±0.13	86.36±0.96	63 (61-64)
pc***	144	95.60±0.14	96.33±0.13	87.04±0.91	65 (63-67)
aux***	157	95.53±0.14	96.32±0.13	86.30±0.99	57 (54-59)
mod*	158	95.52±0.14	96.32±0.13	86.27±0.97	52 (50-54)
dat***	155	95.52±0.14	96.32±0.13	86.26±0.99	66 (61-72)
sin***	155	95.52±0.15	96.31±0.13	86.25±0.99	64 (57-70)
adj***	145	95.52±0.15	96.31±0.13	86.27±1.00	64 (56-72)
pc.gen**	149	95.52±0.15	96.31±0.13	86.27±0.99	66 (54-76)
mas*	152	95.52±0.15	96.31±0.13	86.26±1.00	67 (52-80)
sfo	151	95.52±0.14	96.31±0.13	86.27±0.99	61 (49-72)
set	157	95.52±0.14	96.31±0.13	86.23±0.98	50 (46-53)
an	146	95.52±0.15	96.31±0.13	86.25±1.00	49 (45-53)

Table 2: Performance per group (10-fold cross-validation), sorted in decreasing order of performance (higher precision not shown here). *** means significant at $\alpha = 0.001$, ** at $\alpha = 0.01$, * at $\alpha = 0.05$.

The modal group slightly improves the result in pair with the auxiliary, date, singular, and adjective group, so it seems that in combination with those it is less riskier, and a potentially good adaptation. However, in combination with the copula and past participle group, it slightly harms the performance.

3.3 Top Groups

Both the single-group and pairwise-group experiments showed that the adaptations in the copula and past participle groups are highly motivated, in isolation as well as in combination. The auxiliary group also showed high potential, while the masculine and genitive participle groups had little effect. This leaves four mildly potential groups: modals, singulars, dates, and adjectives, which are added one by one, and so on, to the top-3 combination of copulas, past participles, and auxiliaries. The results are shown in table 4.

It is clear that the singular and date groups somewhat improves the performance. The changes in the singular group has a higher probability of success than those in the date group, but fewer instances, and therefore a lower impact on the performance. One explanation could be that there are more fixed phrases with dates than with singulars, and that by adding the date feature, the model could distinguish those phrases from other noun phrases more easily. If, for some reason, a pure morphosyntactic tagset is preferred, the date feature can be used during tagging, and then removed, as no other information is lost in the process.

The modal group still has a negative effect on the performance, although the change is not significant, while the adjective group has a small, non-significant,

	kop	pc	aux	mod	dat	sin	jj	pc.gen	mas
kop	95.60 63%	95.68	95.61	95.60	95.61	95.61	95.61	95.60	95.60
pc	65%	95.60 65%	95.61	95.60	95.60	95.60	95.60	95.60	95.60
aux	61%	63%	95.53 57%	95.54	95.54	95.54	95.54	95.54	95.53
mod	60%	60%	54%	95.52 52%	95.53	95.53	95.53**	95.52*	95.52*
dat	63%	65%	58%	54%	95.52 66%	95.53	95.53	95.52	95.52
sin	63%	65%	58%	53%	63%	95.52 64%	95.53	95.52	95.52
jj	63%	65%	57%	53%	66%	65%	95.52 64%	95.52	95.52
pc.gen	63%	65%	57%	52%	66%	65%	64%	95.52 66%	95.52*
mas	63%	65%	57%	52%	66%	65%	66%	65%	95.52 67%

Table 3: Combinations of groups. Accuracy is reported in the upper right triangle (σ is 0.14 or 0.15 for all combinations), while probability of success is reported in the lower left triangle. ** means significant at $\alpha = 0.01$, * at $\alpha = 0.05$, the rest at $\alpha = 0.001$.

positive effect. There are simply too few change instances in the adjective group to tell if the changes are good or bad.

The best and safest combination, then, would be one with the copula, past participle, auxiliary, date, and singular groups.

4 Conclusion

In conclusion, it seems as if my initial suspicions about the tagset changes were partially correct. The added distinction between main and auxiliary verbs was beneficial for copulas and temporal auxiliaries, but maybe not for modal verbs, at least not without manual inspection. I also believe that the results for copulas and temporal auxiliaries could be improved if the noise introduced was removed.

The conflation of past participle tags with the corresponding tags for adjectives also proved to be a good change, for tagging purposes. If information on past participles is needed, for example, for parsing, the information could not trivially be retrieved in a later stage.

Other changes gave a more subtle, but significant, boost to performance, such as adding the number feature singular to singular cardinals and ordinals, and the semantic feature date to names of days and months.

Combination	Accuracy	Prob. of success
kop + pc + aux	95.68±0.14	64 (62-65)
+ mod	95.68±0.14	49 (47-52)
+ sin***	95.68±0.14	64 (57-71)
+ dat***	95.68±0.14	61 (55-68)
+ jj	95.68±0.14	58 (47-69)
+ mod + sin	95.68±0.14	51 (49-53)
+ mod + dat	95.68±0.14	51 (49-53)
+ mod + jj	95.68±0.14	50 (48-52)
+ sin + dat***	95.69±0.14	62 (57-67)
+ sin + jj***	95.68±0.14	65 (58-71)
+ dat + jj***	95.69±0.14	63 (57-69)
+ mod + sin + dat	95.68±0.14	51 (49-54)
+ mod + sin + jj	95.68±0.14	51 (49-53)
+ mod + dat + jj	95.68±0.14	51 (49-53)
+ sin + dat + jj***	95.69±0.14	62 (57-67)
+ mod + sin + dat + jj	95.69±0.14	52 (50-54)

Table 4: Combinations with the top-3 combination, evaluated against the top-3 combination. *** means significant at $\alpha = 0.001$, the other changes are not significant.

Some changes had very few actually changed instances in the corpus, so they are still dark horses. Two such related changes are the conflation of several low-frequency genitive participle and adjective tags into two single tags (`pc.gen` and `jj.gen`). I suspect that a conflation with a corresponding noun tag would give more generalisation power, as the words are used as nouns. The change, however, is not trivial, as the participles and adjectives are more underspecified than the occurring noun tags.

Potentially harmful, although the change in performance was not significant, were the removal of the form feature abbreviation, and the addition of the semantic feature set to words representing groups of something.

To add up, the best and safest combination of changes to the SUC tagset, given the results of the experiments, would be one with changes for copulas, past participles, auxiliaries, dates, and singulars.

Acknowledgements

Thanks to Johan Hall, Jens Nilsson, Joakim Nivre (Växjö University) for the SUC version with Granska tags, and to Jonas Sjöbergh (KTH) for the Granska tag lexer.

References

Brants, T. (2000). TnT - a statistical part-of-speech tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP-2000)*, Seattle, Washington.

Carlberger, J. and V. Kann (1999). Implementing an efficient part-of-speech tagger. *Software Practice and Experience 29*(9), 815–832.

Ejerhed, E., G. Källgren, and B. Brodda (2006). Stockholm-Umeå corpus version 2.0, SUC 2.0. Stockholm University, Department of Linguistics and Umeå University, Department of Linguistics.

Ejerhed, E., G. Källgren, O. Wennstedt, and M. Åström (1992). The linguistic annotation system of the Stockholm–Umeå corpus project. Report DGL-UUM-R-33, Department of General Linguistics, University of Umeå.

R Development Core Team (2007). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

How to Build an Open Source Morphological Parser Now

Kimmo Koskenniemi

University of Helsinki
Department of General Linguistics

1 Introduction

This article compares the building of a morphological parser or analyzer some 20–25 years ago to the same task now. Much has changed but some tasks remain quite similar even if computers have now perhaps 10,000 times larger memories and disks than then, and their processing speed has become faster by a comparable ratio.

The building of a morphological parser has become less laborious and faster, but not by a similar factor. The speed and capacity of the computers is not the key factor in this case. The bottleneck is still the manual work and linguistic skill that is required, But, the availability of useful language resources and the emergence and the spreading of new techniques and methods makes a great difference when building a morphological parser.

A morphological parser is a central component in many larger applications and systems of language technology. These often have different requirements, and therefore the researchers need to test, modify and improve the parser in various ways. Modification and flexible use of commercial products is not easy, and therefore the open source aspect in producing a morphological parser has become relevant and is also discussed.

2 How We Did It in the Past

Morphological parsers were not considered particularly interesting in the 1960s and 1970s. We usually explain this by noting that English morphology is so simple that the task is too trivial to deserve much attention. Other languages were studied to a much lesser degree, and furthermore, every language appeared to have a unique solution. Among the early pioneers were Staffan Hellberg who lemmatized the Swedish frequency dictionary (1971), Anna Sågvall who designed an inflectional analysis of Russian (1973), Jean Meunier, Jean Boisvert and Francois Denis who did it for French (1976), and M. L. Hann for German (1974, 1975).

Serious interest in the general aspects emerged soon after a wider array of languages was treated, most notably Martin Kay's early work (1977), and subsequent investigations of Anna Sågvall Hein on the applicability of Kay's general parsing methods to Finnish inflection (1978, 1980). These were soon followed by experiments by Benny Brodda, Gunnel Källgren and Fred Karlsson in the framework of BETA rewrite rule systems e.g., (Källgren, 1981), and a TEXFIN system of linked minilexicons by Lauri Karttunen and his students (1981).

The basic work of Ron Kaplan and Martin Kay in the 1970s and 1980s was published quite late (Kaplan and Kay, 1994) but it affected the contemporary work in early 1980s. Their work resulted in a framework of finite-state transducers where phonological alternations could be described in bidirectional manner, and gave also rise to the two-level morphology (Koskenniemi, 1983) which was easier to implement on computers of those days.

2.1 Handling the Inflection

Most of us who built morphological parsers in the early days, started with inflectional rules by picking example words. The morphophonological alternations and the choice of allomorphs in their inflectional forms were carefully studied. Everything had to be described in terms which were explicit enough for the computer. Often this required some reorganization of the inflectional classes listed in dictionaries and grammars. In essence, we had to

- establish classes for lexemes where each class was homogeneous enough to allow similar mechanical inflection within them and
- implement an inflectional mechanism for each class, so that any new lexemes entered to that class will be correctly inflected

Actually, most parsing methods of that time were not bidirectional enough to let the linguist design the parser by thinking in terms of generating the word forms and then just using the parser in the reverse direction for analyzing word forms. The testing was, therefore, more time consuming and less reliable than now because tools, such as the Xerox LEXC and XFST readily produce full sets of inflectional paradigms which can be inspected for correctness and completeness (Beesley and Karttunen, 2003).

2.2 Accumulating the Lexicon

Once the example words seemed to function properly, one started to extend the lexicon. One usually had to follow the hard way, and find the words one by one. One had to determine the inflectional characteristics of each lexeme using one's linguistic intuitions. Only few builders of parsers were lucky enough to have a computer readable dictionary at their disposal. In this way, much work could be saved by reusing large vocabularies compiled by others.

Quite often, though, one could use existing frequency dictionaries or word form frequency lists computed from the modest corpuses available in those days. Using such word lists, one could start with the most relevant words and reach a fair coverage for the parser.

After some initial vocabulary was included, one tested the parser against corpus texts and incrementally added words encountered in corpuses which failed to be recognized. In this way, the parser was enhanced and extended while it was used. Many of the remaining bugs in the inflectional endings and rules were detected while the lists of unrecognized word forms were studied. Overgeneration, i.e., cases where the parser would accept ungrammatical combinations of stems and endings would not be easy to detect in this way.

3 What Is Easier Now

In addition to more powerful computers, we now have more data at our use and we have technologies such as finite-state calculus, and machine learning methods, not to mention handy scripting languages such as Python and Perl.

3.1 More Data Available

Nowadays, we have lots of computer readable texts available, maybe 100–1000 times more than 25 years ago. This helps the building of morphological analyzers, of course, because larger corpuses contain instances of a larger set of lexemes. Larger corpuses also have occurrences of lexemes in several inflectional forms which makes it possible to guess the base form and inflectional characteristics of such new lexemes. Evidence from several distinct inflected forms makes hypotheses more likely, and having several occurrences of a word form reduces the probability of it being an accidental typo. We have much more data to be processed, but the improvement of the computers more than compensates this.

Much textual data is available in various language banks where researchers can use them, and in collections of international distribution agencies such as *Evaluations and Language Resources Distribution Agency (ELDA)* or *Linguistic Data Consortium (LDC)* from where (even) commercial users can buy copies of corpuses.

Another new source of text data is the Internet. The web contains enormous amounts of data accessible for the interested users. One can easily test the power of Internet as a source by using a search engine such as Google. Just take a neologism and look for occurrences of its different inflectional forms (assuming that Google searches for exact occurrences of forms without any attempts to normalize the occurrences to their base form).

You can try the power of such web corpus data with some newly invented verb such as a Finnish verb *larpata 'performing live action role-playing (LARP)'*. Such a word could not be present in any older lexicon, as the concept and ac-

tivity is fairly recent. One can go ahead and test whether inflectional forms of such a (hypothesized) verb occur in the Internet. There will actually be plenty of hits for all common verbal forms such as larppasivat, larpataan, larppaan. If you try hypothesized nominal inflections of larpata such as larpatassa you will not get any hits (or at most sporadic misspellings).

As soon as we have a hypothesis of a lexeme and its inflection, we can thus test the hypothesis against the corpuses or the Internet. Competing hypotheses can be ranked and the most likely one chosen.

3.2 Finite-State Calculus

The use of finite-state machines and transducers in language processing apparently dates back to Zellig Harris and the late 1950s according to Joshi and Hopely (1996). C. Douglas Johnson's discovered how phonological rules can be described with finite-state transducers (Johnson, 1972), but it was the work of Ronald Kaplan and Martin Kay that made the finite-state calculus in computational linguistics popular. Now this calculus forms the back bone of Xerox language technology. The finite-state methods have in the course of time gained wide use and are also available also as open source implementations such as OpenFST (see Allauzen et al., 2007, for details), SFST by Helmuth Schmid at the University of Stuttgart (2005) and Vaucanson by Jacques Sakarovitch et al., (see Lombardy et al., 2004).

Finite-state methods were present in the early morphological parsers of the two-level model, but then only in a limited extent. For example, in the original and the SIL PC-KIMMO (Antworth, 1990) implementation, the lexicon is treated as a linked set of letter trees, even if it could as well be represented as a theoretically more streamlined and clean transducer. The early rule transducers were constructed manually (and in the PC-KIMMO program this is still the most common case). With up to date technology, lexicons and rules can, of course, be compiled to transducers and then intersected and composed into one single transducer of reasonable size. This allows for optimization and better speed at run time.

The OpenFST, SFST and Vaucanson software packages are quite robust and efficient basic finite-state calculi and they can be used in full scale language technological projects. What they lack, is the kind of sophisticated user interfaces (such as XFST, LEXC, TWOLC) which Xerox has developed on top of its finite-state library.

3.3 Automatic Discovery of Lexical Information

Machine learning could aim at different levels in retrieving lexical information out of texts. A language independent approach such as what Krister Lindén has followed, might concentrate in finding good guesses for single unknown word-forms based on regularities acquired from training data presenting pairs

of base forms and their inflected forms (Lindén, 2008). On the basis of such a guesser, the linguist might process unknown words from new texts more efficiently with much less manual work than in the traditional scheme.

Another approach is to build a guesser as a variant of a finite-state morphological parser where the actual stems of words have been replaced by wild card expressions allowing any phonologically possible stems, as proposed in Beesley & Karttunen (2003). This type of an analyzer outputs usually several possible base form plus morphosyntactic feature combinations. Thus, it is not optimal as a starting point for collecting new lexical entries if used for isolated word tokens. On the other hand, if we process a large batch of word-forms at a time, their collective evidence helps. If there are several forms of a single lexeme in the data, each of them will lead to the correct candidate as a common one. Each word-form will also give rise to sporadic other candidates which will be less frequent. It can therefore be expected that the base forms which are proposed based on many distinct forms are more likely correct than those proposed by fewer forms. With some tuning, this method can produce large amounts of lexical entries for the less frequent lexemes.

4 Copyrights and Licenses

The use of automatic methods depends more on the availability of data in sufficient quantities. The use of most published materials are controlled by a commercial publisher. In order to get copies of materials or in order to use them, one needs to sign a license or otherwise agree on some restrictions. Earlier parsers are also useful if available, but they too are often controlled by commercial companies.

4.1 Protecting Copyrights

International agreements and European legislation protects the author's copyright to a literary work. These general restrictions as such would not hinder the development of morphological parsers. Problems arise because commercial interests are associated with the published texts. The publisher is usually the party that has acquired some exclusive rights and it also possesses the electronic copy of such works.

The electronic versions of texts and other published materials are particularly valuable for the publishers and sensitive because, in electronic form, the materials can be copied without much effort and with no loss in accuracy. Thus, it is understandable that publishers are often reluctant in releasing their texts at all, even if appropriate licenses and agreements would be signed. A publisher may rightfully think that there is little to win and a lot to lose. If a material leaks to the Internet, there is no way to undo the loss. The institute or researcher causing the damage will most probably be unable to compensate the damage.

Many publishers, however, consider language technological modules useful for their core operations. Therefore many publishers are willing to take controlled risks by licensing their texts for research use, but then they need partners and arrangements which they can rely on. A part of these arrangements consists of licenses which the users must sign and comply with. The problematic point with these licenses is that they often are too strict.

It is quite typical that the publisher only allows academic use and forbids any commercial use. Furthermore, the license often requires that the materials must remain within the premises or computers of the institute. This used to be quite acceptable in the era of mainframe computers, but nowadays, many researchers would prefer to have a copy of the material on their desktop computer or accessible through the net.

4.2 Protecting Products and Markets

Commercial owners of parsers typically have an interest to prevent competition on a market where they have a product. Thus they do not let even academic researchers, let alone other commercial companies, to have the source code of the programs or source files of lexicons of their products. The improvement or adjusting parsers to specialized or more challenging needs is not possible.

Sometimes, the owner of a parser would allow the user to improve the system indirectly by sending feedback. In such cases, the commercial company usually requires the full control over such additions. This may discourage the academic users so that they have no motivation to produce substantial additions or revisions to such products.

4.3 Open Source Aspects

Open source is a way of allowing and encouraging joint development of free software. There are many open source licenses, and many more which refer to another aspect of freeness, i.e., the fact that they do not charge for the program. Here, we stress the freeness in a way the General Public License (GPL) interprets it, i.e., that one is free to modify and use and do almost anything except transform the free program into a non free one.

Open source licenses like GPL do not allow restrictions for their use, e.g., a GPL program may not forbid any type of use, not even commercial use. This requirement contradicts with restrictions for exclusively academic use, which is often stated for corpuses, not to mention digital lexical materials.

5 A Plan and Work in Progress

We have started a project is called “Open source Morphology for Finnish”, or OMorFi which aims to produce a morphological parser which could be freely

modified and developed for research purposes, but also used as part of open source software.

5.1 Initial Lexicon

The Research Centre for Languages in Finland (Kotus) as published a word list under the Lesser General Public License (LGPL) which allows the use of the word list and works derived from it in conjunction with other (even non free programs). The list contains about 100,000 word entries with information on their inflectional and consonant gradation characteristics. OMorFi uses this list as a starting point and aims at producing a parser under the LGPL license.

Mr. Tommi Pirinen is completing his Master's thesis on converting the above word list into the formalism of SFST finite-state package (Schmid, 2005). This thesis work makes the verb, noun and adjective entries operational, but omits adverbs, pronouns etc. in the first stage. The word list being under LGPL and SFST under GPL license, the resulting transducer is clearly still under LGPL license, because a computer program does not create intellectual works and the GPL license does not imply any additional claims on the output of a such a free program.

5.2 Extending the Lexicon

The Language Bank of Finland includes texts totaling to some 100 million word tokens under a license which allows their use both for research purposes and for producing language technological modules which do not violate as such the original copyright of the text. Using this material, one may produce further entries using the above methods. When included in the word list, these new ones will also fall under LGPL.

If all contributors are required to submit their additions under LGPL and tools, such as heuristic programs to extract lexical entries, under GPL or LGPL, then the improving result stays safely under LGPL. On the other hand, it appears to be vise to follow the practices of Mozilla.org and require that the contributors share their copyright with the department or the university. Thus, there is some party which can issue all of the resulting system under other licenses, if that turns out to be needed in future. This may indeed, be necessary when combining a parser with other open source software, if the other license is not directly compatible with GPL or LGPL.

5.3 Further Tools

The SFST is not quite as convenient as the Xerox tools for describing Finnish morphology. The treatment of morphophonological alternations is a bit clumsy as the full compilation of two-level rules is not supported as nicely as in the Xerox TWOLC. The description of the rich sets of endings and their sequences

is also less natural in SFST than in Xerox LEXC which supports the concept of interlinking sublexicons.

Thus, one student, Miikka Silfverberg, is in the process of implementing a two-level compiler on top of the SFST finite-state calculus. He will use the formulas invented by Anssi Yli-Jyrä which simplifies and generalizes the compiling of the rules into transducers. Another student may start a project on producing an open source lexicon compiler on a similar platform. Multiple compatibility between some existing systems such as the SFST, PC-KIMMO, XFST and versions of the original two-level implementation is aimed.

In addition to SFST, there are other open source implementations of the finite-state calculus, most notably OpenFST and Vaucanson. A work is in progress to produce an interface layer, HFST (Helsinki FST), which provides a unified API for these three. Then, the finite-state tools would become more independent from any particular basic finite-state software they are using. One can then select the fastest or otherwise best package for a particular task. Such a setting is expected to boost the tuning and development of the packages by their original developers.

Other uses for the tools are also possible. Using the resulting transducers would be particularly easy in web applications because the code needed for running a transducer is very simple. Any language and several different types of applications can be handled with the same simple code. Such web pages could, of course, demonstrate the current capabilities of the morphological parser and attract more contributors and users as well as implement directly useful information retrieval or other functions.

Yet another possibility is to implement an intelligent entry generator which would be open for the public. Many users of open source software, such as OpenOffice, would probably be motivated to contribute to the enhancement of the lexicon and spend some effort in order to benefit from better spelling checking in future. They could use a service where such cooperative users can enter missing words using a web page which dynamically assists in determining the exact inflectional properties of the words to be added.

5.4 Future

The European Union has started the preparatory phase of an infrastructure called CLARIN, Common Language Resource and Technology Infrastructure. Quite soon, there will bee a need for dozens of morphological parsers to support the multilingual services of CLARIN. It would be very useful for the design and maintenance of CLARIN software, if all these language modules would be software-wise compatible. Finite-state transducers would be almost optimal in this sense, as the common programs only need to run different transducers. There is no need to program language specific features because a suitable transducer for each language (which are data for the programs), can han-

dle those idiosyncracies. Wide ranges of languages, even minority languages can, thus, be served with the same run time program.

References

Allauzen, C., M. Riley, J. Schalkwyk, W. Skut, and M. Mohri (2007). Openfst: A general and efficient weighted finite-state transducer library. In J. Holub and J. Zdárek (Eds.), *CIAA*, Volume 4783 of *Lecture Notes in Computer Science*, pp. 11–23. Springer.

Antworth, E. L. (1990). *PC-KIMMO: a two-level processor for morphological analysis*. Number 16 in Occasional Publications in Academic Computing. Summer Institute of Linguistics.

Beesley, K. R. and L. Karttunen (2003). *Finite State Morphology*. CSLI Publications. CSLI.

Hann, M. L. (1974). Principles of automatic lemmatization. *ITL: Revue of applied linguistics 23*, 1–22.

Hann, M. L. (1975). Towards an algorithmic methodology of lemmatization. *ALLC Bulletin Summer 1975*, 140–150.

Hellberg, S. (1971). En modell för upptradande af böjningsserier i ett frekvenslexicon. Stencil, Språkdata, Göteborg.

Johnson, C. D. (1972). *Formal Aspects of Phonological Description*. Mouton The Hague.

Joshi, A. K. and P. Hopely (1996). A parser from antiquity. *Natural Language Engineering 2*(4), 291–294.

Kaplan, R. M. and M. Kay (1994). Regular models of phonological rule systems. *Computational Linguistics 20*(2).

Karttunen, L., R. Root, and H. Uszkoreit (1981). Texfin: Morphological analysis of finnish by computer. A paper read at 71st annual meeting of the SASS, Albuquerque, New Mexico.

Kay, M. (1977). *Morphological and syntactic analysis*, Volume 5 of *Fundamental studies in computer science*, pp. 131–234. North-Holland.

Koskenniemi, K. (1983). *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Produgtion*. Number 11 in Publications. Department of General Linguistics University of Helsinki.

Källgren, G. (1981). *FINVX — A system for the backwards application of Finnish consonant gradation*. Number 42 in Publication. Papers from the Institute of Linguistics, University of Stockholm.

Lindén, K. (2008). A probabilistic model for guessing base forms of new words by analogy. In *CICling-2008, 9th International Conference on Intelligent Text Processing and Computational Linguistics, February 17 to 23, 2008, Haifa, Israel*.

Lombardy, S., Y. Régis-Gianas, and J. Sakarovitch (2004). Introducing vaucanson. *Theoretical Computer Science 328*, 77–96.

Meunier, J., J. Boisvert, and F. Denis (1976). The lemmatization of contemporary french. In A. Jones and R. Churchhouse (Eds.), *The computer in literary and linguistic studies: Proceedings of the Third International Symposium*, pp. 208–214. The University of Wales Press.

Schmid, H. (2005). A programming language for finite state transducers. In *Proceedings of the 5th International Workshop on Finite State Methods in Natural Language Processing (FSMNLP 2005)*.

Sågvall, A. L. (1973). *A system for automatid inflectional analysis: Implemented for Russian*. Almqvist & Wiksell.

Sågvall Hein, A. L. (1978). Finnish morphological analysis in the reversible grammar system. In *COLING-78, Information Abstracts*.

Sågvall Hein, A. L. (1980). An outline of a computer model of finnish word recognition. *Fenno-Ugrica Suecana 3*, 7–26.

Supporting Research Environment for Less Explored Languages: A Case Study of Swedish and Turkish

Beáta Megyesi
Bengt Dahlqvist
Eva Pettersson
Sofia Gustafson-Capková
Joakim Nivre

Uppsala University
Department of Linguistics and Philology

1 Introduction

Language resources such as corpora consisting of annotated texts and utterances have been shown to be a central component in language studies and natural language processing as they, when carefully collected and compiled, contain authentic language material capturing information about the language. Corpora are shown to be useful in language research allowing empirical studies, as well as for various applications in natural language processing. During the last decade, researchers' attention have been directed to building parallel corpora including texts and their translations as they contain highly valuable linguistic data across languages. Methods have been developed to build parallel corpora by automatic means, and to reuse translational data from such corpora for several applications, such as machine translation, multi-lingual lexicography and cross-lingual domain-specific terminology. Parallel corpora exist for many language pairs, mainly European languages with special focus on Western-Europe.

In the past few years, efforts have been made to annotate parallel texts on different linguistic levels up to syntactic structure to build parallel treebanks. A treebank is a syntactically annotated text collection, where the annotation often follows a syntactic theory, mainly based on constituent and/or dependency structure (Abeillé, 2003). A parallel treebank is a parallel corpus where the sentences in each language are syntactically analyzed, and the sentences and words are aligned.

The primary goal of our work is to build a linguistically analyzed, representative language resource for less studied language pairs dissimilar in language structure to be able to study the relations between these languages. The aim is to build a parallel treebank containing various annotation layers from part of speech tags and morphological features to dependency annotation where each layer is automatically annotated, the sentences and words are aligned, and partly manually corrected. The work described here

is part of the project *Supporting research environment for minor languages* initiated by professor Anna Sågvall Hein at Uppsala University. The project aims at building various types of language resources for Turkish and Hindi. We choose Swedish and Turkish, a less studied and typologically dissimilar language pair, to serve as a pilot study for building parallel treebanks for other language pairs. Therefore, efforts are put on developing a general method and using tools that can be applied to other language pairs easily.

The components of the language resource are texts that are in translational relation to each other and syntactically analyzed, and tools for the automatic analysis and alignment of these languages. To build a parallel corpus, we reuse existing resources and create necessary tools for the automatic processing and alignment of the parallel texts in these languages. The purpose is to build the corpus automatically by using a basic language resource kit (BLARK) for the particular languages and appropriate tools for the automatic alignment and correction of data. We use tools that are user-friendly, understandable and easy to learn by people with less computer skills, thereby allowing researchers and students to align and correct the corpus data by themselves. The parallel treebank is intended to be used in linguistic research, teaching and applications such as machine translation.

The paper is organized as follows: section 2 gives an overview of parallel corpora in general and parallel treebanks in particular; section 3 describes the parallel treebank, the methods used for building the treebank and the tools used for visualization, correction and investigation of the treebank. In section 4, we suggest some further improvements and lastly, in section 5, we conclude the paper.

2 Parallel Corpora and Parallel Treebanks

A parallel corpus is usually defined as a collection of original texts translated to another language where the texts, paragraphs, sentences, and words are typically linked to each other. One of the most well-known and frequently used parallel corpora is Europarl (Koehn, 2002) which is a collection of material including 11 European languages taken from the proceedings of the European Parliament. Another parallel corpus is the JRC-Acquis Multilingual Parallel Corpus (Steinberger et al., 2006). It is the largest existing parallel corpus of today concerning both its size and the number of languages covered. The corpus consists of documents of legislative text, covering a variety of domains for above 20 languages. Another often used resource is the Bible translated to a large number of languages and collected and annotated by Resnik et al. (1999). The OPUS corpus (Tiedemann and Nygaard, 2004) is another example of a freely available parallel language resource.

There are, of course, many other parallel corpus resources that contain sentences and words aligned in two languages only. Such corpora often exist for languages in Europe, for example the English-Norwegian Parallel Corpus (Oksefjell, 1999) and the ISJ-ELAN Slovene-English Parallel Corpus

(Erjavec, 2002). It is especially common to include English as one of the two languages in the pair. Parallel corpora that do not include English or another European language are rare.

Parallel treebanks belong to a fairly new type of language resource, consequently we find a smaller amount of resources of this type available. The Prague Czech-English Dependency Treebank (Hajic et al., 2001) is one of the earliest parallel treebanks, containing dependency annotation. The English-German parallel treebank (Cyrus et al., 2003) is another resource with multi-layer linguistic annotation including part of speech, constituent structures, functional relations, and predicate-argument structures. There are also small parallel treebanks including Swedish as one of the languages under development. The Linköping English-Swedish Parallel Treebank, also called LinES (Ahrenberg, 2007) contains approximately 1200 sentence pairs, annotated with PoS and dependency structures, and the Swedish-English-German treebank, SMULTRON (Gustafson-Capková et al., 2007), annotated with PoS and constituent structures.

In most parallel corpora including parallel treebanks, we find English and other structurally similar languages. However, there is a need to develop language resources in general, and parallel corpora and treebanks in particular, for other language pairs. Next, we describe the development of our Swedish-Turkish parallel treebank.

3 The Swedish-Turkish Parallel Treebank

First, we present the content and the annotation procedure of the treebank, then we give an overview of the tools that we use for the visualization and correction of the corpus annotation.

3.1 Corpus Content

The corpus, which has been previously described (Megyesi et al., 2006; Megyesi & Dahlqvist, 2007; and Megyesi et al., 2008) consists of original texts – both fiction and technical documents – and their translations from Turkish to Swedish and from Swedish to Turkish with the exception of one text which is a translation from Norwegian to both languages. In table 1, the corpus material is summarized.

The corpus consists of approximately 165,000 tokens in Swedish and 140,000 tokens in Turkish. Divided into text types, the fiction part of the corpus includes 76,877 tokens in Swedish, and 55,378 tokens in Turkish. The technical documents are larger and contain 90,901 tokens in Swedish, and 85,171 tokens in Turkish. The current material presented here serves as pilot linguistic data for the Swedish-Turkish parallel corpus. We intend to extend the material to other texts, both technical and fiction, in the future.

Document: Fiction	*# Tokens*	*# Types*
The White Castle – Swedish	53232	7748
The White Castle – Turkish	36684	12472
Sofie's world – Swedish	6488	1466
Sofie's world – Turkish	4800	2215
The Royal Physician's Visit – Swedish	17157	3932
The Royal Physician's Visit – Turkish	13894	5456
Document: Non-fiction		
Islam and Europe – Swedish	55945	10977
Islam and Europe – Turkish	48893	14128
Info about Sweden – Swedish	24107	4576
Info about Sweden – Turkish	23660	7119
Retirement – Swedish	3417	818
Retirement – Turkish	3664	1188
Dublin – Swedish	392	169
Dublin – Turkish	394	230
Pregnancy – Swedish	949	409
Pregnancy – Turkish	1042	567
Psychology – Swedish	347	193
Psychology – Turkish	281	220
Movement – Swedish	543	300
Movement – Turkish	568	369
Social security – Swedish	5201	846
Social security – Turkish	6669	2025

Table 1: The corpus data divided into text categories with number of tokens and types.

3.2 Corpus Annotation

The corpus material is processed automatically by using various tools making the annotation, alignment and manual correction easy and straightforward for users with less computer skills. This is necessary, as our ambition is to allow researchers and students of particular languages to enlarge the corpus by automatically processing and correcting the new data by themselves.

First, the original materials, i.e., the source and target texts received from the publishers in various formats are cleaned up. For example, rtf, doc, and

pdf documents are converted to plain text files. In the case of the original pdf-file, we scan and proof-read the material and, where necessary, correct it to ensure that the plain text file is complete and correct. After cleaning up the original data, the texts are processed automatically by using tools for formatting, linguistic annotation and sentence and word alignment. Figure 1 gives an overview of the main modules in the corpus annotation procedure.

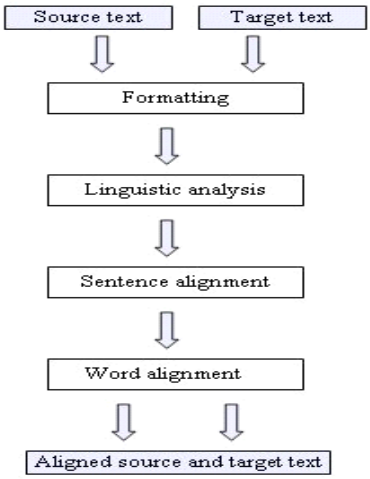

Figure 1: The modules of corpus annotation.

During formatting, the texts are encoded using UTF-8 (Unicode) and marked up structurally using XML Corpus Encoding Standard (XCES) for the annotation format. The plain text files are then processed by various tools in the BLARKs developed for each language separately when necessary. A sentence splitter is used to break the texts into sentences, and a tokenizer is used to separate words from punctuation marks.

Once the sentences and tokens are identified, the data is linguistically analyzed. For the linguistic annotation, external morphological analyzers, part of speech taggers and syntactic dependency parsers are used for the specific languages. We use several annotation layers for the linguistic analysis, first on a morphological level, then on a syntactic level.

The Swedish texts are morphologically annotated with the Trigrams 'n' Tags part of speech tagger (Brants, 2000), trained on Swedish (Megyesi, 2002) using the Stockholm-Umeå Corpus (SUC, 1997). The tokens are annotated with parts of speech and morphological features and are disambiguated according to the syntactic context. The results for the morphological annotation of Swedish show an accuracy of 96.6%. The most erroneous tags in the materials are: i) proper nouns which should be tagged as common nouns, ii) particles which should be tagged as adverbs, iii) prepositions which should be annotated as particles or adverbs, iv) nouns

with wrong morphological analysis and finally v) participles which should be tagged as verbs. These errors constitute 46% of all errors.

The Turkish material is analyzed morphologically by using an automatic morphological analyzer developed for Turkish (Oflazer, 1994). Each token in the text is segmented and annotated with morphological features including part of speech. The Turkish material is morphologically analyzed and disambiguated using a Turkish analyzer (Oflazer, 1994) and a disambiguator (Yuret and Türe, 2006). Evaluation of the Turkish tagging and disambiguation shows an average accuracy of 78.6%. Problematic confusions in the Turkish tagging seems to be between i) determiners and numerals, ii) postpositions in nominative and postpositions in genitive, and iii) determiners and pronouns. These errors account for 24.9% of all errors.

```
<s id="s11.4">
<w pos="DT_UTR_SIN_IND" head="3" deprel="DET" id="w11.4.1">Någon</w>
<w pos="JJ_POS_UTR_SIN_IND_NOM" head="3" deprel="DET" id="w11.4.2">annan</w>
<w pos="NN_UTR_SIN_IND_NOM" head="4" deprel="SUB" id="w11.4.3">titel</w>
<w pos="VB_PRT_SFO" head="0" deprel="ROOT" id="w11.4.4">fanns</w>
<w pos="AB" head="4" deprel="ADV" id="w11.4.5">inte</w>
<w pos="MAD" head="4" deprel="IP" id="w11.4.6">.</w>
</s>
<s id="s10.5">
<w pos="+Adj" head="3" deprel="MODIFIER" id="w10.5.1">Başka</w>
<w pos="+Num+Card^DB+Noun+Zero+A3sg+Pnon+Nom" head="6" deprel="SUBJECT"
        id="w10.5.2">bir</w>
<w pos="+Noun+A3sg+Pnon+Nom" head="6" deprel="OBJECT" id="w10.5.3">başlık</w>
<w pos="+Adj^DB+Verb+Zero+Past+A3sg" head="0" deprel="ROOT" id="w10.5.4">yoktu</w>
<w pos="+Punc" head="6" deprel="PUNC" id="w10.5.5">.</w>
</s>
```

Figure 2: An example of morphological and syntactic annotation in XCES format.

The other linguistic layer contains information about the syntactic analysis. For the grammatical description, we choose dependency rather than constituent structures, as the former has been shown to be well suited for both morphologically rich and free word order languages such as Turkish, and for morphologically simpler languages, like Swedish. Both the Swedish and the Turkish data are annotated syntactically using MaltParser (Nivre et al., 2006a), trained on the Swedish treebank Talbanken05 (Nivre et al., 2006b) and on the Metu-Sabancı Turkish Treebank (Oflazer et al., 2003), respectively. MaltParser was the best performing parser for both Swedish and Turkish in the CoNLL-X shared task on multilingual dependency parsing (Buchholz and Marsi, 2006), with a labeled dependency accuracy of 84.6% for Swedish and 65.7% for Turkish. Currently, we manually correct the syntactic annotation in each language. Figure 2 illustrates an example taken from Orhan Pamuk's book *The White Castle*, showing the morphological and syntactic annotation from the formatter and analyzers for

the sentence “Some other title did not exist.” in Swedish and Turkish in XCES format.

After the linguistic analysis, the sentences are aligned automatically, and the words are linked to each other in the two languages. We use standard techniques for the establishment of links between source and target language segments. Paragraphs and sentences are aligned by using the length-based approach developed by Gale and Church (1993).

Once the sentences are aligned in the source and target language, we send it for manual correction to a student who speaks both languages. We automatically compare the links before and after the manual correction and the user gets statistics about the differences. The results show that between 67% and 94% of the sentences were correctly aligned by the automatic sentence aligner depending on the text type.

Lastly, phrases and words are aligned using the clue alignment approach (Tiedemann, 2003), and the toolbox for statistical machine translation GIZA++ (Och and Ney, 2003). Results show that the word aligner aligned approximately 69% of the words correctly.

In addition to the automatic morpho-syntactic annotation and alignment, we correct the linguistic analysis and links manually, and visualize the corpus in different ways without showing the structural markup when used, for example, in teaching. These tools will be described next.

3.3 Tools for Visualization and Correction

In the project, our goal is to reuse and further develop freely available, system independent, user-friendly tools for the annotation, visualization, correction and search in our corpus, both considering the mono-lingual and the parallel treebanks.

As basis for the annotation, we use the Uplug toolkit which is a collection of tools for processing corpus data, created by Jörg Tiedemann (2003). Uplug is used for sentence splitting, tokenization, tagging by using external taggers, and paragraph, sentence and word alignment. All the essential processing tools are implemented in a graphical interface, UplugConnector (Megyesi and Dahqvist, 2007) which accesses both the modules in the Uplug toolkit (Tiedemann, 2003), and other programs for linguistic annotation.

The Uplug package consists of a number of perl scripts accessible by line commands with a large number of options and sometimes utilizing piping between commands. To facilitate easier access and usage of these scripts, a graphical user interface, UplugConnector, was developed in Java for the project. Here, the user can in a simple fashion choose a specific task to be performed and let the graphical user interface (GUI) set up the proper sequence of calls to Uplug and subsequently execute them. Figure 3 below illustrates the Uplug Connector interface.

The user can optionally give the location of the source and target files, decide where the output should be saved, and specify the encoding for the input and output files. For the markup, basic structural markup, sentence

segmentation, and tokenization are available. Further, the Uplug Connector GUI has been constructed to give the possibility to include calls to new scripts outside Uplug for complementary analysis, when such needs arise. The user can easily access another resource if the available ones do not fit his/her needs, for example an external tokenizer, sentence splitter, tagger or parser. In the toolkit, the user can also call for the sentence and word aligners and their visualization tools.

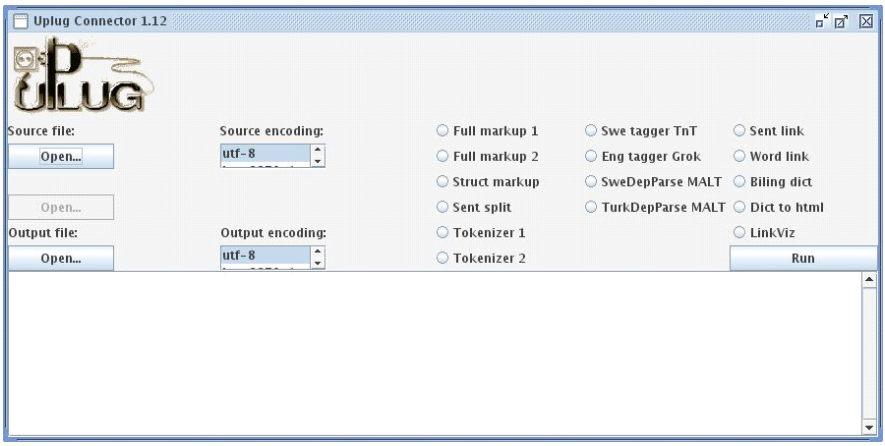

Figure 3: The Uplug Connector.

As the XML representation of the result is not user friendly even for people used to this kind of annotation, we use various interfaces for the visualization of the linguistic annotation and alignment results. In addition, since the automatic alignment generates some errors, we also use tools for the manual correction of these.

As a tool for the correction of the sentence alignment, we choose the system ISA (Interactive Sentence Alignment) developed by Tiedemann (2006). ISA is a graphical interface for automatic and manual sentence alignment which uses the alignment tools implemented in Uplug. It handles the manual correction of the sentence alignment in a user-friendly, interactive fashion. Figure 4 shows ISA with the aligned sentences taken from Orhan Pamuk's book *The White Castle*.

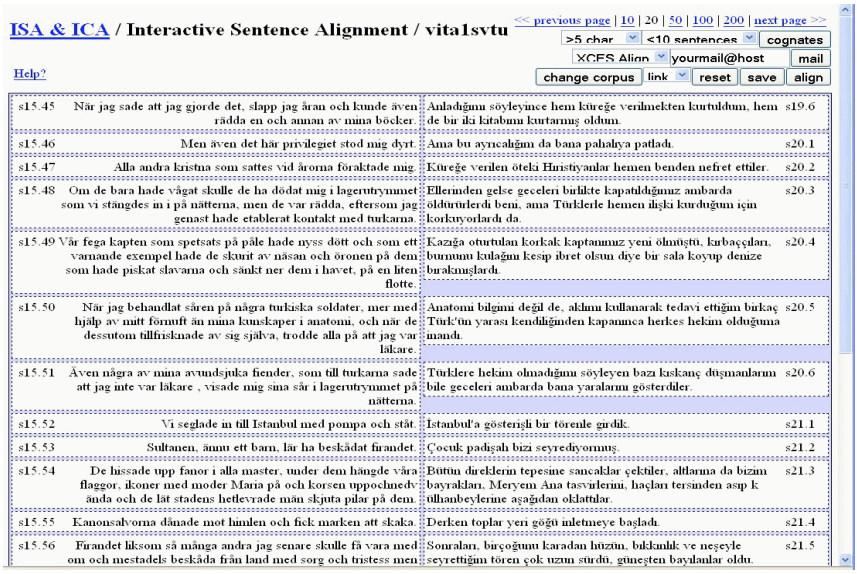

Figure 4: ISA showing the aligned sentences from *The White Castle*.

For displaying the corrected sentence output from ISA after manual correction of the alignment together with the linguistic analysis, a script utilizing the structural XML-parser Hpricot (2006) was developed. It takes as input the tagged XML-files for the language pair together with the XML file containing the sentence alignment results produced by ISA and generates an HTML-file which displays the sentences aligned together with the morphological information for each word shown in pop-up windows as shown in figure 5. The visualization tool makes it easier for students and researchers to study the part of speech and inflectional features of the words and chosen structures for translation than the structurally marked up version of the corpus.

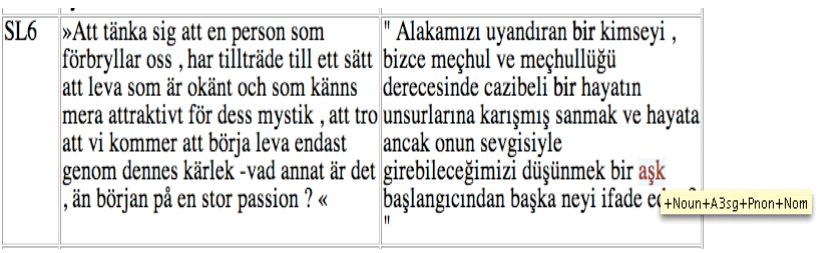

Figure 5: Visualization of aligned sentence pairs with linguistic annotation shown in the pop-up window.

To visualize the word alignment result in a simple way, a new script for HTML-visualization of the word alignment result was included in the UplugConnector. This takes as input the text file with word link information produced by Uplug, see figure 6, and shows the word-pair frequencies. This visualization actually presents a bilingual lexicon created from the source and target language data.

Sofies värld

Nr	Frekvens	Svenska	Turkiska
1	62	"	"
2	58	.	.
3	58	?	?
4	34	,	,
5	29	och	ve
6	23	Sofie	Sofie
7	18	Men	Ama
8	17	en	bir
9	14	!	!
10	14	:	:

Figure 6: HTML-visualization of word alignment.

For the visualization and correction of the parallel syntactic trees, we choose Stockholm Tree Aligner (Lundborg, et al., 2007).1 The tool allows the user to create links between corresponding nodes in two treebanks, hence allowing word and phrase alignment correction between our languages. The tool also contains a search function that implements the TigerSearch Query Language with additions for searching alignments. The visualization with Stockholm Tree Aligner for the sentence “Some other title did not exist.” is visualized as syntactic trees for Turkish and Swedish showing the dependency relations between the elements in each sentence in figure 7.

4 Further Developments

In the near future, we are going to apply the automatic annotation procedure on other languages of different types, such as Hindi and Persian and study the differences between the language pairs and the effects on the construction of parallel treebanks.

[1]See http://www.ling.su.se/dali/downloads/treealigner/index.htm.

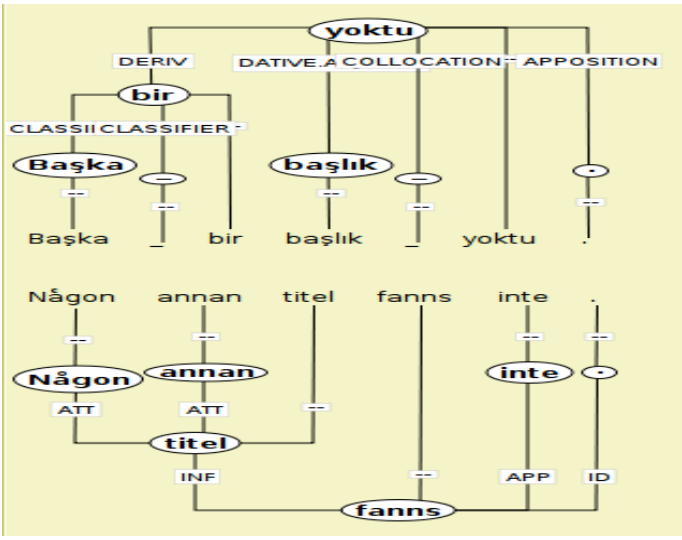

Figure 7: Dependency relations in Turkish and Swedish for the same sentence.

5 Conclusion

We have presented a Swedish-Turkish parallel treebank − a less processed language pair − containing approximately 165,000 tokens in Swedish, and 140,000 tokens in Turkish. The treebank is automatically created by re-using and adjusting existing tools for the automatic alignment and its visualization, and basic language resource kits for the automatic linguistic annotation of the involved languages. The automatic annotation and alignment is also partly manually corrected. The treebank is already in use in language teaching, primarily in Turkish.

Acknowledgments

We are grateful to Jörg Tiedemann for his kind support with Uplug, and Kemal Oflazer and Gülşen Eryiğit for the morphological annotation of Turkish. We would like to thank the publishers for allowing us to use the texts in the corpus. The project is financed by the Swedish Research Council and the Faculty of Languages at Uppsala University.

References

Abeillé, A. (ed.) (2003). *Building and Using Parsed Corpora*. Text, Speech and Language Technology. Kluwer, Dordrecht.

Ahrenberg, L. (2007). LinES: An English-Swedish Parallel Treebank. In *Proceedings of Nordiska Datalingvistdagarna*, NODALIDA 2007, Tartu, Estonia.

Ahrenberg, L., M. Merkel, and M. Andersson (2002). A system for incremental and interactive word linking. In *Proceedings from The Third International Conference on Language Resources and Evaluation* (LREC 2002), Las Palmas, 2002, pp. 485-490.

Brants, T. (2000) TnT − A Statistical Part-of-Speech Tagger. In *Proceedings of the 6th Applied Natural Language Processing Conference*. Seattle, USA.

Buchholz, S., and E. Marsi (2006). CoNLL-X Shared Task on Multilingual Dependency Parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, 149–164.

Church, K. W. (1993). Char align: A program for aligning parallel texts at the character level. In *Proceedings of the 31st Annual Conference of the Association for Computational Linguistics*, ACL.

Cyrus, L., H. Feddes, and F. Schumacher (2003). FuSe – A Multi-Layered Parallel Treebank. In *Proceedings of the Second Workshop on Treebanks and Linguistic Theories*, 14-15 November 2003, Växjö, Sweden.

Erjavec, T. (2002). The IJS-ELAN Slovene-English Parallel Corpus. *International Journal of Corpus Linguistics,* **7**(1), pp.1-20, 2002.

Gale, W. A. and K. W. Church (1993). A Program for Aligning Sentences in Bilingual Corpora. *Computational Linguistics,* **19**(1), 75-102.

Gustafson-Capková, S., Y. Samuelsson, and M. Volk (2007). *SMULTRON* (version 1.0) – The Stockholm MULtilingual parallel Treebank. An English-German-Swedish Parallel Treebank with Subsentential Alignments. http://www.ling.su.se/dali/research/smultron/index.htm.

Hajic, J., E. Hajicová, P. Pajas, J. Panevová, P. Sgall, and B. Vidová-Hladká (2001). *Prague Dependency Treebank 1.0* (Final Production Label). CDROM CAT: LDC2001T10., ISBN 1-58563-212-0, 2001.

Hpricot. A Fast, Enjoyable HTML and XML Parser for Ruby http://code.whytheluckystiff.net/hpricot/ 2006.

Ide, N. and G. Priest-Dorman. 2000. Corpus Encoding Standard – Document CES 1. Technical Report, Dept. of Computer Science, Vassar College, USA and Equipe Langue et Dialogue, France.

Koehn, P. (2002). Europarl: A Multilingual Corpus for Evaluation of Machine Translation. Information Sciences Institute, University of Souther California.

Lezius, W. (2002). TIGERSearch - Ein Suchwerkzeug für Baumbanken (German) in: Stephan Busemann (editor): Proceedings der 6. Konferenz zur Verarbeitung natürlicher Sprache (KONVENS 2002), Saarbrücken.

Lundborg, J., T. Marek, M. Mettler, M. Volk (2007). Using the Stockholm TreeAligner. In *Proceedings of the Sixth International Workshop on Treebanks and Linguistic Theories*. Editors: Koenraad De Smedt, Jan Hajič and Sandra Kübler. NEALT Proceedings Series, Vol. 1 (2007), 73-78. © 2007 The editors and contributors. Published by Northern European Association for Language Technology (NEALT)

MacIntyre, R. (1995). Penn Treebank tokenization on arbitrary raw text. http://www.cis.upenn.edu/~treebank/tokenization.html. University of Pennsylvania

Megyesi, B. (2002). *Data-Driven Syntactic Analysis − Methods and Applications for Swedish*. PhD Thesis. Kungliga Tekniska Högskolan. Sweden.

Megyesi, B. B., A. Sågvall Hein, and E. Csato Johanson (2006). Building a Swedish-Turkish Parallel Corpus. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation* (LREC 2006). Genoa, Italy.

Megyesi, B. B., A. Sågvall Hein, and E. Csato Johanson (2006). Building a Swedish-Turkish Parallel Corpus. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation* (LREC 2006). Genoa, Italy.

Megyesi, B. B. and B. Dahlqvist (2007). A Turkish-Swedish Parallel Corpus and Tools for its Creation. In *Proceeding of Nordiska Datalingvistdagarna*, NODALIDA 2007.

Megyesi, B. B., B. Dahlqvist, E. Pettersson, and J. Nivre (2008). Swedish Turkish Parallel Treebank. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation* (LREC 2008). Marocco.

Nivre, J., J. Hall and J. Nilsson (2006a). MaltParser: A Data-Driven Parser-Generator for Dependency Parsing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (LREC 2006), 2216–2219.

Nivre, J., J. Hall and J. Nilsson (2006b). Talbanken05: A Swedish Treebank

with Phrase Structure and Dependency Annotation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (LREC 2006), 1392–1395.

Oflazer, K. (1994). Two-level Description of Turkish Morphology. *Literary and Linguistic Computing*, **9**:2.

Oflazer, K., B. Say, and Hakkani-Tür (2003). *Building a Turkish Treebank.* In Anne Abeillé (ed.) Treebanks: Building and Using Parsed Corpora, Kluwer, 261–277.

Och, F. J. and H. Ney (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, volume **29**:1, pp. 19-51.

Oksefjell, S. (1999). A Description of the English-Norwegian Parallel Corpus: Compilation and Further Developments. *International Journal of Corpus Linguistics*, **4**:2, 197-219.

Resnik, P., M. Broman Olsen and M. Diab (1999). The Bible as a Parallel Corpus: Annotating the "Book of 2000 Tongues". *Computers and the Humanities*, **33:**1-2, pp. 129-153, 1999.

Samuelsson, Y. and M. Volk (2006). Phrase alignment in parallel treebanks. In Jan Hajiç and Joakim Nivre, eds. *Proceedings of the 5th Workshop on Treebanks and Linguistic Theories*, pp. 92-101, Prague.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufiş, and D. Varga (2006). The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation* (LREC 2006). Genoa, Italy, 24-26 May 2006.

SUC. Department of Linguistics, Umeå University and Stockholm University. 1997. SUC 1.0 Stockholm Umeå Corpus, Version 1.0. ISBN:91-7191-348-3.

Tiedemann, J. (2003). *Recycling Translations − Extraction of Lexical Data from Parallel Corpora and their Applications in Natural Language Processing.* PhD Thesis. Uppsala University.

Tiedemann, J. (2004). Word to word alignment strategies. In *Proceedings of the 20th International Conference on Computational Linguistics* (COLING 2004). Geneva, Switzerland, August 23-27.

Tiedemann, J. and L. Nygaard (2004). The OPUS corpus − parallel & free. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation* (LREC 2004). Lisbon, Portugal, May 26-28,

2004.

Tiedemann, J. (2005). Optimisation of Word Alignment Clues. In *Journal of Natural Language Engineering*, Special Issue on Parallel Texts, Rada Mihalcea and Michel Simard, Cambridge University Press.

Tiedemann, J. (2006). ISA & ICA – Two Web Interfaces for Interactive Alignment of Bitext. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation* (LREC 2006). Genoa, Italy.

Yuret, D. and F. Türe (2006). Learning morphological disambiguation rules for Turkish. In *Proceedings of HLT NAACL 2006*, pages 328-334, New York, NY.

Cultivating a Swedish Treebank

Joakim Nivre
Beáta Megyesi
Sofia Gustafson-Čapková
Filip Salomonsson
Bengt Dahlqvist

Uppsala University
Department of Linguistics and Philology

1 Introduction

Treebanks, or syntactically annotated corpora, are an invaluable resource for the development and evaluation of syntactic parsers, as well as for empirical research on natural language syntax. Treebanks for Swedish have a long and venerable history, represented by the pioneering work on Talbanken (Einarsson, 1976a,b) and SynTag (Järborg, 1986). In recent years, however, treebank development for Swedish has mostly been limited to smaller projects, such as the reconstruction of Talbanken into Talbanken05 (Nivre et al., 2006), efforts to create a Swedish treebank with texts from the medical domain (Kokkinakis, 2006), and development of small parallel treebanks for Swedish-English-German (Gustafson-Čapková et al., 2007), and Swedish-English (Ahrenberg, 2007). As a consequence, there is still no Swedish treebank of the same scale as the largest available treebanks, such as the Penn Treebank for English (Marcus et al., 1993) or the Prague Dependency Treebank for Czech (Hajič et al., 2001).

Given the high cost of manual annotation and post-editing in treebank development, the possibility to reuse existing annotated resources is potentially of great importance. Often the efficient reuse of such resources is hampered by the fact that different resources, even for the same language, have been developed with different annotation guidelines or encoding standards. In many cases, however, it is possible to overcome these obstacles through a process of cross-corpus harmonization and annotation projection.

In this paper, we describe an ongoing project with the aim of bootstrapping a large Swedish treebank, ultimately with a size of about 1.5 million tokens, by reusing two existing annotated corpora: the previously mentioned treebank Talbanken, consisting of about 350,000 tokens, and the more recent Stockholm-Umeå Corpus (Ejerhed and Källgren, 1997), a part-of-speech-tagged corpus of about 1.2 million words. This treebanking effort is part of the

project *Methods and Tools for Automatic Grammar Extraction*, supported by the Swedish Research Council and initiated by Anna Sågvall Hein at Uppsala University. Besides treebank development, this project involves research on grammar induction with evaluation in the context of machine translation.

The paper is structured as follows. We first give an overview of the project and the different steps needed to develop a new treebank from existing resources and then focus on the two most interesting steps: the harmonization of tokenization and sentence segmentation, and the projection of annotation from one corpus to the other using data-driven taggers and parsers. The latter step also involves annotation refinement, where the syntactic annotation in Talbanken is extended and modified to better suit present-day requirements Finally, we briefly discuss how much is gained by reusing existing corpora and annotation, as opposed to creating a new treebank from scratch.

2 Bootstrapping a Large Swedish Treebank

Our ultimate goal is to produce a Swedish treebank containing 1.5 million tokens by reusing two existing annotated corpora. In section 2.1, we describe important properties of the two corpora; in section 2.2 we describe the major steps that need to be taken to reuse them as part of a single, consistently annotated treebank.

P10835069001	0000	«	GM	074
P10835069002	*DEN	PODPHH	SS	074
P10835069003	1000	RC	SSET	074
P1083506900410002	SOM	PORPHH	SS	074
P1083506900510002	VÄNTAR	VVPS	FV	074
P1083506900610002	MED	PR	OAPR	074
P1083506900710002	1100	IF	OA	074
P1083506900811003	ATT	IM	IM	074
P1083506900911003	TA	VVIV	IV	074
P1083506901011003	UT	ABZA	PL	074
P1083506901111003	ÅLDERSPENSIONEN	NNDDSS	OO	074
P1083506901210002	TILL	PR	TAPR	074
P1083506901310002	EFTER	PR	TATAPR	074
P1083506901410002	67-ÅRSMÅNADEN	NNDDSS	TATA	074
P10835069015	FÅR	FVPS	FV	074
P10835069016	HÖGRE	AJKP	OOAT	074
P10835069017	PENSION	NN	OO	074
P10835069018	.	IP	IP	074

Figure 1: Annotated sentence from Talbanken: *Den som väntar med att ta ut ålderspensionen till efter 67-årsmånaden får högre pension* (Those who do not claim their old age pension until after the 67-year month get a higher pension).

2.1 Component Corpora

Talbanken (Einarsson, 1976a,b) is a syntactically annotated corpus, containing both written and spoken Swedish, produced in the 1970s at the Department of Scandinavian Languages, Lund University, by a group led by Ulf Teleman. In total, the corpus contains about 350,000 tokens, divided into 200,000 tokens of written text (professional prose and high school essays) and 150,000 tokens of spoken language (interviews, debates, and informal conversations). The annotation consists of two layers: a lexical layer, with parts of speech and morphosyntactic features, and a syntactic layer, with a relatively flat phrase structure and grammatical functions (or dependencies). The annotation scheme, known as MAMBA, is described in Teleman (1974) and illustrated in figure 1, which shows a small extract from Talbanken.

The main asset of Talbanken, from our point of view, resides in the syntactic annotation, which contains enough information to support the extraction of both phrase structure and dependency structure representations, as shown in Nilsson et al. (2005) and Nivre et al. (2006), and therefore provides a good base representation for a treebank. Moreover, since Talbanken is by far the largest available corpus of Swedish with manually validated syntactic annotation, including it in the new treebank not only lets us reuse a manually validated syntactic annotation of 350,000 tokens, but also gives us a good basis for training parsers that can be used in the annotation of additional data.

The Stockholm-Umeå Corpus (SUC) (Ejerhed and Källgren, 1997) is a balanced corpus of written Swedish, modeled after the Brown Corpus and similar corpora for English, developed at Stockholm University and at Umeå University in a project led by Gunnel Källgren and Eva Ejerhed. The corpus consists of 1.2 million tokens of text from a variety of different genres, the corpus encoding follows the guidelines of the Text Encoding Initiative (TEI), and the annotation includes lemmatization, parts of speech, morphosyntactic features, and named entities. Since SUC was first released in the 1990s, its annotation scheme has become a de facto standard for Swedish, especially in research on part-of-speech tagging, where SUC data is standardly used for training and evaluation (see, e.g., Carlberger and Kann, 1999; Nivre, 2000; Megyesi, 2002). The annotation scheme is illustrated in figure 2, which shows a small extract from the corpus.

Given that SUC is a larger and more recently developed corpus, which has been extensively used to train taggers and other tools for Swedish, it makes sense to use SUC as a model for the new treebank wherever possible, thus minimizing the need for (new) manual validation and maximizing the conformance with current practice in Swedish language technology. This means, among other things, that principles of tokenization and sentence segmentation should be kept intact in SUC but modified for Talbanken in cases of conflict. We will refer to this as the *harmonization* of tokenization and sentence segmentation. The same holds for the annotation of parts of speech and mor-

```
<s id=fh06-089>
<w n=1487>Senare<ana><ps>AB<m>KOM<b>sen</w>
<w n=1488>på<ana><ps>PP<b>på</w>
<w n=1489>1940-talet<ana><ps>NN<m>NEU SIN DEF NOM<b>1940-tal</w>
<w n=1490>byggde<ana><ps>VB<m>PRT AKT<b>bygga</w>
<NAME TYPE=PERSON>
<w n=1491>John<ana><ps>PM<m>NOM<b>John</w>
<w n=1492>von<ana><ps>PM<m>NOM<b>von</w>
<w n=1493>Neumann<ana><ps>PM<m>NOM<b>Neumann</w>
</NAME>
<w n=1494>i<ana><ps>PP<b>i</w>
<NAME TYPE=PLACE>
<w n=1495>Princeton<ana><ps>PM<m>NOM<b>Princeton</w>
</NAME>
<w n=1496>i<ana><ps>PP<b>i</w>
<NAME TYPE=PLACE>
<ABBR>
<w n=1497>USA<ana><ps>PM<m>NOM<b>USA</w>
</ABBR>
</NAME>
<w n=1498>sina<ana><ps>PS<m>UTR/NEU PLU DEF<b>sin</w>
<num>
<w n=1499>första<ana><ps>RO<m>NOM<b>första</w>
</num>
<w n=1500>datamaskiner<ana><ps>NN<m>UTR PLU IND NOM<b>datamaskin</w>
<d n=1501>.<ana><ps>MAD<b>.</d>
</s>
```

Figure 2: Annotated sentence from the Stockholm-Umeå Corpus: *Senare på 1940-talet byggde John von Neumann i Princeton i USA sina första datamaskiner* (Later in the 1940s John von Neumann at Princeton in the USA built his first computers).

phosyntactic features, where the kind of annotation used in SUC has to be *projected* to Talbanken, which unfortunately uses a different scheme.[1] Since no simple mapping exists from the Talbanken scheme to the SUC scheme (nor in the other direction), this projection will have to be induced by training a tagger on the SUC corpus, using it to reannotate Talbanken, and finally correcting the errors performed by the tagger in a manual post-editing phase. In the following, we will use the term *morphological annotation* (in contrast to *syntactic annotation*) to include both basic parts of speech and morphosyntactic features.

[1]Other kinds of annotation found in SUC, such as lemmatization and named entities, are outside the scope of the current project but should in principle be projected in the same way from SUC to Talbanken.

2.2 Treebank Development

Given the considerations so far, we propose the following overall plan for the production of a new treebank based on Talbanken and SUC:

1. Convert both corpora with their existing annotation into a common standard for corpus encoding (XCES with standoff annotation).
2. Harmonize tokenization and sentence segmentation in Talbanken, applying as far as possible the principles adopted in SUC.
3. Project morphological annotation from SUC to Talbanken, using a data-driven tagger trained on SUC with manual post-editing.
4. Refine the syntactic annotation in Talbanken by automatic inference.
5. Project syntactic annotation from Talbanken to SUC, using a data-driven parser trained on Talbanken with manual post-editing.

In the following two sections, we describe the problems involved in harmonization, annotation refinement and annotation projection in a little more detail.

3 Harmonization

To harmonize the two corpora, we convert the tokenization and sentence segmentation of Talbanken according to the principles of SUC.

3.1 Tokenization

In the tokenization of SUC, abbreviations are always represented as single tokens. This means that when abbreviations in the original text contain spaces, the different elements are concatenated into one token where spaces in the original text are represented by underscores. Moreover, different variants of the same abbreviations are normalized to one form. Thus, the following variants of the abbreviation of *till exempel* (for example):

t. ex.
t ex

are all tokenized as one token:

t_ex

In Talbanken, on the other hand, abbreviations consisting of several elements are annotated as multi-word expressions. Each element of abbreviation is treated as a separate token, but only the first token is assigned proper lexical and syntactic annotation, while the subsequent token are assigned the dummy

tag ID (in both the lexical and the syntactic annotation). To find the abbreviations in Talbanken, we automatically extract tokens annotated with ID tags together with the preceding token, and convert these into a single token with the form prescribed by SUC's tokenization standards.

Certain numerical expressions, such as *3–5*, are also tokenized differently in the two corpora, such that SUC often has a single token where Talbanken splits the expression into several tokens (which may or may not be annotated as a multi-word expression). For certain types of numerical expressions, it is again possible to perform the harmonization automatically, but most cases here need to be checked manually.

3.2 Sentence Segmentation

The sentence segmentation also differs in the two corpora. Above all, lists have a different structural annotation. In SUC, items in lists are handled as different sentence units, while in Talbanken the entire list consisting of several items is treated as one sentence if there are clear syntactic relations between the items. This could lead to errors when we use a data-driven parser to project the syntactic annotation from Talbanken to SUC, since the sentences in Talbanken that would serve as training data would have a different structure compared to the sentences in SUC that need to be parsed. Therefore, we treat each list item in Talbanken as a separate sentence unit as far as possible.

4 Annotation Projection and Refinement

4.1 Morphological Annotation

In order to harmonize the morphological annotation of the two corpora, we project the part-of-speech tags and morphological features from SUC to Talbanken. We do this by training the data-driven TnT tagger (Brants, 2000) on SUC, bootstrapping the tagger by training it on a considerably larger automatically tagged corpus (Forsbom, 2005), and then applying the trained model to Talbanken. Finally, we correct the automatic annotation manually following SUC's annotation principles. The result is a merged corpus with consistent morphological annotation.

At the time of writing, all closed class words in Talbanken have been checked and we predict that the work on morphological annotation will be completed during the spring of 2008. One of the advantages of reusing a previously annotated corpus, even if the annotation is inconsistent, is that checking can be speeded up by pattern matching on the combined old and new annotation. This is convenient especially for closed word classes, where the lack of morphological features often makes the two annotation systems equivalent so that some words need to be checked only if the new and the old annotations are inconsistent.

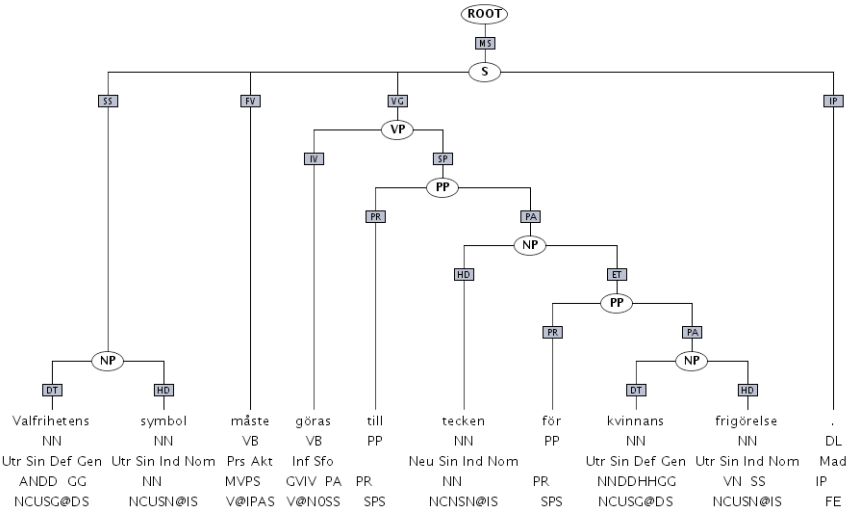

Figure 3: Refined syntactic annotation for the sentence *Valfrihetens symbol måste göras till tecken för kvinnans frigörelse* (The symbol of free choice must be turned into a sign of women's liberation).

4.2 Syntactic Annotation

For the syntactic annotation, we have to project the syntactic analysis of Talbanken to SUC, since SUC lacks syntactic annotation. This is achieved in the first place by training the data-driven MaltParser (Nivre et al., 2006) on Talbanken and using the trained model to parse SUC. At a later stage, we may complement this annotation with the output of different parsers, which will enable us to use an ensemble of classifiers to facilitate the manual correction of the syntactic annotation.

However, before we can train parsers on Talbanken, the syntactic annotation needs to be refined to meet the requirements of present-day language technology. The original annotation has very flat structures and lacks phrase categories such as NP, VP, PP, etc. In order to obtain a hierarchical analysis involving both phrase structure and dependency structure (grammatical functions), we therefore need to do two things:

- Infer phrase categories using information about grammatical functions (both internal and external to a given phrase) and about the parts of speech of constituent words.
- Infer additional structure, such as NPs within PPs, VPs within VPs, based on information about grammatical functions, parts of speech, and inferred phrase categories.

The methodology for automatic annotation refinement is described in Nilsson et al. (2005). Figure 3 shows an example of the current version of the refined syntactic annotation. Our goal is to complete this work during the spring of 2008, which means that a first version of the entire treebank, with purely automatic syntactic annotation in the SUC part, could be released in the fall of 2008. A version where all the annotation has been checked manually remains as a long-term goal for our efforts.

5 How Much Is Gained?

A reasonable question to ask is how much is actually gained by reusing existing corpora, as opposed to building a new treebank from scratch, given the considerable amount of work involved in the harmonization and projection processes. Let us therefore make an attempt at quantifying the gains and balancing them against the disadvantages.

By reusing all the annotation in SUC and the syntactic annotation in Talbanken, we save all the work needed to manually correct tokenization, sentence segmentation, and morphological annotation of 1.2 million tokens, and syntactic annotation of 350,000 tokens. In addition, we save the work needed to check tokenization and sentence segmentation for 350,000 tokens in Talbanken, minus a few person weeks spent on harmonization. Finally, although the morphological annotation of 350,000 tokens in Talbanken still has to be checked manually, both the efficiency and the accuracy of this process can be improved by making use of the old morphological annotation for consistency checking.

To give just one illustrative example, the string *men* in Swedish can be either a coordinating conjunction (but) or a noun (injury). After projecting the new morphological annotation from SUC to Talbanken, it was found that one occurrence of *men* was tagged as a noun in the old annotation and as a conjunction in the new annotation, whereas the remaining 364 occurrences were tagged as conjunctions in both cases. Unsurprisingly, the single occurrence with inconsistent annotation turned out to be a tagging error, which in this way could be detected and corrected. With very high probability, the remaining 364 occurrences are correctly tagged as conjunctions (since the old annotation has been checked manually) and therefore do not need to be checked.[2]

To sum up, we see that cross-corpus harmonization and annotation projection can lead to substantial gains in the manual work needed to validate segmentation and annotation. This of course has to be weighed against a number of other factors, in particular that the new treebank has to be based on old data (in the case of Talbanken, texts from the 1970s) and that the annotation

[2]Other examples are *man*, which is ambiguous between a pronoun (one) with 699 occurrences and a noun (man) with 67 occurrences, and *Vi*, which has a single occurrence as the name of a magazine and 328 occurrences as a capitalized pronoun (we).

schemes have to be inherited from at least one of the old corpora. Still, in situations where manual effort has to be minimized, the approach taken appears to be a viable methodology for producing a large-scale treebank from existing resources.

6 Conclusion

In this paper, we have presented ongoing work to produce a large treebank of Swedish by reusing two existing annotated corpora, Talbanken and SUC. A key component in the bootstrapping methodology is the use of cross-corpus harmonization and annotation projection, supported by automatic conversion procedures and data-driven linguistic analyzers, with a minimum of manual validation. In this way, we hope to be able to create a large-scale, high-quality Swedish treebank, a resource that is badly needed for research and development in language technology, as well as for empirical linguistic research.

References

Ahrenberg, L. (2007). LinES: An English-Swedish parallel treebank. In *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA)*, pp. 270–273.

Brants, T. (2000). TnT – a statistical part-of-speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP)*.

Carlberger, J. and V. Kann (1999). Implementing an efficient part-of-speech tagger. *Software Practice and Experience 29*, 815–832.

Einarsson, J. (1976a). Talbankens skriftspråkskonkordans. Lund University, Department of Scandinavian Languages.

Einarsson, J. (1976b). Talbankens talspråkskonkordans. Lund University, Department of Scandinavian Languages.

Ejerhed, E. and G. Källgren (1997). Stockholm Umeå Corpus. Version 1.0. Produced by Department of Linguistics, Umeå University and Department of Linguistics, Stockholm University. ISBN 91-7191-348-3.

Forsbom, E. (2005). Big is beautiful: Bootstrapping a pos tagger for swedish. Poster presentation at the GSLT retreat. Gullmarsstrand, January 27-29.

Gustafson-Čapková, S., Y. Samuelsson, and M. Volk (2007). SMULTRON (version 1.0) – The Stockholm MULtilingual parallel TReebank. http://www.ling.su.se/dali/research/ smultron/index.htm. An English-German-Swedish parallel treebank with sub-sentential alignments.

Hajič, J., B. Vidova Hladka, J. Panevová, E. Hajičová, P. Sgall, and P. Pajas (2001). Prague Dependency Treebank 1.0. LDC, 2001T10.

Järborg, J. (1986). Manual för syntaggning. Technical report, Göteborg University, Department of Swedish.

Kokkinakis, D. (2006). Towards a swedish medical treebank. In J. Hajič and J. Nivre (Eds.), *Proceedings of the Fifth Workshop on Treebanks and Linguistic Theories*, pp. 199–210.

Marcus, M. P., B. Santorini, and M. A. Marcinkiewicz (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics 19*, 313–330.

Megyesi, B. (2002). *Data-Driven Syntactic Analysis: Methods and Applications for Swedish*. Ph. D. thesis, KTH: Department of Speech, Music and Hearing.

Nilsson, J., J. Hall, and J. Nivre (2005). MAMBA meets TIGER: Reconstructing a Swedish treebank from Antiquity. In P. J. Henrichsen (Ed.), *Proceedings of the NODALIDA Special Session on Treebanks*.

Nivre, J. (2000). Sparse data and smoothing in statistical part-of-speech tagging. *Journal of Quantitative Linguistics 7*, 1–17.

Nivre, J., J. Hall, and J. Nilsson (2006). Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pp. 2216–2219.

Nivre, J., J. Nilsson, and J. Hall (2006). Talbanken05: A swedish treebank with phrase structure and dependency annotation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, pp. 1392–1395.

Teleman, U. (1974). *Manual för grammatisk beskrivning av talad och skriven svenska*. Studentlitteratur.

Oversettelsesassistenten

Torbjørn Nordgård

LingIT AS and NTNU
Norway

1 Introduction

In Norway, as in many other small language communities, there is a shortage of high quality bilingual corpus data. Thus it is hard to make use of standard Statistical Machine Translation (SMT) techniques. So, in this paper we will discuss scalability and domain adaptation for an experimental machine translation system that is easy to assemble and which requires standard and generally available machine-readable linguistic data. The system, called *Oversettelsesassistenten* (*The Translation Assistant*), henceforth *OA*, was developed at NTNU in 2005 - 2007 as a prototype that makes use of readily available resources like monolingual full form word lists, machine readable lexicographic dictionaries, POS taggers, finite state automata and monolingual corpora. OA has already been used as a reference system in the evaluation of the LOGON system (see Oepen et al, 2004).[1]

2 OA Components

Oversettelsesassistenten contains the following components, ordered sequentially as displayed in the figure:

POS Tagger ⇒ Finite State Transducer ⇒ Decoder

The components are described below, but first we give an overview of the linguistic resources used. Source language full forms are taken from NorKompLeks (Nordgård, 2000), target language full forms are extracted from the English section of Multext (Ide and Véronis, 1994), target language corpus is the NYT sections of English Gigaword. Translation correspondences were extracted from *Engelsk stor ordbok* (Henriksen et.al., 2001). These resources have been automatically prepared in order to obtain full form correspondences from Norwegian to English. The resulting resources have not been inspected thus far, except for grammatical tag correlations between the various sources.

[1] This work was done while I worked at NTNU, partly together with Ola Huseth.

2.1 POS Tagger

The tagger has a trigram HMM engine with a Viterbi decoder and linear interpolation between n-gram models. It has as suffix guessing facility for unknown words, in addition to intuitive and simple "tricks" for recognition of proper nouns and numbers. The tagset comprises 44 tags in total. The training data set is quite small; 101488 words and 5551 sentences. The data is collected from Norwegian newspapers, and it is annotated by students from NTNU. Tagging accuracy is 95%.

An example from the tagger in operation: The transition path for the observation sequence "Jeg synes at tagging er morsomt." is

pron	*pronoun*
verb_pres	*verb in present tense*
subj	*"subjunction", i.e., complementizer*
noun_sg_indef	*noun, singular and indefinite form*
verb_pres	*verb in present tense*
adj_pos	*adjective in positive degree*
EOS	*end of sentence*

which gives the tagging result

Jeg<pron>
synes<verb_pres>
at<subj>
tagging<noun_sg_indef>
er<verb_pres>
morsomt<adj_pos>
.<EOS>

The tagset is adapted to the goal of the system, i.e., translation from Norwegian to English. This is why morphological features like gender are absent in the tagging result.

2.2 Finite State Transducer

When the tagger has finished its work, the result is passed to a Finite State Transducer (FST). The objective of the FST is to prepare the source string for the decoder by performing actions like

- insertion of grammatical words (e.g.; *the*)
- enforcing "do-support"
- possessive fronting (e.g.; katten min → min katt)

- perform "safe" initial translations of phrases (e.g.; *i morgen* → *tomorrow*)

The general idea of preparing source language patterns for translation into the target language is well-known in the MT field. Even though the method has limitations it is nonetheless an efficient and powerful strategy as long as it is used with care.

In OA the transduction is implemented as Prolog DCG rules. The DCG formalism by itself is far more powerful than finite state grammars, but all of the defined transducer rules are within what FST machines are able to do. Crucially, the DCG formalism makes it easy to formulate linear recursive deterministic phrase scanning when the input sequence is a POS tagged expression. Consider the following input sequence to the transducer:

```
tg(['XSTARTX', 'Han':'pron',
    'beundrer':'verb_pres', 'ikke':'adv',
    'danskene':'nun_pl_def','.':'EOS']).
```

The pronoun "Han" is unambiguously third person singular, and it is immediately followed by a verb in the present tense and an adverb. In English this syntactic configuration requires do-support. The object NP *danskene* is in the definite form, which should be preceded by the definite article in the target language. From this information the transducer is able to produce

```
[XSTARTX, han:pron, does:_, ikke:adv,
    beundrer:vform=inf, the:_,
    danskene:noun_pl_def, .:EOS]
```

That is, do-support with the third person singular form of the verb (*does*), a restriction on the form of the main verb *beundrer* (the translation of *liker* must be in the infinitival form), and insertion of *the* in front of the definite NP *danskene*. These modifications of the input are the results from the operations of two FST rules. The output of the tagger is then given to the translation decoder (see next section).

In short, the job of the transducer is to adjust the input prior to the general lexical translation operations.

2.3 Translation Decoder

The translation decoder has the general properties of a standard HMM engine. The observation sequence is the output from the FST described above. The transitions are the set of possible translations that are compatible with the elements of the observation elements.

Since the availability of bilingual language corpora is very limited, translation probabilities on the word and phrase level are approximated on

data from a small bilingual corpus so that dictionary defined base form correspondences are counted. The probabilities are evenly distributed over the entire inflection paradigms. The effect of this strategy is that we have some indications of a prior probability that can be used (i.e., translation correspondences that are attested). It is better than nothing, but if a larger amount of bilingual data becomes available, the prior selection of translation candidates will be far more reliable. Luckily, there are quite a lot of available English text data. Thus, n-grams calculated from the language model play a very significant role in the system.

When the decoding table is filled, the bilingual full form lexicon is consulted. If no restriction is placed on a source item, all translation possibilities are added to the network. If there are restrictions present (as vform=inf, see above), target language items that adhere to the restrictions are selected. If a target language item has been selected by the transducer (for instance the:_, cf above), it will be the only element at that position in the network.

Since we are using a trigram language model the states in the network are pairs of items from the target language (technically, one element of a pair might be a multiword element, but we leave those issues aside here). The most relevant parts of the decoding table for the translation of the Norwegian sentence *Noen studenter skrev en interessant artikkel* is provided below. By following the back pointers from the best final state in the network (i.e., story.), the reader can confirm that the best translation is *Some students wrote an interesting story.*

0 XSTARTX any -13.89 BP: -1
0 XSTARTX anybody -20.22 BP: -1
0 XSTARTX anyone -17.60 BP: -1
0 XSTARTX anything -16.53 BP: -1
0 XSTARTX some -12.70 BP: -1
0 XSTARTX something -15.33 BP: -1
1 any students -37.47 BP: START any
1 anybody students -60.74 BP: START anybody
1 anyone students -52.89 BP: START anyone
1 anything students -49.69 BP: START anything
1 some students -32.14 BP: START some
1 something students -46.08 BP: START something
2 students scribbled -58.04 BP: some students
2 students typed -60.34 BP: some students
2 students wrote -46.96 BP: some students
3 scribbled a -78.42 BP: students scribbled
3 scribbled about -103.86 BP: students scribbled
3 scribbled an -103.038 BP: students scribbled
3 scribbled one -102.80 BP: students scribbled
3 typed a -81.78 BP: students typed
3 typed an -83.98 BP: students typed
3 typed one -81.95 BP: students typed

3 wrote a -57.66 BP: students wrote
3 wrote an -59.50 BP: students wrote
3 wrote one -60.38 BP: students wrote
4 a interesting -66.91 BP: wrote a
4 an interesting -69.80 BP: wrote an
4 one interesting -75.67 BP: wrote one
5 interesting article -74.49 BP: an interesting
5 interesting clause -94.96 BP: a interesting
5 interesting item -95.47 BP: a interesting
5 interesting piece -77.97 BP: an interesting
5 interesting product -79.79 BP: an interesting
5 interesting section -86.02 BP: a interesting
5 interesting story -76.16 BP: an interesting
6 article . -135.27 BP: interesting article
6 clause . -170.19 BP: interesting clause
6 item . -170.70 BP: interesting item
6 piece . -138.73 BP: interesting piece
6 product . -143.54 BP: interesting product
6 section . -149.76 BP: interesting section
6 story . -105.03 BP: interesting story

Note that locally "wrote a", "a interesting" and "interesting article" have better values than "wrote an", "an interesting" and "interesting story", but the latter alternatives are chosen during back pointer traversal. This phenomenon is a well-known property of HMMs.

The current system is limited to n-gram size 3. It is reasonable to assume that if the n-gram size is increased translation quality will increase as well, provided the amount of training data is sufficient. We leave this issue to future versions of OA.

3 Domain Adaptation

From the LOGON project there are some Norwegian - English corpus data available: training and test data with reference translations. The data is thematically tied to mountain hiking. In the experiment we have used three distinct Norwegian English corpora:

- LOGON's initial small training data (the *Tur* corpus), consisting of 104 Norwegian source sentences and three English reference translations for each sentence.

- LOGON's main training corpus (the *Jotunheimen* corpus), which has 2146 Norwegian sentences with three reference translations each.

- One of LOGON's test corpora (the Jotunheimen test set, unknown vocabulary part), containing 92 test sentences and three reference

translations. The notion "unknown vocabulary" is irrelevant for our purposes since the test data is completely unknown to the OA system, and no operations are made in order to prepare the system for this particular test data set.

3.1 *Tur* Adaptation

First we adapt the system to the small initial training data by using lexical FST rules, but initially we test the system with no adaptations to the *Tur* data set. Then we modify the target language model by including reference translations from the *Jotunheimen* corpus (the source sentences in *Tur* and *Jotunheimen* are not overlapping). In the next experiment we use FST rules derived from properties of the test document, but language models are not "boosted" by domain sentences as in the second experiment. And finally, we include both new rules and target domain adaptation. The results are as follows:

- No rules based on the training document, no domain n-grams: BLEU SCORE: 0.3141
- No domain rules, domain n-grams from the *Jotunheimen* Corpus: BLEU SCORE: 0.3627
- With rules based on the training document, no domain n-grams: BLEU SCORE: 0.4498
- With rules based on the training document, domain n-grams from the *Jotunheimen* Corpus: BLEU SCORE: 0.5221

If we take the first experiment as baseline, we observe a 15% increase in BLEU score when domain n-grams are included. Rule adaptations give a very clear effect: 43% increase in BLEU score, and the combination of these results in an increase of 66%. The resulting BLEU score is quite respectable. The effects of both strategies provide very nice BLEU score increase, but the rules based on this particular corpus is of course methodologically dubious with respect to general capability of the system when it is confronted with data from other sources, as we will see below.

3.2 *Jotunheimen* Training Data

In the next experiment we perform three tests on the *Jotunheimen* data set. The first test is not optimized by any of the two strategies, the second makes use of the rules made on the basis of *Tur*, and the third uses n-grams from the *Jotunheimen* data set itself:

- No rules based on the *Tur* data set, no domain n-grams: BLEU SCORE: 0.3146

- With rules based on the *Tur* data set, no domain n-grams: BLEU SCORE: 0.3181

- With rules based on the *Tur* data set, domain n-grams from the *Jotunheimen* Corpus: BLEU SCORE: 0.3795

Note that the effect of the *Tur* rules is hardly recognizable, but the effect of n-grams from the same test set is obvious, i.e., an increase of 20%. The reason why the *Tur* rules has so limited effect must be attributed to the fact that these rules are very lexically dependent, and they only have effect when exact triggering strings are observed in the source tagging sequence.

3.3 *Jotunheimen* Test Data

Let us now go into one of the held-out sections of the *Jotunheimen* data set. Note at the outset that this appears to be much "easier" sentences than those in the training set since the baseline BLEU score is much better here (0.3685, see below). We observe the same insignificant effect of the *Tur* rules (0.3692). Addition of domain dependent n-grams appears to be far more important, i.e., a BLEU score increase of 10%. This effect is however not as clear as when the n-gram models are derived from the same data set (as in the Jotunheimen training data test in the previous pragraph), as one can expect.

- No rules based on the *Tur* data set, no domain n-grams: BLEU SCORE: 0.3685

- With rules based on the *Tur* data set, no domain n-grams: BLEU SCORE: 0.3692

- With rules based on the *Tur* data set, domain n-grams from the *Jotunheimen* training corpus: BLEU SCORE: 0.4127

One might wonder why the testing on unseen data gives better results than testing on training data. The answer is simply that the test data accidentally contains a higher proportion of one and two-word sequences than the training data, see Johannessen, Nordgård and Nygaard (in preparation) for details. To sum up thus far, domain n-gram adaptation appears to be very effective when OA is to be fine-tuned with respect to some limited domain.

3.4 Domain N-gram Boosting

When domain dependent n-grams are added to the training data, it is interesting to figure out the optimal "boost" level. In all the experiments hitherto the boost factor is 2000, which means that each domain n-gram is multiplied by 2000. No n-grams have been removed. In the table below various boost levels are used when the system has translated sentences from the *Jotunheimen* test set.

Boost: 1	BLEU SCORE: 0.3375
Boost: 10	BLEU SCORE: 0.3564
Boost: 100	BLEU SCORE: 0.3876
Boost: 1000	BLEU SCORE: 0.4086
Boost: 2000	BLEU SCORE: 0.4127
Boost: 4000	BLEU SCORE: 0.4145
Boost: 6000	BLEU SCORE: 0.4182
Boost: 8000	BLEU SCORE: 0.4181
Boost: 10000	BLEU SCORE: 0.4200
Boost: 14000	BLEU SCORE: 0.4200

These results clearly show that some form of boosting should be done. Curiously, a low boost gives worse results than no boost. For the present experiments it seems that something around 10,000 is the best boost factor we can get.

4 Conclusions

In this paper, we have outlined the architecture of *Oversettelsesassistenten*, which is a system based on well-known techniques (HMM-based POS-tagging, finite state transduction, HMM-based decoding applied to translations models) put together in a hybrid system which requires little training data in the form of aligned bilingual corpora. The system does require monolingual full form word lists and corpora for the target language, in addition to a standard bilingual dictionary which has to be synchronized to the monolingual word lists.

We have seen that rules specially designed for certain corpora can boost translation quality in terms of BLEU scores, but this approach does not generalize properly to other texts from the same domain. N-gram boosting has a very positive effect, however, and this approach generalize to certain limits, which probably varies from case to case. We believe that the OA approach can be used in MT system design when the relevant resources are available, and, crucially, when high quality aligned bilingual corpora are absent.

References

Henriksen, P., V. C. D. Haslerud og Ø. Eek. (2001). *Engelsk stor ordbok – Engelsk-norsk / Norsk-engelsk*. 1 edition, Kunnskapsforlaget, Oslo.

Ide, N. and J. Véronis. (1994). MULTEXT: Multilingual Text Tools and Corpora. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING 1994,* Kyoto, Japan.

Johannessen, J. B., T. Nordgård and L. Nygaard (in preparation). Evaluation of the LOGON demonstrator. Manuscript in preparation for LOGON monograph.

Oepen, S., H. Dyvik, J. T. Lønning, E. Velldal, D. Beermann, J. Carroll, D. Flickinger, L. Hellan, J. B. Johannessen, P. Meurer, T. Nordgård and V. Rosén. (2004). Som å kapp-ete med trollet? Towards MRS-based Norwegian-English machine translation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation*, Baltimore, MD.

Nordgård, T. (2000). NORKOMPLEKS - A Norwegian Computational Lexicon. In *Proceedings of COMLEX 2000. Workshop on Computational Lexicography and Multimedia Dictionaries*. Wire Communications Laboratory, University of Patras, Greece.

How Predictable is Finnish Morphology? An Experiment in Lexicon Construction

Aarne Ranta

Chalmers University of Technology and University of Gothenburg
Department of Computer Science and Engineering

1 Introduction

A Finnish noun has hundreds of forms, and a Finnish verb has thousands. The classification of the ways of forming them in *Nykysuomen Sanakirja* ("Dictionary of Contemporary Finnish", NSSK, Sadeniemi, 1961) gives 82 paradigms for nouns and 45 for verbs. At the same time, Finnish is not only complex but extremely regular: fluent speakers can identify the correct paradigm of a new word by just seeing one or two forms.

In this paper, we will first set up a framework of computational morphology using **smart paradigms**, heuristic functions that take just one or a few forms of a word as their arguments and infer the complete inflection table of the word from them. Then we present an implementation of Finnish smart paradigms, and relate them to traditional NSSK-style paradigms. We evaluate the access functions by counting the number of incorrect inferences in different kinds of word lists. From this, we can estimate the average number of forms needed to infer the correct inflection of a Finnish word. We also present a bootstrapping method for large-scale lexicon building, where the number of forms is increased in successive iterations.

2 Morphological Lexicon

A morphological lexicon is a list of words with all their inflectional forms. To take a simple example, an English morphological lexicon might have the following kinds of entries:

```
00123 N house houses house's houses'
00456 V house houses housed housed housing
10871 V write writes wrote written writing
```

Each entry has a unique identifier (here an integer), a part-of speech tag (here N or V), and a complete list of forms.

A morphological lexicon can be seen as the **canonical form** of the description of the words of a language. This means that it explicitly gives all the data that is needed to produce and recognize all forms and all words. It does *not* mean that anyone ever has to write the lexicon explicitly, or store it in its full form. To *write* a lexicon, an efficient way can be to use regular expressions (Beesley and Karttunen, 2003) or paradigm functions (Huet, 2005; Forsberg, 2007). To *store* a lexicon, finite-state transducers are a compact format, which avoids replicating common parts of words. Finally, also to *apply* a lexicon, finite-state transducers are adequate, for both analysis (finding the descriptions of a given form) and synthesis (finding the forms matching a given description).

Transducers are the essence of **finite-state morphology**, which moreover uses regular expressions as source code for the morphology. However, using transducers at runtime is independent of the source format: a morphological lexicon in canonical form, what ever way it is produced, can be easily compiled to a transducer. In the following, we will use the **word and paradigm model** (Hockett, 1954) when defining morphology. To add a word to a lexicon in this model, a word and a paradigm are given; the paradigm is a function that from the word produces a complete list of its forms, which can be added as an entry to a morphological lexicon and/or used for extending a transducer with more edges and states.

3 What Is a Paradigm?

Paradigm is not a mathematically precise notion like finite-state transducer is. We can, however, easily build a mathematical characterization by asking ourselves what a paradigm is supposed to do. We start with an **inflection table**: it is a function from form descriptions (tags of a given tag set) to actual forms (strings). A part of speech is morphologically defined in terms of what forms there are in its inflection table. For example, English nouns are inflected for both number and case, and a noun inflection table is thus a function of type

```
Number * Case -> Str
```

i.e., a function from the cartesian producs of numbers and cases to strings. The notation we use is from GF (Grammatical Framework, Ranta, 2004). English verbs have a different type,

```
VForm -> Str
```

where the verb form type `VForm` comprises possible verb forms: infinitive, third person singular present, past indicative, past participle, and present participle. If the verb *be* is included, some more forms are needed.

A **paradigm**, then, is a function that builds an inflection table from a given word form. Thus an English noun paradigm is a function of type

```
Str -> (Number * Case -> Str)
```

For instance, the regular noun paradigm

```
regNoun : Str -> (Number * Case -> Str)
```

could be defined as the function that adds the endings Ø, *s*, *'s*, and *s'* to a given nominative singular form to produce all the four forms. The form serving as the argument of a paradigm coincides in this case with the **stem**, where the stem is defined as an immutable prefix to which endings are glued.

So, how many noun paradigms are there in English? If we require the arguments of paradigms to be immutable stems, we end up with a high number of paradigms, including

- fully regular nouns, such as *house*, where the stem is the nominative singular form
- nouns ending with a *y* (preceded by a consonant), such as *fly*, where the stem is *fl*
- some nouns ending with *ouse*, such as *mouse*, where the stem is *m*

This definition of paradigms was consistently carried out in Swedish by Hellberg (1978), who ended up with hundreds of paradigms. He called the immutable prefix the **technical stem**. In traditional linguistics, the notion of a stem is more abstract. Paradigms may permit **stem alternations**, so that e.g., English *fly* is considered regular with the alternation of *y* to *ie* before the plural *s*. Paradigms may also permit multiple stems: the verb form series *drink, drinks, drank, drunk, drinking* is generated from the three stems *drink, drank, drunk* rather than from the technical stem *dr*. Yet another way to see this case is in terms of a stem *drVnk* with a place holder *V* for a vowel, which undergoes the thematic vowel alternation *i-a-u*.

What is the most intuitive way of presenting paradigms, if they are to be used by lexicographers who are not linguists? We can take a cue from dictionaries. The technique of paradigm identifiers (usually numbers) is sometimes used. Thus *Le Petit Robert* of French has

```
concevoir v.tr. <28>
```

An alternative would be to give a few forms that identify the paradigm:

```
concevoir v.tr. conçois, concevons, conçoive,
  concevrai, conçu
```

This idea is related to the technique of multiple stems in linguistics, with the difference that the stems need not really be independent of each other; the idea of a thematic series is just that the series gives enough information to determine all forms of the word.

4 Smart Paradigms

Thematic form lists are often used with a variable number of arguments: for instance, if a verb is completely regular, only one form is given. We will call a thematic list with just the necessary number of terms a **smart paradigm**. We have found smart paradigms to be an intuitive and efficient way of building lexica in practice. The smart paradigm is a set of functions that take one or more arguments, and the user (who is able to recognize current inflection even if not predict it) can start with one argument, see what it generates, and give more forms if necessary.

The programmer's variant of a smart paradigm is a set of **construction functions**—functions with an increasing number of argument lists and an inflection table as value. Given that we define the type `V` as the type of inflection tables with all verb forms in English, we can present the verb construction function as the following group of functions:

```
mkV_1 : (talk : Str) -> V
mkV_2 : (omit,omitted : Str) -> V
mkV_3 : (drink,drank,drunk : Str) -> V
mkV_5 : (go,goes,went,gone,going : Str) -> V
```

The identifiers `talk`, `omit`, etc., are variables, which all have the type string. Their names suggest to the user what form is expected for each variable, which is a part of the documentation of the paradigm. For GF itself, they make no semantic difference. The type of `mkV_2`, for instance, is just the same as

```
Str -> Str -> V
```

i.e., a function that takes two strings to a verb.

An alternative representation of the above set of construction functions is a web form with five slots. Depending on how many slots are filled, some of the `mkV` functions is applied, and the full inflection table is displayed to the user. If she is satisfied with the result, she can push the "Save" button to extend the lexicon; otherwise, she can add more forms until the table is correct.

The GF Resource Grammar Library (Ranta, 2008) is an implementation of the morphology and a part of the syntax of 13 languages in the GF grammar formalism (Ranta, 2004). The morphology implementations provide smart paradigms for 9 of the languages included: Danish, English, Finnish, French, German, Italian, Norwegian, Spanish, and Swedish (Arabic, Catalan, Russian, and Urdu do not yet have smart paradigms at the moment of writing). The presentation of the paradigms looks exactly as above. Effort has been spent to make the paradigms smart also in the sense of requiring a minimum of forms. For instance, the one-place noun paradigm for English inspects the singular form of the word to select the proper plural form in a wide variety of cases.

```
mkN_1 : (word : Str) -> N = \word ->
```

```
let
  words : Str = case word of {
    _ + ("a"|"e"|"i"|"o"|"y") + "o" => word + "s" ;
    _ + ("s"|"x"|"sh"|"o")          => word + "es" ;
    _ + ("a"|"e"|"o"|"u") + "y"     => word + "s" ;
    x + "y"                         => x  + "ies" ;
    _                               => word + "s"
    }
in
mkN_2 word words
```

The function uses regular expressions to inspect the suffix of the word. The plural created is sent, together with the singular, to the 2-argument `mkN` function. Actual use of the 2-argument variant is needed only for truly irregular nouns, such as *mouse* or *man*.

A few words on GF notation might be in place. The above piece of code defines a function called `mkN_1`. The definition consists of a type and a defining expression. The type here is a function from `Str` to `N`. The definition is a lambda term binding the variable `word`. For this variable, `mkN_1` returns the value of calling `mkN_2` with two arguments: `word` itself and `words`, which is defined locally in a "let" expression. The definition of the local constant `words` makes a case analysis of the string bound to the variable `word`. The cases are tried in the order in which they appear. The first case splits the string into three parts: any prefix (matched by the wild card _), followed by any of the strings *a*, *e*, *i*, *o*, *y*, terminated by the string *o*. This case matches strings such as *bamboo* and *embryo*, which get the plural ending *s* attached to the stem. Other words ending with an *o* are passed to the next case, which uses the same ending *es* as it also assigns to words ending with *s*-like sounds. Words ending with a *y* are treated next. The last case uses a catch-all pattern, giving a default *s* plural to all words that were not matched before.

5 The Inflection of Nouns in Finnish

We shall now finally enter Finnish morphology. We go through the inflection of nouns in more detail than verbs: even though verbs have more forms, they are more uniform and regular: NSSK has 82 paradigms for nouns and 45 for verbs.

A noun form has several components: stem, case, number, possessive, and optional particles. Here is an example:

```
ves  +   i +     ssä +     ni +              kin
"water"  Plural  Inessive  Possessive Sg_1  "also"
"also in my waters"
```

With all possible combinations of the components, the number of different forms can be estimated to be around 1,500. In practice, grammar books only

-	singular	plural	meaning
nominative	***vesi***	*vedet*	"water(s)"
genitive	***veden***	***vesien***	"of water(s)"
partitive	***vettä***	***vesiä***	"portion of water(s)"
essive	***vetenä***	***vesinä***	"as water(s)"
translative	*vedeksi*	*vesiksi*	"to as water(s)"
inessive	*vedessä*	***vesissä***	"in water(s)"
elative	*vedestä*	*vesistä*	"from in water(s)"
illative	***veteen***	***vesiin***	"to in water(s)"
adessive	*vedellä*	*vesillä*	"on water(s)"
ablative	*vedeltä*	*vesiltä*	"from on water(s)"
allative	*vedelle*	*vesille*	"to on water(s)"
abessive	*vedettä*	*vesittä*	"without water(s)"
comitative	-	*vesine*	"with water(s)"
instructive	-	*vesin*	"by means of water(s)"

Figure 1: The inflection table of a Finnish noun.

need to consider around 30 forms, since the possessives and particles are (almost) purely agglutinative. The 2 numbers and 14 cases specify the core of the paradigms, where the case and the number often form portmanteau fusions, and the stem may change in function of the ending. In two of the cases, no number distinction is made, and we thus end up with 26 noun forms, as shown in the inflection table in figure 1.

Many of the 26 noun forms can be obtained from each other by just changing an ending. For instance, five of the singular local cases (inessive, elative, adessive, ablative, abessive) form a group where just the ending needs to be interchanged. NSSK uses 8 forms as determining the others. In the GF implementation, we have used 10 arguments in the worst-case noun constructor. These 10 forms are guaranteed to cover all stem and vowel alternations that a Finnish noun can undergo, so that all the 26 paradigm forms (and thereby the 1,500 full forms) can be generated by deterministic concatenative procedures. These ten forms are printed in boldface in figure 1.

As partly shown in the inflection table of *vesi*, Finnish words are subject to the following kind of alternations:

- **Vowel harmony**: the *ä* in endings is realized as *a*, if the stem contains one of the letters *a*, *o*, *u*.

- **Consonant gradation**: many consonants and consonant clusters have a **strong** and a **weak** grade, which are selected as a function of the ending; in the *vesi* example, *d* is the weak grade of *t*.

- **Stem vowel alternation** the last vowel of the stem varies as a function of the ending; in the *vesi* example, *i* alternates with *e*.

In the NSSK paradigm set, vowel harmony and consonant gradation are treated separately from the paradigms. Thus, for instance, "Paradigm 1" (*valo*, "light") has at least 30 variants, if the effects of vowel harmony and consonant gradation are built in to reach the "technical stem" notion of a paradigm. Thus the 82 noun paradigms of NSSK involve a certain amount of abstraction, and their user therefore has to know how to apply the vowel harmony and consonant gradation rules.

The sheer number 82, together with the complex sound alternations required, makes the NSSK paradigm system tedious and error-prone to use. In GF, we have relieved this burden by using smart paradigms, where the user gives a number of characteristic forms and the program identifies the paradigm to which the word belongs. The following five constructors are currently used:

```
mkN_1  : (talo : Str) -> N
mkN_2  : (talo,taloja : Str) -> N
mkN_3  : (talo,taloja,talon : Str) -> N
mkN_4  : (talo,taloja,talon,taloa : Str) -> N
mkN_10 : (talo,talon,taloa,talona,taloon,talojen,
       taloja,taloina,taloissa,taloihin : Str) -> N
```

The worst-case variant uses 10 forms, which give full certainty. The shorter variants use 1, 2, 3, or 4 forms, with increasing certainty. In practice, the 10-form variant is needed only for a small number of words, which either belong to obsolete paradigms or have become irregular via "wear" in use.

In addition to the above paradigms, there are two more used for the formation of compound nouns:

```
mkN : (pika : Str) -> (juna  -> N) -> N
mkN : (oma  : N)   -> (tunto -> N) -> N
```

The first case covers the most common type of compounds, where an immutable prefix is added to an inflectible noun. In the second type, both parts of the compound are inflected in agreement. In the selection of a paradigm, it is crucial to know what the last part of a compound is, in order to get the vowel harmony right and, in some cases, to decide how many syllables the word has.

Given the above system of paradigms—which will be explained in more detail in later sections—one can build a user-friendly lexicon construction system as follows. First of all, an **irregularity lexicon** is created to make the use of the 10-place constructor unnecessary. When encountering a new word, we first check if it is included in the irregularity lexicon. If not, we try to define it with the 1-place noun constructor. If this fails (in human inspection), we give yet another form, and so on. This defines a smooth process in which, with very little training, a lexicographer can treat hundreds of nouns in an hour.

The interesting question is now: how smooth is the procedure of adding characteristic forms—how many forms are needed in average? In other words: how predictable is Finnish inflection? We shall now go on to present an experiment that gives us some figures. After that, we will take a look at the definitions of the constructors and the paradigms underlying them; this inspection will both explain the figures and give a hint on what kinds of words will typically need more forms to inflect correctly.

6 A Noun Prediction Experiment

The first experiment was carried out with written Finnish from four different genres, treating each genre in the same way. Four samples of equal size were gathered, from the following sources:

- *Aino*, a children's book by Kristiina Louhi (1989)
- *Duodecim*, a professional article within clinical medicine (2007)
- *Swadesh*, the list of the 207 "most primitive words" proposed by Swadesh (1955)
- *Dictionary*, a medium-size Finnish-English dictionary (1971)

From each source, a random sample of 99 distinct nouns were collected. In *Aino* and *Duodecim*, we took the first nouns occurring in the text. In *Swadesh*, we took all nouns and enough many random adjectives to reach the desired number (adjectives in Finnish are inflected in the same way as nouns). In *Dictionary*, we took the first noun of every fifth page. In all cases, compound nouns were eliminated by only considering their last parts; it is impossible to predict the inflection of a compound without identifying its last part, and this is not a mechanical task in general.

The random sample of nouns was presented as a list of their nominative singular forms:

```
Aino
tunti
äiti
tähti
puhelin
kuuloke
```

From this list we create a **gold standard** listing all the 10 thematic forms of each noun:

```
Aino Ainon Ainoa Ainona Ainoon
  Ainojen Ainoja Ainoina Ainoissa Ainoihin
```

```
tunti tunnin tuntia tuntina tuntiin
  tuntien tunteja tunteina tunneissa tunteihin
äiti äidin äitiä äitinä äitiin
  äitien äitejä äiteinä äideissä äiteihin
tähti tähden tähteä tähtenä tähteen
  tähtien tähtiä tähtinä tähdissä tähtiin
puhelin puhelimen puhelinta puhelimena puhelimeen
  puhelimien puhelimia puhelimina puhelimissa
  puhelimiin
kuuloke kuulokkeen kuuloketta kuulokkeena
  kuulokkeeseen kuulokkeiden kuulokkeita kuulokkeina
  kuulokkeissa kuulokkeisiin
```

The gold standard was created using the bootstrapping method of section 9, manually verified.

In the experiment, the outcomes of the 1-place, 2-place, 3-place, and 4-place noun constructors were compared with the gold standard. From each comparison, the number of nouns whose inflection is incorrect (w.r.t. the gold standard) was counted. The constructors are designed to be **monotonic**, so that giving more arguments never destroys a correct inflection. This means that the number of errors decreased as the number of arguments increased.

Different kinds of material were expected to give different degrees predictability. We expected *Duodecim* to contain mostly new and technical words, which would thereby be regular; this would be a comforting result concerning the main purpose of GF, which is to build translation and generation systems for technical domains. We expected *Swadesh* and *Aino* to contain many old, worn, irregular words, and *Dictionary* to give a reasonable average. The results of the experiment, covered in the following table, show that our expectations were only partly satisfied:

args	***Aino***	***Duodecim***	***Swadesh***	***Dictionary***
1	8	16	31	19
2	1	6	15	4
3	0	3	7	2
4	0	1	2	1

The slightly surprising outcome was that *Aino* was the easiest material to predict. *Duodecim* is close to *Dictionary*, whereas *Swadesh* is rich in the words of type *vesi-veden* (“water”), which is an ancient paradigm no longer productive. The altogether 3 unpredicted words all represent irregular paradigms intentionally left outside the heuristics: *kevät* (“spring”), *mies* (“man”), *sydän* (“heart”).

As a preliminary conclusion for general text we could thus say:

- For 80% of nouns, the inflection is correctly inferred from just one form (the nominative singular).

- For 90% of words, it is enough to have one more form (the partitive plural).
- Adding the genitive and partitive singular gets all nouns right, except for a fixed set of nouns that can be given in advance.

7 A Large-Scale Noun Prediction Experiment

The noun prediction experiment with different genres completed, we got hold of a freely available electronic word list (*KOTUS*, Kotimaisten Kielten Tutkimuskeskus, 2006) named after its provider *Kotimaisten kielten tutkimuskeskus* ("Research Centre for Domestic Languages"). KOTUS has 50 noun paradigms, which are used for annotating the lemmas in the word list. We implemented these paradigms using the 20 GF paradigms of section 10, with recourse to the worst-case noun constructor for some rare KOTUS paradigms. Then we took the word list and removed most compounds (those clearly enough marked) and *plurale tantum* words (which our paradigms could not directly handle). We ended up with a list of 27,680 nouns, and used the paradigm annotations to expand this to a gold standard and repeated the experiment.

The outcome of this second experiment largely confirmed the earlier results:

args	***KOTUS #***	***KOTUS %***
1	4993	18.0
2	1062	3.8
3	792	2.9
4	789	2.9

The insignificant drop between 3 and 4 suggests that the singular partitive should rather be treated in the irregularity lexicon.

We also tested an alternative order of forms, using the genitive singular rather than the partitive plural as the second form. In the small experiments above, this order gave slightly better results, especially for the Swadesh corpus. In KOTUS, however, the error rate with two forms increased to 3597, that is, 13.0%. This result confirmed that the partitive plural is more useful than the genitive singular in predicting inflection. The *a priori* reasons originally leading to this choice are given below in section 11.

8 Predictability as the Average Number of Required Forms

Now, how predictable is Finnish morphology? One possible answer is given by calculating the **average number of forms** needed to identify the inflection of a Finnish noun in the KOTUS list. Assuming that all words we fail to predict

with 3 arguments need 10 forms, we get

$$(792*10+(1062-792)*3+(4993-1062)*2+(27680-4993))/27680 = 1.42$$

forms in average. Assuming that those words can be found in the irregularity lexicon and hence only need one form, we get 1.16 forms in average. Which of these figures should be used?

When looking at the nouns in KOTUS that we failed to predict, our earlier expectations where partly falsified. Irregular "old" words like *kevät* and *mies* are of course present, but the majority are *new* words whose orthography has not been naturalized to match the pronunciation. A typical example is the word *brie* ("brie cheese"), which has the French pronunciation [bri:], and it is hence inflected like *pii* ("silicon"). But the orthography can mislead a smart paradigm to inflect it like *tie* ("road").

For a Finnish speaker, inflecting new loan words is to a large extent dependent on education. For instance, KOTUS follows the correct French pronunciation of the words *calvados* [kalvados] and *tournedos* [turnedo:] when selecting the Finnish paradigm—but this is hardly a part of the common knowledge of native Finnish speakers.

As a conclusion, loan words like *brie*, *calvados*, and *tournedos* with their unpredictable pronunciation are a far more significant source of unpredictable inflection than the "old" words in a static irregularity lexicon. To help future lexicographers, the smart paradigms of the GF Resource Grammar Library should be improved so that the full set of 10 forms are not needed as often as now. But at this point, a prudent conclusion is to say that Finnish nouns need 1.42 forms in average.

9 A Bootstrapping Method for Lexicon

The creation of the gold standard in section 6 required manual work. Its amount was gradually reduced in the course of the project, as we found a more economical way of using the smart paradigms and GF. This way suggests a general method of bootstrapping a lexicon, which would be usable for large word lists as well, and independently of language—even though its efficiency depends on how predictable the morphology of the language is.

The starting point in bootstrapping is a list of words in the form required by the one-argument smart paradigm. We will use Finnish nouns as example here, with the nominative singular as the first form:

```
meri
sade
nainen
kivi
rivi
```

```
tohtori
apina
kulkija
kukka
auto
rakkaus
```

To each word in this list, we apply the 1-place noun constructor, just to produce one more form, the partitive plural:

```
meri    (merejä    >> meriä)
sade    sateita
nainen  naisia
kivi    (kivejä    >> kiviä)
rivi    rivejä
tohtori (tohtoreja >> tohtoreita)
apina   (apinoja   >> apinoita)
kulkija (kulkijia  >> kulkijoita)
kukka   kukkia
auto    autoja
rakkaus rakkauksia
```

For reasons explained in section 11, it is enough to pay attention to those nouns that end with an *i*, as well as 3-syllabic nouns ending with an *a* or *ä*, to produce a "2-form gold standard". In the above list, five words are manually changed.

The 2-form gold standard is processed with the 2-place noun constructor, to produce a 3-form list; now, the genitive singular is added. In this case, we mostly have to change some 2-syllabic words that don't have expected consonant gradation, as well as nouns ending with *us* but inflected like *rakkaus* ("love") rather than *pakkaus* ("package").

```
auto    autoja     (audon >> auton)
rakkaus rakkauksia (rakkauksen >> rakkauden)
```

The 3-argument noun constructor adds one more form, the partitive singular. Some words may have to be changed, typically for words in the *i-e* paradigm:

```
meri    meriä      meren (mertä >> merta)
```

Actually, the number of words with a deviant partitive singular is limited, and the last step can hence be avoided by extending the irregular lexicon with these words—this is what we assume in the following estimate of work that is needed.

The amount of work needed for producing a morphological lexicon can be defined as the number of words that need to be written or read by the lexicographer: to build a lexicon with 100 lemmas, she has, in average, to

- check 30 partitive plural forms
- change 15 partitive plural forms
- check 50 genitive singular forms
- change 5 genitive singular forms
- change the whole inflection of 2 words (18 forms)
- altogether, read 80 forms and change 38 of these

Processing 100 words in GF to create the full set of forms (or any subset of it) takes 0.4 seconds, so that with a suitable script (available from GF homepage) the processing part takes around 2 seconds. If changing a word form takes 20 seconds and just reading it 5 seconds, the work spent on 100 lemmas is around 16 minutes. Thus turning a list of 3,000 noun lemmas (outside the irregularity lexicon) into a complete morphological lexicon would take one effective working day, plus less than 2 minutes of GF processing time.

Another estimate can be calculated from the average 1.42 forms needed to identify the inflection of a Finnish noun (section 8 above). If 0.42 forms per lemma have to be produced and one production takes 20 seconds, then 100 lemmas require 14 minutes. With this count, one working day is enough for producing a lexicon of 3,480 words.

10 Basic Paradigms for Finnish Nouns

So far we have not gone into the details of Finnish inflection, but just presented an experiment and a method, which would be applicable to other languages as well. We will now give an outline of the inflection paradigms that we used.

The system of paradigms works in two layers:

- a user-accessible smart paradigm set `mkN`
- an underlying basic paradigm set close to linguistic description

The undelying basic paradigm set is reminiscent of the NSSK and KOTUS sets, only fewer (20 instead of 82 or 50). The lower number is due to three factors:

- higher level of abstraction
- the use of multiple arguments
- the omission of rare and obsolete paradigms (which are treated lexically in the irregularity lexicon)

Here are the paradigms, displayed as GF type signatures:

```
dLujuus  : (lujuus : Str) -> N
dNainen  : (nainen : Str) -> N
dPaluu   : (paluu : Str) -> N
dPuu     : (puu : Str) -> N
dSuo     : (suo : Str) -> N
dKorkea  : (korkea : Str) -> N
dKaunis  : (kaunis : Str) -> N
dLiitin  : (liitin : Str) -> N
dOnneton : (onneton : Str) -> N
dUkko    : (ukko,ukon : Str) -> N
dSilakka : (silakka,silakan,silakoita : Str) -> N
dArpi    : (arpi,arven : Str) -> N
dRae     : (rae,rakeen : Str) -> N
dPaatti  : (paatti,paatin : Str) -> N
dTohtori : (tohtori : Str) -> N
dPiennar : (piennar,pientaren : Str) -> N
dNukke   : (nukke,nuken : Str) -> N
dJalas   : (jalas : Str) -> N
dUnix    : (unix : Str) -> N
dSDP     : (SDP : Str) -> N
```

The name of a paradigm begins with a `d` (for "declension"), followed by an example word belonging to that paradigm. Some paradigms have two arguments, mainly to control consonant gradation. It is usually possible to produce the weak grade automatically from the strong grade and vice-versa, but whether a word undergoes this gradation is a lexical property of the word. For instance, *outo* ("strange") and *auto* ("car") are both inflected by `dUkko`, but *outo* has the weak grade in the genitive *oudon* (and other forms demanding the weak grade), whereas *auto* preserves the strong grade in *auton* and all other forms. Similar examples can be found for all 2-argument paradigms, which was what motivated the use of a second argument (an alternative would have been to have two different paradigms).

The most complex paradigm is `dSilakka`, of polysyllabic words ending with *a,ä,o,ö*. The rules for forming the partitive plural (the third argument) are so complex and uncertain, that we decided to make the choice lexical (see however Karlsson (1977), § 16, for a summary of the rules).

The last noun paradigm, `dSDP`, deals with acronyms recognized by a colon between the stem and the ending: *SDP:n, SDP:tä, SDP:hen*. The correct ending depends on the pronunciation of the last letter as the name of a letter of the Finnish alphabet. There are 9 different sets of endings that can result.

11 Smart Paradigms for Finnish Nouns

The high-level smart paradigms (the noun constructors `mkN`) analyse their arguments and dispatch to some of the underlying paradigms. Here is a slightly simplified but fully functional version of the 1-argument constructor in GF code, covering the most productive suffixes that help to select a paradigm.

```
mkN_1 : Str -> N = \ukko ->
  let
    ukon = weakGrade ukko + "n" ;
  in
  case ukko of {
    _ + "nen" => dNainen ukko ;
    _ + ("aa"|"ee"|"ii"|"oo"|"uu"|"yy"|"ää"|"öö") =>
      dPuu ukko ;
    _ + ("ie"|"uo"|"yö") => dSuo ukko ;
    _ + ("ton"|"tön") => dOnneton ukko ;
    _ + "e" => dRae ukko (strongGrade ukko + "en") ;
    _ + ("a"|"o"|"u"|"y"|"ä"|"ö") => dUkko ukko ukon ;
    _ + "i" => dPaatti ukko ukon ;
    _ => dUnix ukko
  } ;
```

The function uses the auxiliary functions `strongGrade` and `weakGrade`, which implement consonant gradation. Thus consonant gradation is treated separately from the paradigms. The same concerns vowel harmony, which is encapsulated in a low-level auxiliary function not even visible in the code examples of this paper.

The main cases where `mkN_1` makes an uncertain guess are the following:

- all nouns ending with a short *aouyää* are sent to `dUkko` with consonant gradation
- all nouns ending with a short *i* are sent to `dPaatti` with consonant gradation
- all nouns ending with a short *e* are sent to `dRae` with consonant gradation

Except for consonant gradation, the errors produced by these guesses can be corrected by adding the partitive plural form. Hence the 2-argument constructor, again slightly simplified:

```
mkN_2 : (ukko,ukkoja : Str) -> N =
  \ukko,ukkoja ->
  let
    ukon = weakGrade ukko + "n"
  in case <ukko,ukkoja> of {
```

```
<_ + "i", _ + ("ia" | "iä")> =>
  dArpi ukko (init (init ukon) + "en") ;
<_ + "i", _ + ("eita" | "eitä")> =>
  dTohtori ukko ;
<_ + ("a"|"ä"|"o"|"ö"), _ + ("a"|"ä")> =>
  dSilakka ukko ukon ukkoja ;
<_ + "e", _ + ("eja" | "ejä")> =>
  dNukke ukko ukon ;
<_, _ + ("a" | "ä")> => mkN_2 ukko ukon ;
_ => Predef.error
  (["last arg should end 'a/ä', not"] ++ ukkoja)
} ;
```

The 1-argument constructor is used as a catch-all case, but it still requires the second argument to end with an *a* or *ä*, so that it makes sense as a partitive plural form. This guards against mistaken use of the paradigm with some other forms.

12 Verbs

In many languages, verb morphology is more complex than noun morphology, because verb inflection contains noun inflection as its proper part—the participle forms. Finnish is no exception, and there are actually several kinds of participles and other "nominal forms". At the same time, verbs are morphologically simpler than nouns, since there are fewer paradigms. NSSK has 45 verb paradigms. The GF Resource Grammar Library has 8 "deep" paradigms, and only the 1- and 2-argument smart constructors are exposed to lexicographers, in addition to a worst-case 12-argument function.

We made an experiment similar to the one with nouns on the verb variants of the 99-word *Swadesh* and *Dictionary* corpora. Here are the numbers of erroneous predictions:

args	***Swadesh***	***Dictionary***
1	10	1
2	3	1

The unresolved verb in *Dictionary* is *nähdä* ("see"). It is also one of the 3 in *Swadesh*, the other ones being *seistä* ("stand") and *pyyhkiä* ("wipe"). These first two are deviant and belong to the irregularity lexicon, whereas *pyyhkiä* would require one more form to detect consonant gradation.

The 10 verbs that come out incorrectly with `mkV_1` in *Swadesh* are ones with lexically defined consonant gradation, e.g., *pelätä* ("fear"). Its first person singular present indicative has a strong grade consonant (*pelkään*), but there are similar verbs that do not undergo consonant gradation when working out from the infinitive form: *palata - palaan*.

The conclusion for verbs is even more encouraging than for nouns: more that 90% can be predicted from one form, and the rest from two forms (except for verbs in the irregularity lexicon). The problem of loan words with deviating pronunciation is largely avoided, because verbs, unlike nouns, cannot be loaned without attaching suffixes that clearly identify the paradigm: *chattailla* ("to chat on the internet"), *mailata* ("to send an email"). Selecting the suffix can however still be a problem for derivational morphology.

13 Related Work

Finnish morphology has played an important role in the development of computational morphology in general, due to its high complexity combined with a high grade of regularity. Pioneering work was carried out in the late 1970's by Sågvall Hein (1978, 1980), Koskenniemi (1980), and Brodda and Karlsson (1978). Koskenniemi (1983) is generally seen as a landmark, not only because it scaled up to a large coverage analysis and synthesis, but also because it introduced a new finite-state method known as **two-level morphology**, which was later applied to a wide variety of languages. We know only few computational approaches using paradigms rather than morphophonemic processes; Koskenniemi (1980) and Carlson (2005) are examples of ones.

As for lexicon construction, the method presented above is related to the lexicon extraction method of Forsberg et al. (2006). The difference is that, in that work, sets of characteristic forms are searched from a corpus of words with unknown parts of speech, whereas in the present approach, the forms are produced by just analysing one given form with known part of speech. More remotely related work includes the influential algorithm for unsupervised learning of morphology of Goldsmith (2006), in particular a further development applied to Finnish by Creutz and Lagus (2007). In these approaches, paradigms are not given in advance: they are the very goal of the search.

14 Conclusion

In the GF Resource Grammar Library, we have built a complete system of inflectional paradigms for nouns and verbs of Finnish (as well as 12 other languages). Upon that system, we have implemented a set of smart paradigms, which can be used for inferring the complete paradigm from a small set of forms. An experiment with Finnish nouns showed that less than 1.5 forms, in average, are needed to infer the correct inflection, with a total number of forms between 26 and 1,500, depending on how forms are counted.

From the experiment, we have derived a general method of bootstrapping a morphological lexicon, based on iterated addition of forms. This method can give an efficient way to create resources for new languages. For the particular case of Finnish, much of this need has recently been catered for by the

release of the KOTUS word list with paradigm annotations under the GNU LGPL license (the same as the GF Resource Grammar Library uses). Hence, in Finnish, our method is needed only for dealing with words not included in this word list.

References

Beesley, K. and L. Karttunen (2003). *Finite State Morphology*. CSLI Publications.

Brodda, B. and F. Karlsson (1978). An Experiment with Automatic Morphological Analysis of Finnish. Papers from the Institute of Linguistics, University of Stockholm, 40.

Carlson, L. (2005). Inducing a Morphological Transducer from Inflectional Paradigms. In *Inquiries into Words, Constraints and Contexts. Festschrift for Kimmo Koskenniemi on his 60th Birthday*. CSLI.

Creutz, M. and K. Lagus (2007). Unsupervised Models for Morpheme Segmentation and Morphology Learning. *ACM Trans. Speech Lang. Process. 4*(1), 3.

Forsberg, M. (2007). *Three Tools for Language Processing: BNF Converter, Functional Morphology, and Extract*. Ph. D. thesis, Dept. of Computer Science and Engineering, Chalmers University of Technology and Gothenburg University.

Forsberg, M., H. Hammarström, and A. Ranta (2006). Morphological Lexicon Extraction from Raw Text Data. In T. Salakoski (Ed.), *FinTAL 2006*, Volume 4139 of *LNCS/LNAI*.

Goldsmith, J. (2006). An Algorithm for the Unsupervised Learning of Morphology. *Nat. Lang. Eng. 12*(4), 353–371.

Hellberg, S. (1978). *The Morphology of Present-Day Swedish*. Almqvist & Wiksell.

Hockett, C. F. (1954). Two models of grammatical description. *Word 10*, 210–233.

Huet, G. (2005). A Functional Toolkit for Morphological and Phonological Processing, Application to a Sanskrit Tagger. *The Journal of Functional Programming 15(4)*, 573–614.

Karlsson, F. (1977). *Finsk grammatik*. Suomalaisen Kirjallisuuden Seura.

Koskenniemi, K. (1980). On automatic lemmatization of finnish. *Fenno-Ugrica Suecana 3*, 27–44.

Koskenniemi, K. (1983). *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*. Ph. D. thesis, University of Helsinki.

Kotimaisten Kielten Tutkimuskeskus (2006). KOTUS Wordlist. `kaino.kotus.fi/sanat/nykysuomi`.

Ranta, A. (2004). Grammatical Framework: A Type-Theoretical Grammar Formalism. *The Journal of Functional Programming 14(2)*, 145–189.

Ranta, A. (2008). Grammatical Framework Homepage. `digitalgrammars.com/gf`.

Sågvall Hein, A. (1978). Finnish Morphological Analysis in the Reversible Grammar System. In *COLING 78, Information Abstracts*.

Sågvall Hein, A. (1980). An outline of a computer model of finnish word recognition. *Fenno-Ugrica Suecana 3*, 7–26.

Sadeniemi, M. (Ed.) (1961). *Nykysuomen sanakirja*. WSOY.

Swadesh, M. (1955). Towards Greater Accuracy in Lexicostatistic Dating. *International Journal of American Linguistics 21*, 121–137.

Corpora in Grammar Learning Evaluation of ITG

Anju Saxena
Uppsala University
Department of Linguistics
and Philosophy

Mikaëla Lind
Katarinaskolan
Uppsala

1 Using Corpora in Teaching

There has been a growing interest in using natural language corpora in teaching and in research, partly due to the growing availability of computer-readable linguistic corpora, and partly due to an increasing interest in examining language in its natural context as opposed to investigating constructed language examples in isolation. Researchers, teachers and students now have access to different types of language corpora to discover facts about language, for example about word frequency distributions in a language or a language-type, in which context words tend to occur and which grammatical patterns are associated with a particular linguistic item (Ghadessy et al 2000). There have been two primary approaches to the use of corpora in language teaching/learning:

(1) the "COBUILD approach" (or the indirect approach)
(2) the Data-Driven Learning approach (or the direct approach)

Until recently, the COBUILD approach was the predominant approach. Corpora, in this approach, are used by researchers and producers in building dictionaries and other language learning materials. Traditionally it has been very large corpora which have been used for this purpose. Further, within this approach, the user (a student, for example) receives results of a project involving corpora as end-products (for example, in the form of a language learning package). Learners do not get to use the corpus themselves in order to come up with their own analyses and learn from that.

In the Data-Driven Learning approach students use corpora directly in their own learning. They use the corpus, for example, to discover linguistic patterns and to organize the linguistic patterns that they observe, arriving at generalizations inductively and verifying deductive rules. Such exposure to corpora provides students with the opportunity not only to extract relevant examples of various linguistic structures, but also provides them with material for discussion when they find gaps, to verify and extend their hypotheses and to arrive at generalizations. In favor of the Data-Driven Learning approach, Tim Johns (1991) states:

> What distinguishes the data-driven learning approach is the attempt to cut out the middleman … and give direct access to the data so that the learner can take part in building up his or her own profiles of meanings and uses. (Johns 1991, as quoted in Aston 1997)

Johns (1991) divides Data-Driven Learning into three phases:

(i) observation
(ii) classification
(iii) generalization

One advantage of using corpora in teaching is that instead of learning about linguistic theories *in vacuo* (a more passive learning method, where facts are fed to students in the form of lectures and ready-made examples), students have a chance to test these theories themselves against linguistic corpora and thereby learn about these theories or concepts for themselves (a more active learning method). When corpora are used by students as part of their learning, the distinction between learning and doing research becomes "blurred", as the students, by a discovery procedure (thus, research), learn things for themselves (Knowles 1990). The use of corpora in teaching can, in this way, affect both teachers' as well as students' roles. This approach is as equally relevant in a classroom set-up as in self-study situations.

The corpora used for teaching purposes may be large or small. The purpose here is *exploration* and not to arrive at a water-tight description of the phenomenon under consideration. It is possible that if small corpora are used, the results may not be exactly the same as if the corpora were large, but this difference does not detract from the advantages of using corpora in teaching.

The gap between the COBUILD and DDL approaches is, however, getting smaller. More access to corpora (especially for non-commercial purposes) provides better conditions for their use in producing language learning tools as well as in using them directly in teaching/learning. The aim of this paper is to describe our experience in using corpora in grammar teaching/learning in two Linguistics courses offered by the Department of Linguistics and Philology at Uppsala University. The courses are *Lingvistik I* (Linguistics I) and *Lingvistik II* (Linguistics II). We will begin by describing the web-based platform – ITG: IT-based collaborative learning in grammar – in section 2, followed in section 3 by an account of our experience (including student feedback) with using ITG in these two regular courses at our university.[1]

[1] Mikaëla Lind was responsible for the computer lab sessions where ITG was used in the two courses discussed here. Sections 1 and 2 were written by Saxena and section 3 was written jointly by Saxena and Lind.

2 ITG – Collaborative Learning in Grammar

ITG is a web-based platform <http://spraakbanken.gu.se/itg/itg/itg.jnlp>. Its development began in the project "IT-based Collaborative Learning in Grammar" which started January 1, 2002 with financial support from Distum/Nätuniversitet in Sweden with Anju Saxena as the Principal Investigator and the Department of Linguistics at Uppsala University as its host. Several departments at Uppsala University as well as the Department of Linguistics at Stockholm University and the Department of Linguistics at the University of Gothenburg (and later on also Språkbanken, the Department of Swedish Language at Gothenburg) were the collaborating partners in this project. ITG is now maintained and developed further by Språkbanken. The main aims of the project were:

- To use annotated corpora of Swedish and other languages as the basis for learning grammatical patterns;
- To develop web-supported collaborative learning in grammar where corpora of natural language material form the basis for group activities;
- To use the web-supported collaborative method in regular courses in grammar at collaborating Departments of Linguistics;
- To show the wider applicability of the platform and tools developed in this project, for example by applying them to additional languages and additional exercise types, using ITG in other contexts;
- A significant consideration during all the stages of this project has been that the technical aspects of using or working with the web-based system should not increase the workload of students or faculty.

ITG has a modular architecture, composed of four types of modules (see Figure 1, below): the encyclopedia module, the text corpus module, the exercise module and the resource module. The encyclopedia module and the resource module are aimed at providing information about a selected number of relevant themes and a pool of resources for further reading. In this paper the focus is on the Text corpus module and the Interactive exercise module.

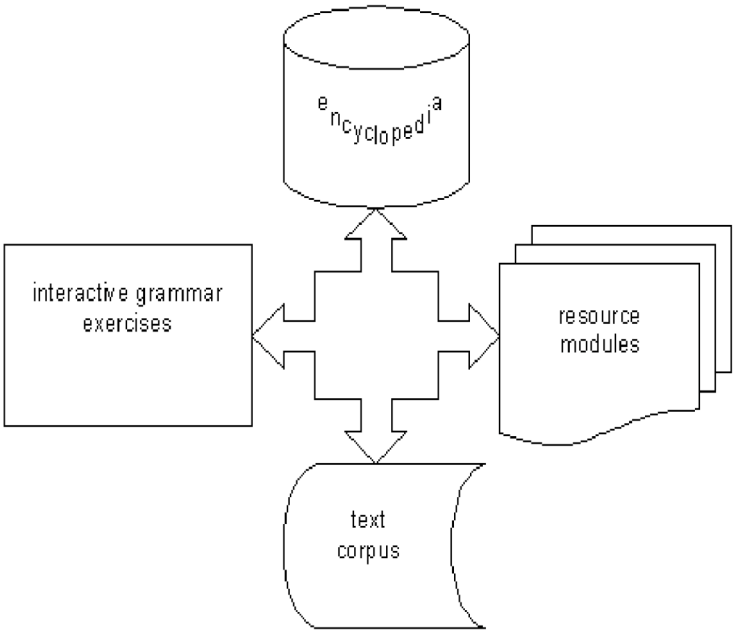

Figure 1: Organization of ITG

2.1 The 'Text Corpus' Module

The 'Text corpus' module contains a range of annotated corpora of Swedish and other languages. The corpora which are accessible to an occasional visitor of this website (category: Guest) are: the Stockholm-Umeå Corpus (SUC), ASU – a longitudinal Swedish as second language corpus collected at Stockholm University (spoken and written parts), the Kinnauri corpus, Sfi – the Swedish for immigrants corpus, Skrivsyntax, ssm – Svenska som målspråk (Swedish as target language). Most of the corpora which are used in ITG are tagged for part of speech and lemmatized, and in some cases they are also syntactically annotated.

The fact that these corpora were available when the project started, had its advantages. But, it also raised issues of compatibility. A lot of effort was spent on standardizing the corpus resources. We have been forced from the onset to seriously discuss how to integrate existing NLP resources in our application, as well as how to make the application itself extensible, so that e.g. new language corpora or new annotations can be added (for details, see Saxena and Borin 2002). Preliminary work on incorporating a Turkish corpus in collaboration with Eva Csato has shown positive results, suggesting the feasibility of expanding the data base of the platform with a moderate effort.

A graphic interface function has been added to the text corpus module (see below, figure 2). With the help of this graphic interface students/researchers are

able to see a 'dispersion map' of how and where one particular morpheme or word occurs in the corpus, providing support in their work on the functions of grammar (Olsson and Borin 2000).

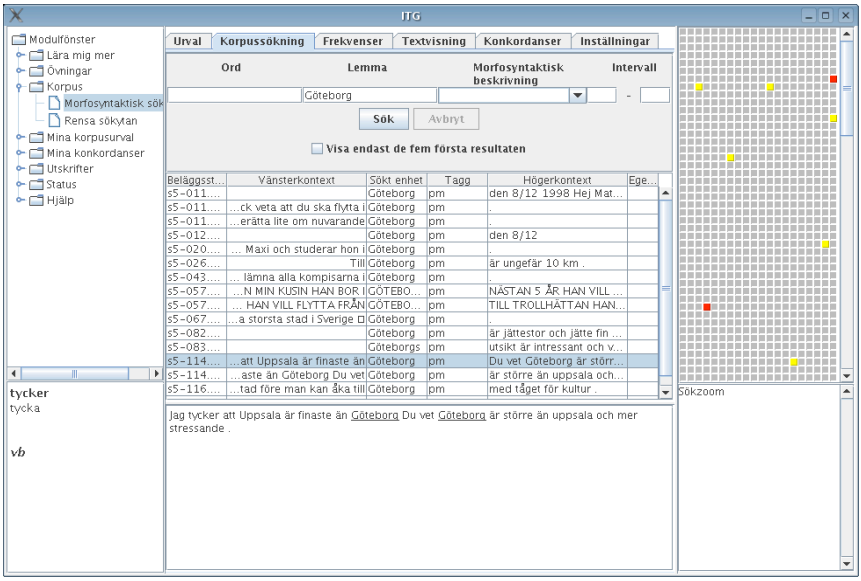

Figure 1: Organization of ITG

The use of the text corpus module is not necessarily connected to the exercise module (or to any other module). One can use the corpora available in this module as much for teaching purposes as for one's own research.

2.2 The 'Interactive Exercise' Module

The aim of the 'Interactive exercise' module is to provide students with means of learning and improving their skills in grammar. This is achieved in the two aforementioned courses in two ways: First, by means of corpus search (e.g., search for various lexical and grammatical elements in the corpus to find out, for instance, frequency, position and combination, word-formation patterns). The other use is by providing a framework for exercises on a selected theme. For example, one theme is "syntactic function" where the computer generates at random one example sentence from a pre-selected corpus with one or more words underscored, forming a syntactic constituent. The task for the student is to determine the syntactic function of this constituent (subject, object, etc.). He or she is expected to select the correct answer from a list of potential answers. If the answer is correct, this is shown to the user in two ways: (i) at the top of the screen, the correct answer along with green color indicate that the answer was correct and in the left bottom corner summary statistics of questions attempted

and the total number of correct answers are shown. If the answer was incorrect, the student gets immediate feedback in several ways: (i) at the top, along with the student's response, the correct answer appears (along with red color), (ii) in the right-hand bottom corner the correct answer appears, and (iii) in the left-hand bottom corner summary statistics of questions attempted and the total number of correct answers are shown (see Figure 3). The student can carry on with her exercises as long as she wants. The exercise module is constructed in such a way that there will not be any repeated test-sentence within 50 questions. For pedagogical reasons, the same format is used for other themes (for example, part of speech exercises).

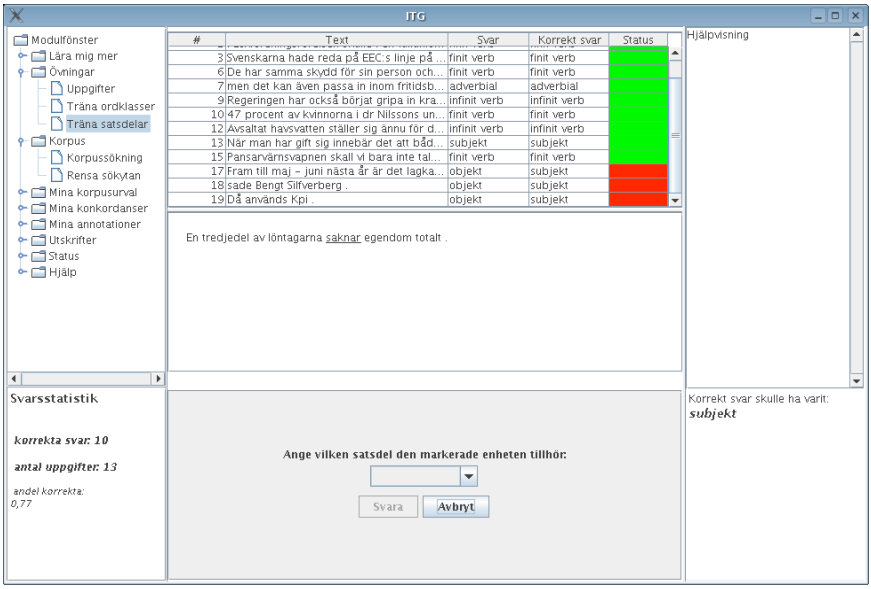

Figure 1: Organization of ITG

2.3 Design principles and general philosophy of ITG

The architectural organization of ITG has several advantages, the two most significant ones being *extensibility* and *conceptual decentralization*. Extensibility means that new functions can easily be integrated in the application. 'Conceptual decentralization' is especially significant as it allows the possibility of adjusting to individual learning styles. For example, if the student prefers to start out with the 'corpus' material (for example, doing a corpus search for a certain grammatical element) and go from there to the appropriate exercises (for example, part of speech exercises) when she feels the need to do so, she has that choice. At the same time, the application allows the possibility of starting out at other entry points, e.g., 'exercises', with the option of calling up the relevant 'encyclopedia' material at each instant.

Generally, we have used standard WWW and open-source software – i.e., software which is generally free and where the source code is freely available and modifiable by the user – for implementing the modules. This design philosophy has the advantage of making the application maximally platform-independent, as well as providing a familiar interface – a standard web browser – for students and faculty. Access to Internet for both students and teachers is the only technical requirement for the collaborative learning method proposed here.

3 Experience Using ITG at Uppsala University

Since the fall 2005 ITG is used as part of our courses *Lingvistik I* (Linguistics I), *Lingvistik II* (Linguistics II) and *Grammatik* (Grammar) in the Department of Linguistics and Philology, Uppsala University. We will focus our attention here on *Lingvistik I* and *Lingvistik II*, where we have primarily used only the Swedish corpora (POS-tagged and syntactically annotated). The organization of these courses has been as follows: lectures, group assignments and computer lab sessions. There have been three ITG lab sessions in each course. The groups were divided into smaller sub-groups consisting of 10–24 students in each sub-group. This was partly due to the number of computers available in the computer lab and partly so that the lab session leader could manage to supervise students individually. The students had free access to ITG during the course period. During the first lab session, the lab session leader (Mikaëla Lind) informed students about ITG, what a corpus is and how it could be used for grammar training, and about the SUC corpus. The focus in the first lab session has been on word formation, the theme of the second session was parts of speech and the theme of the third session was syntactic analysis.

Summaries of the student evaluations[2] are presented below, classified thematically (as some points occurred repeatedly in these evaluations).

3.1 Plenty of Group Opportunities

A recurring comment is that students would like more scheduled time together with a teacher, and above all more group assignments. Thanks to these lab sessions they get another three opportunities to practice and ask questions in small groups. This is appreciated by the students. The following are some comments from students (provided in English translations):

> I would like even more teacher guided group assignments.

[2] Students fill in course evaluations anonymously in the middle of a course and at the end of a course.

> The teaching was varied (lectures, lab sessions and group assignments).
>
> The lab sessions are great fun and you learn a lot from them.
>
> The lab sessions have been really good and instructive.
>
> The lab sessions were great, everything became a lot clearer.
>
> During the lab sessions in particular you get the opportunity to receive individual help from the teacher, which has been very helpful and important.

The purposes of the ITG lab sessions varied depending on which course they were part of, even if the instructions given to the students were the same. In *Lingvistik I* they serve as an introduction and practice, in Grammar as an exercise in order to strengthen knowledge. In *Lingvistik II* they are mainly intended as repetition and are therefore placed at the beginning of the course. This has been appreciated by those students who have not studied grammar for a while as they, thanks to this warm up, then feel more prepared for the more advanced level which is to come.

3.2 Good Mixture

A number of students commented that they like the fact that different kinds of teaching methods are used – lectures, group assignments, and, ITG lab sessions – as this combination provides a variation in teaching/learning. Some are of the opinion that the lab sessions provide an opportunity for a repetition, whereas for others it is an opportunity to learn and get some clarity in linguistic analysis. In this way the ITG lab sessions allow each and everyone to learn at their own pace and according to their own interests and needs. These are two student comments:

> The lab sessions are a useful supplement. If there is something you don't understand during a lecture you get another opportunity to listen, practice and understand at a lab session.
>
> The lab sessions are very helpful. They give an opportunity to, if you would like, look at different items in depth.

3.3 Practising from Home

A lot of students consider it helpful to be able to practice from home with the help of ITG. They get an introduction at a scheduled lab session and then they practice as much as they like from home, thanks to having access to a username and password valid throughout the course.

3.4 Unnecessarily long

The first lab session felt unnecessary, one student comments. This is another comment, possibly from the same person:

> However, it has felt as if the lab sessions with ITG have been unnecessarily long as you can just as well use the program from home and it gets a bit boring when you have been using it for a while.

It would be sufficient to be given an introduction, and then you can go on practising from home. However, our pedagogical idea is that the students should be able to practise with a teacher present. We have noticed that students, while working with ITG, raise questions about grammar and language structure which would not have arisen if they were working with traditional exercises. We believe that many of the students realized, for instance that there is a difference between verb particles and prepositions, and between predicatives and adverbials, thanks to plenty of practice. Many times they need to put their questions at the start in order to understand the following times.

3.5 Advantages of teaching in the computer lab

We have noticed that sitting in front of a computer has a number of advantages for the students. Several people made notes by writing in a word processor document instead of using pen and paper, which probably suits some people better. In addition, some students (but surprisingly few) were innovative and realized that they could look for information relevant for the lab on the internet. For example when they were not sure about some part of speech, they would look up a page on the net to read more, mainly on Wikipedia and *Nationalencyklopedin*, we believe. Some thought of looking for a definition of the word “lemma”, which is one of the tasks in the first lab session. So, apart from being a useful tool for practising grammar, the ITG lab sessions also give an enhanced possibility to search for information.

3.6 Co-operation

Working in a computer lab where students need to share computers, has the advantage that students are encouraged to work together in teams, discussing the problem and then arriving at a solution. One important duty of the teacher in such situations is to encourage students to raise questions and also to learn to discuss a theoretical issue. By verbalizing one's thoughts and comments on how somebody else is thinking, one deepens and strengthens the knowledge.

4 Conclusion

To conclude, student evaluations suggest that students are mostly appreciative of the ITG lab sessions as part of their regular course work. ITG offers an accessible, flexible tool – based on natural language corpora – to help them deepen their knowledge about grammar and do linguistic analysis. Students find ITG flexible – it allows them to work at their own pace, and also, if needed or desired, they use ITG anytime (and as much as they like) during the course period from home or university any time of the day (apart from the scheduled computer lab sessions). Further, it allows them to work in small groups and with scope for discussions. In this way they learn by doing linguistic analysis as well as discussing various aspects of grammar, thus gaining a deeper understanding than would have been the case otherwise.

References

Aston, G. (1997). Small and arge corpora in language learning. In B. Lewandowska-Tomaszczyk and P.J. Melia (eds.), *PALC 97: Practical Applications in Language Corpora*, pp. 51–62. Łódź University Press. <http://www.sslmit.unibo.it/~guy/wudj1.htm>.

Ghadessy, M., et al. (eds.) (2001). *Small Corpus Studies and ELT: Theory and Practice*. John Benjamins.

Johns, T. (1991). Should you be persuaded: Two examples of data-driven learning. *English Language Research Journal* **4**:1–16.

Knowles, G. (1990). The use of spoken and written corpora in the teaching of language and linguistics. *Literary and Linguistic Computing* **5**(1):45–48.

Olsson, L.-J. and L. Borin (2000). A web-based tool for exploring translation equivalents on word and sentence level in multilingual parallel corpora. In *Erikoiskielet ja kännösteoria – Fackspråk och översättningsteori – LSP and Theory of Translation. 20th VAKKI Symposium*, pp. 76–84. University of Vaasa.

Saxena, A. and L. Borin (2002). Locating and reusing sundry NLP flotsam in an e-learning application. In *LREC 2002 Workshop Proceedings: Customizing Knowledge in NLP Applications – Strategies, Issues and Evaluation*, pp. 45–51. Las Palmas.

Prospects and Trends in Data-Driven Machine Translation

Jörg Tiedemann

University of Groningen
Alfa Informatica

1 Introduction

In the past decade we have seen an amazing revival of machine translation (MT) as the major field of research in computational linguistics. Many reasons can be mentioned to explain this phenomenon: Globalization and the success of the Internet may be one of them forcing companies and individuals to adapt to a multilingual world. Political initiatives may also spur the fast development of MT systems, especially for previously unsupported language pairs. The increased capability of computers to handle large amounts of data might be another factor contributing to the accelerated development especially of data-driven MT approaches. But we should not forget to mention that working with translation is also a lot of fun attracting people all over the globe. And, finally, MT is an appealing research area because it faces many unsolved problems integrating various tasks in natural language processing.

In this paper we look at some general issues in data-driven machine translation and discuss possible strategies to improve translation quality. We review some of the recent trends in the research community without looking into the details of possible implementations and system development.

2 Typical Problems with Data-Driven MT

The current trend in MT research is to use existing large collections of data to learn to translate from one language to another. This includes not only parallel data from documents and their translations but also comparable corpora, monolingual corpora, bilingual dictionaries and other language(-pair)-specific resources. The term *data-driven* techniques is very broad and does not only include statistical and example-based approaches to MT. Rule-based and dictionary-based approaches require real-world data almost as much as their statistical counterparts. The difference between the various approaches is the amount of *trust* the data is given, i.e., how much the translation model

is *driven* by the patterns found in the data and how much manual work is needed to clean up and formalize them in the paradigm used. Especially the last part, *manual work*, frightens many researchers and funding organizations because of the risk of exploding costs and development time. This pushes them towards fully automatic systems requiring more and more data and computational power for training. Progresss can certainly be seen with this approach, however, translation quality is still poor when looking at the current state of the art. Furthermore, data-driven MT bears the risk of turning MT into a research competition with the focus on achieving the world record in BLEU scores. Instead one should concentrate on real users' needs again: MT should provide readable translations of texts not isolated sentences. It should not be based on an optimized number of matching N-grams but should produce coherent documents with some fluency across sentences.

2.1 Limited Use of Context

One of the main problems with current approaches to MT is the restriction to the translation of isolated sentences. Most MT models consider the words appearing in one sentence only without looking at any information from the surrounding context. In this way the task of translation turns out to be very hard even for trained human translators due to the large amount of ambiguities in natural languages. On the other hand, human translators work differently, taking a wide range of contextual clues into account when translating from one language into another. Because of this, they have much less problems with ambiguity than a computational model with limited context and no domain knowledge. Most expressions can be translated unambiguously in their context even though they have several meanings in isolation or within the sentence they appear in. A well-known principle in word-sense disambiguation (WSD) is the "one sense per discourse" assumption (Gale et al., 1992) which is similarly applicable in machine translation. The possibilities and the usefulness of WSD in machine translation has been discussed for a long time. In statistical machine translation (SMT), disambiguation is mostly left to the target language model using local context only. Recently, WSD-style classifiers have been integrated in SMT in order to improve phrase and word selection (Carpuat and Wu, 2005; Giménez and Márquez, 2007; Vickrey et al., 2005). However, they could only show modest improvements if any at all or limited themselves to blank-filling tasks instead of full translation. The importance of wider context for disambiguation in SMT has already been recognized in Brown et al. (1991). However, the integration of a WSD module with rich contextual features into a typical phrase-based MT system appeared to be unsuccessful until some recent implementations (Carpuat and Wu, 2007a,b). According to their findings it seems to be very important to develop a WSD module that is dedicated to the task of translation, i.e. focused on the proper selection of target language words and phrases. Translation ambiguity is often different to the

(1) Movie: Cheaper by the Dozen

Does this mean we can't go to Dylan's birthday party? - That's exactly what it means.

- We bought his **presents** already.
- Vi har ju redan köpt hans **present**.

(2a) Movie: The Polar Express

He said Santa would have to fly faster than the speed of light ... to get to every house in one night.

And to hold everyone' s **presents** ... his sled would be bigger than an ocean liner.

Och om alla **julklappar** skulle få plats måste släden vara större än ett fartyg.

(2b) Movie: The Polar Express

Checking out my **presents** .

Jag kollar också in mina **julklappar** .

Figure 1: Contextual clues for disambiguation.

one addressed in monolingual WSD. Furthermore, ambiguity is language-pair dependent, i.e. polysemy may be (partially) preserved when translating from one language to another or additional ambiguity can be introduced because of a richer distinction between sub-senses in one of the languages. Hence, WSD in MT has to be language-pair dependent and usually requires a rich set of contextual and domain-specific features.

The importance of wider context for lexical choice can easily be seen in some simple examples. Let us look at a corpus of translated subtitles (Tiedemann, 2007) which provides an interesting resource for translation issues across various genres, domains and language pairs[1]. For example, the English noun *presents* can be translated into the Swedish counterparts *presenter*, *gåvor* or *julklappar*. The first two are more or less synonymous whereas the latter is used for Christmas presents only.

Looking at the examples in figure 1 we can see that none of the English sentences to be translated contain a clear contextual clue to choose between the general sense and the Christmas sense (besides the weak indication by *sled* in 2a). However, the previous sentence in example (2a) contains the name *Santa* which intuitively promotes the translation in the sense of Christmas presents. Similar examples in which contextual clues can be found beyond sentence boundaries are very frequent.

[1]The subtitle corpus is a collection of 38,825 subtitle files in 29 languages. The entire corpus is sentence aligned (361 language pairs) and covers 2,780 movies. It is part of OPUS, which is a collection of parallel corpora and tools freely available at http://www.let.rug.nl/tiedeman/OPUS/.

The example (2b) in figure 1 shows another common case. Here, no contextual clue can be found even in the surrounding sentences (which are omitted here). However, from before (example (2a) from the same movie), we know that we talk about Christmas and, hence, the translation *julklappar* becomes more plausible than *presenter*. This demonstrates how disambiguation decisions should be saved and considered in later cases.

There are many similar examples in our data collection. For instance, the verb *swings* in a movie about Muhammed Ali is translated as *slår* (hits) whereas it is translated as *svänger* in *Der Untergang*. For the latter we can find the contextual clue *I want to dance* in previous sentences. Similarly, *fight* is translated as *boxas* throughout the Muhammed Ali movie whereas it is consistently translated as *slåss* in *Gladiator* and *krigar* in *Helen of Troy*. *License* is translated as the general term *tillstånd* in *The Godfather* whereas it is translated as *körkort* (drivers license) in the movie *Taxi Driver* (for which the title could be used as a clue).

To sum up, it seems to be necessary to go beyond the sentence boundaries when trying to resolve a large amount of ambiguities. Contextual clues should also include information about domain and topic and in addition disambiguation should make use of previous decisions. The main difficulty is, of course, the identification of appropriate patterns and the integration of disambiguation history. The risk is to give too much weight to spurious features in wide context and to propagate errors.

2.2 Missing Quality in Generation

Naturally, most MT research is focused on the transfer of linguistic units from source language to target language. On the other hand, the generation of proper target language sentences is often simplified to a large degree. However, readability of a translation highly depends on the grammaticality and fluency of the target language produced. It is not a coincidence that human translators usually translate into their native language. Proficiency is needed much more in the target language than in the source language. Despite of this fact many MT systems spend more time analyzing the source language properly in order to transfer all linguistic units recognized into appropriate target language expressions. Generation is then often a rudimentary component, for example, a simple language model or a module fixing agreement and other morphological issues. Human translators on the other hand do not spend so much effort in analyzing every single expression in the foreign language but focus on a proper formulation in the target language. A fluent and grammatically correct translation is more satisfying than a complete but unreadable translation even if some minor facts are missing. Certainly, the translation should be as correct as possible and should not change the meaning expressed.

The challenge for MT is to find a way to approach human translation with its proficiency. An MT system should be able to grasp the general meaning of a

foreign text (with permissible mistakes) and to generate perfect target language output representing similar contents. For this, analysis and transfer could be done on a shallow semantic level focusing more on the proper generation of readable target language text. And, again, readability and fluency should not be limited to isolated sentences.

Finally, generation should also take the type of reader into account. Proper translation has to be adjusted to the actual user group which is targeted by the translation. A text for domain professionals has different properties than, for example, one produced for children or teenagers. Certainly, the original text will be focused on a certain target group already but translation can still benefit from modeling the target group of the translation appropriately.

2.3 The Trap of Open-Domain MT

Many MT projects emphasize the development of general-purpose open-domain systems. However, working with a combination of various domains and topics leads to increased amounts of ambiguities that have to be handled. On the other hand domain knowledge is important for proper translation and helps to reduce ambiguity to a large degree. In many data-driven approaches the focus is set on increasing the amount of training data to improve translation quality. However, blending various domains into one general model also introduces a lot of uncertainty into the system which is avoided if the domain could be detected and designated models would be used to translate texts coming from that domain.

As discussed earlier, ambiguous words are often translated consistently within one text. Various senses can be ruled out when knowing the topic and domain. Looking at human translators again we can see that they are topic aware and often trained for certain domains. Hence, MT should not only be domain and topic specific but should also use dynamic models that adjust themselves according to the contextual history. An open-domain MT system should then work in the style of a mixture of experts in which translation is delegated to the appropriate sub-domain model. And, the delegation should not only be based on the current sentence but on recent context as well.

2.4 Oversimplification and Exaggeration

Even though many projects claim to work on hybrid systems we can still see two major directions: rule-based MT systems with deep linguistic analysis and statistical/example-based MT systems with shallow and often language independent techniques. In hybrid approaches people try to integrate syntactic information into statistical models and statistical information into rule-based systems. However, the amount of linguistics and statistics used in these system is usually fixed for any input. Purely statistical (language independent) approaches suffer from *oversimplification* in cases of complex linguistic patterns

which are difficult to transfer into similar target language units. Rule-based systems suffer from *exaggeration* of linguistic processing in very simple cases that would better be handled by simple pattern matching techniques. Furthermore, rule-based systems are not very robust in terms of handling unknown constructions which leads to coverage problems.

Hybrid MT systems should tackle these problems in a clever way such that each particular input is handled by only those components which can process that type of data properly and most efficiently. Statistical systems should be able to consult linguistic components in the cases where extra information is needed (using some notion of uncertainty in translation for this translation). Conversely, deep linguistic processing is not necessary in many cases, for example the ones that can be translated (almost) literally into the target language (avoiding errors that may be introduced by the various components). Furthermore, it could be a good idea to leave elements underspecified if ambiguity cannot be resolved with high confidence. Making no decision is often better than taking the wrong turn. For example, the following translations (including the metaphorical use of *geschluckt* (swallowed) in the sense of *believing a lie*) can be found in the subtitles of the movie *Good Bye Lenin*:

Hat sie's geschluckt?- Ja, klar.
Har hon svalt det ? - Ja , naturligvis .
Has she swallowed it ? - Yes , of course .

A deep semantic analysis of the source sentence would probably lead to a lot of ambiguities possibly causing translation difficulties whereas the translation is more or less literal in this example.

An MT system should choose the *easiest* way to get to the desired output avoiding follow-up errors of non-perfect processing steps. The challenge is to detect the amount of processing necessary for a particular input to allocate appropriate units in the system and to estimate confidence in specific outputs of various components.

2.5 Working with Sparse Data

Data-driven MT like all statistical NLP techniques is doomed to work with sparse data. For example, sufficient data for statistical MT will always be available only for a few language pairs and only for certain domains. Hence, an important task in data-driven techniques is to make better use of the data at hand. Comparing SMT/EBMT with human translators again, we can see that there is an important difference in translation learning. In both cases examples of previous translations are used to improve translation skills. However, statistical approaches to MT focus on handling larger amounts of data in order to improve coverage and quality, whereas humans are able to learn from much fewer examples and even profit from looking at the same examples over

and over again. This is because humans analyze/generalize examples in many different ways, looking for patterns and clues trying to understand these examples better and better. Machines could do the same, starting with simple techniques, memorizing previous translations, finding frequent equivalents but then starting to identify more complex, less frequent but important patterns in the example data. Examples should be explored more exhaustively than done by SMT and example-based approaches using broader context and more flexible patterns. Give the machine more time to learn instead of just more data! There is more to discover in every text! Maybe machine learning techniques should be developed which are closer to human learning considering the success of humans in tasks such as translation. This should probably also include a component that filters extracted patterns in some post-processing step in order to "forget" irrelevant information and to focus on quality instead of quantity.

3 What Can We Do?

Most of the issues mentioned above are difficult to address in automatic learning processes. Many research groups work on the integration of linguistic knowledge and richer context into data-driven MT. The main issue is to find a way to identify appropriate patterns in the data avoiding distracting signals that lead to overfitting problems. The following discusses several ideas in addressing some of the issues mentioned in the previous sections without giving a solution ready to be implemented.

3.1 Adaptive Models

Not without reason, domain adaptation is a hot topic in MT research. As discussed earlier a lot of ambiguity could be removed by considering information about domain and topic and by including broader contextual clues. Recent experiments in domain adaptation have shown that out-of-domain data is very hard to handle by statistical (and probably other types of) MT systems (Callison-Burch et al., 2007).

In SMT, a recent trend is to work with mixture models that include data from various domains (Foster and Kuhn, 2007; Civera and Juan, 2007). Initially people investigated adaptive *language models* in SMT using existing markup or information retrieval techniques (Zhao et al., 2004; Byrne et al., 2004). The target language model is responsible for a lot of disambiguation and, hence, domain adaptation in this part of the system is a sensible thing to do. However, lexical choice is of course also guided by the translation model and, therefore, several groups have looked into the adaptation of translation models as well (Hildebrand et al., 2005; Koehn and Schroeder, 2007). So far, the success is rather modest but research in this field has just begun and will probably go beyond the current results in the near future.

Future systems should include modules to detect domains and topics automatically. The output of this type of classification should be used directly by the system to dynamically adjust parameters of the translation models. One idea is to simply adjust the weights attached to components of a multi-domain mixture model. Other ways of adjusting model parameters dynamically might be possible.

Including contextual history into the system as discussed earlier could be done by building cache-based translation models. In language modeling, caching is known to reduce perplexity (Clarkson and Robinson, 1997; Iyer and Ostendorf, 1996) and similar techniques might be applicable in translation models as well. This would address the idea of consistently preferring certain word/phrase senses within one text. Stylistic information might also be integrated into such a dynamic translation engine. Another idea is to allow the system to benefit from user feedback in order to correct certain mistakes that will lead to follow-up errors in the dynamic settings. Such negative data can be very valuable for the success of the MT system in the future.

3.2 Proper Level of Complexity

Combining various approaches into one hybrid MT system is the goal for many research groups. Defining a system with dynamic selection of strategies depending on the input is a challenging task. The main problem is to detect the necessity of further processing. A simple approach could be to define a cascading system with fallback strategies in which simple techniques are tried first and in cases of low certainties more complex techniques are applied thereafter. Certainty can be measured using the internal parameters of the translation model or some kind of extrinsic evaluation of the output. For example parsability could be a criterion or some measure of readability and fluency in connection with previous context.

Another idea is to build a system that shows intermediate results to the user but continuous processing with more complex models until the user stops the process (or the entire process is done). In this way, the system would work as an interactive system allowing the user to interrupt further processing. This enables the user to wait longer hoping for better results or to go further if satisfied or in a hurry. Furthermore, post-correction could easily be integrated in such an interactive system, again, helping the system to learn from the feedback by the user.

3.3 Data Mining for Contextual Clues

In previous sections, we have discussed the necessity of rich contextual features to improve lexical choice and other types of disambiguation in MT. The main problem is to identify appropriate patterns automatically in given training

data[2]. The main problem here is the huge search space and the risk of including spurious features. It is not only that contextual features can be arbitrarily far away but patterns may also include any combination of such features making it impossible to carry out an exhaustive search through the data. However, certain strategies may help to explore the feature space systematically.

For word and phrase selection, we can first define default translations based on the most frequent alignments found in the bilingual training data. The next step is to search the example data for consistent indicators for each alternative translation. This can be done by iteratively checking surrounding features, starting with local context and then expanding to wider context and feature combinations. Here, well-known data mining techniques for association rule extraction can be applied.

Most of these algorithms are based on frequent set mining extracting rules with high confidence and a minimum of support. The antecedent of such a rule in our task would then be the contextual condition and the consequent is the selected translation. For each ambiguous word or phrase a set of rules could be extracted covering various kinds of contextual dependencies. Doing this exhaustively for all words and phrases in the translation table is still not feasible. However, it can be done for selected items to gradually improve disambiguation in the system.

To test this technique on a tiny example we extracted the sentences and their surrounding context (two sentences before and two after) that contain the English word *presents* from our subtitle corpus and disambiguated each occurrence according to the alignment in Swedish. We kept only those sentences for which the Swedish counterparts contain the strings *presentera* (to present), *present* (a present), or *julklapp* (Christmas present). The term is actually not very frequent in the corpus and after filtering there are 26 occurrences left. Here, we limited ourselves to alphabetic surface words (in lower case) only and ignored all other types of annotation we could use. Using an implementation by Borgelt (2003) of a standard algorithm for the extraction of association rules, Apriori (Srikant et al., 1997) we obtained the following rules implying the *julklapp* sense:

```
julklapp <- christmas (11.5, 100.0)
julklapp <- santa (11.5, 100.0)
julklapp <- santa and (11.5, 100.0)
```

Note that *Santa* never occurs within the same sentence as *presents* and also *Christmas* is only in local context in half of the cases.

Due to the fact that we did not use any disambiguation based on part-of-speech labelling the homographic use of *presents* as a verb can also be recognized by simple contextual rules. The following association rules (for sets smaller than four words) have been found for the verb sense denoted by the Swedish equivalent *presentera*:

[2]Here, we consider the case of using parallel translation data to train the system.

```
presentera <- ladies (11.5, 100.0)
presentera <- gentlemen (11.5, 100.0)
presentera <- ages (11.5, 100.0)
presentera <- proudly (15.4/4, 100.0)
presentera <- ladies gentlemen (11.5, 100.0)
presentera <- ladies ages (11.5, 100.0)
presentera <- ladies proudly (11.5, 100.0)
presentera <- ladies all (11.5, 100.0)
presentera <- ladies and (11.5, 100.0)
presentera <- ladies the (11.5, 100.0)
presentera <- gentlemen ages (11.5, 100.0)
presentera <- gentlemen proudly (11.5, 100.0)
presentera <- gentlemen all (11.5, 100.0)
presentera <- gentlemen and (11.5, 100.0)
presentera <- gentlemen the (11.5, 100.0)
presentera <- ages proudly (11.5, 100.0)
presentera <- ages all (11.5, 100.0)
presentera <- ages and (11.5, 100.0)
presentera <- ages the (11.5, 100.0)
presentera <- proudly all (11.5, 100.0)
presentera <- proudly and (11.5, 100.0)
presentera <- proudly the (15.4, 100.0)
```

Here, *proudly presents* is a typical collocation in the corpus whereas the association to *Ladies and Gentlemen* usually involves wider context. Certainly, the example above does not prove the usefulness of rule mining approaches in MT. However, their potentials should be clearly visible.

There are many practical issues that have to be addressed in the rule mining approach. First of all, we still have to limit the context to be used even in simple bag-of-words approaches. However, we can easily go beyond the sentence boundaries as shown in the simple example above which is desirable as we have discussed in earlier sections. Note that in real application we want to include various kinds of linguistic annotation as possible features (part-of-speech labels, chunk labels, dependency relations, etc). Another difference to classical data mining techniques is that positional information might be important. Including positional dependencies would increase the search space tremendously and, therefore, only some possibilities can be explored. Some heuristics have to be defined to select specific settings to be checked by the rule extraction algorithm. Again, one can start with simple features moving to more complex ones.

A nice feature of a rule extraction approach is that intermediate results can immediately be applied in the MT system. Training may continue infinitely trying to explore other patterns in the example data using more and more complex feature sets. Furthermore, the exploration can be done in parallel for each ambiguous item in the lexicon. A global search is not necessary.

A general problem with a data mining approach is the requirement of frequent sets. Interesting associations may be missed in this way due to sparseness of the data (a general problem in all data-driven techniques). There are

algorithms to extract high confident rules without minimum support. However, there is a large risk of collecting noisy relations with spurious associations. It might help to restrict the contextual freedom for such rules in order to decrease the chance of such spurious rules. It could be an idea to define the minimum support required for a rule as a function of feature complexity used in the antecedent of the rule.

Another idea is to merge contextual features of semantically related items into semantic classes. For example, algorithms that explore distributional similarity in large monolingual corpora can be used to extract classes of related words (Lonneke van der Plas, 2006). They can be used to form a contextual feature representing the semantics of all these related words. The intuition behind this is to obtain a higher level of abstraction and to focus on the meaning of words rather than their physical appearance in the text. In this way, one hopes to find better generalizations with increased support and confidence. For example, the translation of *presents* into the Swedish *julklappar* as discussed previously may use the contextual clue of a word related to Christmas. The problem, however, is the additional noise produced by the algorithms for extracting such classes and the ever increasing number of features.

The final task is to integrate the identified patterns into the MT system. This, of course, highly depends on the general approach taken. A rule-based system could simply create transfer-rules from contextual association rules found in the data mining step. They might be validated by experts before being put to use. Automatic validation using MT evaluation metrics may be another option. In SMT systems association rules and the confidence scores connected to them could be included as an additional component in the global statistical model. Example-based MT systems could use them for ranking possible bitext pairs.

4 Conclusions

The main purpose of this paper is to collect ideas about possible improvements of data-driven techniques in machine translation. The main claims can be summarized as follows:

- MT should go beyond translating isolated sentences. Wider context should be used for disambiguation and for improving readability and fluency of the output. For this contextual history should be cached and used to adjust the MT model to the current discourse dynamically.
- The generation of the target language should be improved. Grammaticality and readability are neglected by many systems forgetting about the real users needs.
- Domain knowledge and topic recognition should be integrated into wide-coverage MT systems. The system should function as a mixture of translation experts with specific domain knowledge.

- Hybrid MT systems should take advantage of the success of different approaches. However, the combination of various techniques should be more flexible allowing the system to acquire additional information if necessary but leaving things simple if possible.

- Data-driven techniques should make better use of the limited amount of data available. Carefully mining example data for patterns may help MT systems more than adding large amounts of noisy data.

- Mechanisms allowing the system to benefit from user feedback could improve quality and validate model parameters.

Admittedly, none of the claims are carefully evaluated here and, therefore, cannot be taken as the ultimate road for MT development. However, they are meant to give inspiration to further advances in an exciting research area.

References

Borgelt, C. (2003). Efficient implementations of apriori and eclat. In *Proceedings of the 1st Workshop of Frequent Item Set Mining Implementations (FIMI 2003)*, Melbourne, FL, USA.

Brown, P. F., S. D. Pietra, V. J. D. Pietra, and R. L. Mercer (1991). Word-sense disambiguation using statistical methods. In *Meeting of the Association for Computational Linguistics*, pp. 264–270.

Byrne, W., S. Khudanpur, W. Kim, S. Kumar, P. Pecina, P. Virga, P. Xu, and D. Yarowsky (2004). The Johns Hopkins University 2003 Chinese-English machine translation system. In *Proceedings of the 2003 NIST MT Workshop*.

Callison-Burch, C., P. Koehn, C. S. Fordyce, and C. Monz (Eds.) (2007). *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic: Association for Computational Linguistics.

Carpuat, M. and D. Wu (2005). Word sense disambiguation vs. statistical machine translation. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Morristown, NJ, USA, pp. 387–394. Association for Computational Linguistics.

Carpuat, M. and D. Wu (2007a). How phrase sense disambiguation outperforms word sense disambiguation for statistical machine translation. In *Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2007)*, Skövde, Sweden, pp. 43 – 52.

Carpuat, M. and D. Wu (2007b). Improving statistical machine translation using word sense disambiguation. In *Proceedings of the Joint Conference*

on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007), Prague, pp. 61 – 72.

Civera, J. and A. Juan (2007). Domain adaptation in statistical machine translation with mixture modelling. In *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, pp. 177–180. Association for Computational Linguistics.

Clarkson, P. and A. J. Robinson (1997). Language model adaptation using mixtures and an exponentially decaying cache. In *Proc. ICASSP '97*, Munich, Germany, pp. 799–802.

Foster, G. and R. Kuhn (2007). Mixture-model adaptation for SMT. In *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, pp. 128–135. Association for Computational Linguistics.

Gale, W. A., K. W. Church, and D. Yarowsky (1992). One sense per discourse. In *HLT '91: Proceedings of the workshop on Speech and Natural Language*, Morristown, NJ, USA, pp. 233–237. Association for Computational Linguistics.

Giménez, J. and L. Márquez (2007). Context-aware discriminative phrase selection for statistical machine translation. In *Proceedings of WMT 2007 at ACL*.

Hildebrand, A., M. Eck, S. Vogel, and A. Waibel (2005). Adaptation of the translation model for statistical machine translation based on information retrieval. In *In Proc. of the 10th EAMT Conference*.

Iyer, R. and M. Ostendorf (1996). Modeling long distance dependence in language: Topic mixtures vs. dynamic cache models. In *Proc. ICSLP '96*, Volume 1, Philadelphia, PA, pp. 236–239.

Koehn, P. and J. Schroeder (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, pp. 224–227. Association for Computational Linguistics.

Lonneke van der Plas, Gosse Bouma, J. M. (2006). Automatic acquisition of lexico-semantic knowledge for qa. In C.-R. Huang (Ed.), *Ontologies and Lexical Resources for Natural Language Processing*. Cambridge University Press, Cambridge, UK, University of Sinica.

Srikant, R., Q. Vu, and R. Agrawal (1997). Mining association rules with item constraints. In D. Heckerman, H. Mannila, D. Pregibon, and R. Uthurusamy (Eds.), *Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining, KDD*, pp. 67–73. AAAI Press.

Tiedemann, J. (2007). Improved sentence alignment for movie subtitles. In *Proceedings of RANLP 2007*, Borovets, Bulgaria, pp. 582–588.

Vickrey, D., L. Biewald, M. Teyssier, and D. Koller (2005). Word-sense disambiguation for machine translation. In *Proceedings of HLT/EMNLP 2005*.

Zhao, B., M. Eck, and S. Vogel (2004). Language model adaptation for statistical machine translation with structured query models. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, Morristown, NJ, USA, pp. 411. Association for Computational Linguistics.

Riding, Driving and Traveling – Swedish Verbs Describing Motion in a Vehicle in Crosslinguistic Perspective

Åke Viberg

Uppsala University
Department of Linguistics and Philology

1 Introduction

This paper will discuss a small but central group of motion verbs in Swedish from a crosslinguistic perspective: the verbs describing motion in a vehicle. Data will be drawn primarily from two translation corpora.

The lexicalization of motion verbs involves a variety of basic semantic parameters which can take a number of values. Typologically Path (to/from, up/down, in/out etc) has been the center of attention (Talmy 1985, Bohnemeyer et al 2007) and this is indeed the most basic one, but there are many other parameters that are relevant (see Viberg 1992, 2006a for an overview of Swedish motion verbs).

An important characteristic of subjects that refer to human beings is their propensity for self-propelled motion. This feature is acquired early by infants (Spelke et al. 1995). Unlike a ball that starts moving because it is hit by something, human beings have an inner source of energy, which can make them move. Bodily locomotion verbs more or less by definition describe self-propelled motion. But even motion by other types of subjects such as natural forces (the wind, rain, etc) can be conceptualized as self-propelled, which makes this a parameter that is not limited to bodily locomotion. The most basic type of self-propelled motion for humans is walking, which is expressed in Swedish with the verb *gå* described in an earlier paper (Viberg 1999a). A smaller but rather central group of motion verbs which typically take a human subject describe motion in a vehicle such as *ride* and *drive* in English. Vehicle and means of transportation in general represent a basic parameter that has traditionally been subsumed under Manner. This paper will be concerned with the semantic differentiation between the basic Swedish verbs *rida*, *köra*, *åka*, *fara* och *resa* as well as with their patterns of polysemy which extend beyond the motion field. Interestingly, the concepts of Force and Control are central as organizing principles for most of them (the exception is *resa*) and for this reason Swedish *driva,* the cognate of English *drive,* is also included in the discussion. The paper will primarily be concerned with verbs that describe motion in a vehicle in general and not with verbs that specifically refer to

aquamotion (Koptjevskaja-Tamm et al., forthc.) or motion in the air (*flyga* 'fly').

In this paper, Swedish will be compared to a selection of closely related languages on the basis of data from two translation corpora. The analysis of Swedish is also based on the monolingual corpora in the Swedish Language Bank (Språkbanken). (Examples from these corpora have been translated by me.) One of the translation corpora is the English Swedish Parallel Corpus (ESPC) prepared by Altenberg & Aijmer (2000), which contains original texts in English and Swedish together with their translations. The texts are divided into two broad genres: Fiction and Non-fiction with several subcategories. The original texts in each language contain around 500 000 words. The other corpus is more restricted and will be referred to as the multilingual pilot corpus (MPC). It is being compiled by the author and consists at present of extracts from 10 novels in Swedish with translations into English, German, French and Finnish (totally around 250,000 words in the Swedish originals). For some of the texts, there are also translations into Danish and Dutch. As an illustration of the types of data this article is based on, examples from the corpus are shown in table 1. I will use some of the examples to introduce the major semantic parameters that will be treated in this paper.

One basic contrast is the differentiation between verbs profiling the control of a means of transport, such as riding and driving, versus the profiling of being transported, such as traveling and voyaging. The verb *rida* 'ride' is not as frequent as the other verbs but has been included since verbs with an equivalent meaning have extended their meaning in some of the other languages. This is most striking in Dutch, where *rijden* still can be used about riding a horse but is also frequently used with reference to driving and traveling in a car (and other vehicles). The basic Swedish verb used to express the driving of a car or other vehicle is *köra* 'drive' and this contrasts with *åka* and *fara* which profile the sense of being transported. Usually, the meanings do not mutually exclude one another: driving strongly invites the inference that the driver is being transported but not necessarily since you can drive certain vehicles with a remote control (in particular model cars). The verbs *åka* and *fara* can be used even when the subject refers to the driver but the control of the vehicle in this case is backgrounded. These verbs cannot be used when the control is strongly profiled as in "Do you know how to drive?" This contrast is not upheld lexically in several languages.

German tends to use the verb *fahren,* the cognate of Swedish *fara,* both to express driving and traveling, whereas Danish tends to use *køre*, the cognate of *köra* 'drive' with both of these meanings. Finnish *ajaa* has both of these uses, too. Compare examples (2) and (4) in table 1. A further semantic contrast is shown in example (5). Swedish *köra* can be used as a verb of transportation in a vehicle, where the displacement of the object of the verb is profiled. Danish *køre* and Dutch *rijden* can cover this sense as well.

(1)	Swedish:	”**Rida** skulle vi”, sa Ronja. AL
	English:	“We were going to **ride**,” said Ronia.
	German:	„**Reiten** wollten wir”, sagte Ronja.
	French:	„Tout ce qu’on voulait, c’était **monter à cheval**”, dit Ronya.
	Finnish:	– Meidän piti kai **ratsastaa**, Ronja sanoi.
	Danish:	”Vi skulle jo **ride**,” sagde Ronja.
	Dutch:	„We zouden toch gaan **rijden**,” zei Ronja.
(2)	Swedish	Och frun kunde inte ***köra*** traktorn. KE
	English	and his wife couldn't ***drive*** the tractor.
	German	Und seine Frau konnte nicht Traktor ***fahren***.
	French	Et sa femme ne savait pas ***conduire*** le tracteur.
	Finnish	Eikä vaimo osannut ***ajaa*** traktoria.
	Danish	Og konen kunne ikke ***køre*** traktoren.
	Dutch	en zijn vrouw kon geen tractor ***rijden***.
(3)	Swedish	***Åk*** med ner. KE
	English	***Come on*** down with us.
	German	***Fahr*** mit runter.
	French	***Va*** avec eux.
	Finnish	***Lähde*** mukaan
	Danish	***Kør*** med ned
	Dutch	***Rijd*** maar mee.
(4)	Swedish	och de ***for*** tillbaka till campingen. KE
	English	so they ***went*** back to the camping site.
	German	und die beiden Männer ***fuhren*** zum Campingplatz zurück.
	French	et ils ***retournèrent*** au camping.
	Finnish	ja he ***ajoivat*** takaisin leirintäalueelle.
	Danish	og de ***kørte*** tilbage til campingpladsen.
	Dutch	en ze ***reden*** terug naar de camping.
(5)	Swedish	Annie behövde inte ***köra*** henne sa hon. KE
	English	Annie needn't ***drive*** her down, she said.
	German	Annie brauche sie nicht zu ***fahren***, sagte sie.
	French	Ce n'était pas la peine qu'Annie la ***conduise***, dit-elle.
	Finnish	Hän sanoi ettei Annien tarvinnut ***lähteä viemään***.
	Danish	Annie behøvede ikke at ***køre*** hende, sagde hun.
	Dutch	Annie hoefde haar niet te ***rijden***, zei ze.

Table 1: Verbs describing riding and motion in a vehicle in the MPC corpus.

Verbs of traveling contrast with verbs describing self-propelled motion, the basic form of which is walking (for human beings). Languages differ with respect to whether there is a (more or less) obligatory contrast between walking and being transported in a vehicle. In Swedish, the verb *gå* ‘go’ can only be used with reference to walking (when the subject is human), and this leads to a frequent error even among many rather advanced learners of Swedish as a second language who can say *Jag gick med tåget/bilen till*

Uppsala 'I went by car/train to Uppsala, where *åka* must be used. One source of contrastive data for Swedish verbs of traveling is the Swedish grammar intended for Swedish as a second language, which has been translated into 18 languages (Viberg et al. 1984–). Examples (6)-(7) are taken from various versions of the grammar (op. cit. §15.5).

(6)	Swedish:	Min fru måste ***åka*** tunnelbana till jobbet.
	English:	My wife has to ***go*** to work by underground.
	Greek:	I jinéka mu prépi na ***pái*** me ton elektrikó sti ðuliá.
	Rumanian:	Soția mea trebuie să ***meargă*** cu metroul la service.
	Turkish:	Karım işine tunnel ile ***gitmek*** zorundadı
(7)	Swedish:	Men min arbetsplats ligger så nära att jag kan ***gå***.
	English:	But my office is so close that I can ***walk***.
	Greek:	Enó i ðikí mu ðuliá íne tóso kondá pu boró na ***páo*** me ta póða.
	Rumanian:	Locul meu de muncă este însă aşa de aproape, încît pot să ***merg pe jos***.
	Turkish:	Ama benim işyerim çok yakındır, ben işime ***yürüyerek gidibilirim***.

Data from a pedagogical grammar are not as reliable as data from corpora but the general picture appears to be correct. The verbs used to express the meanings of Swedish *gå* and *åka* and the other Swedish verbs discussed so far are shown in table 2. As can be observed, the verb corresponding to 'go' is also used to describe motion in a vehicle in English, (Modern) Greek, Rumanian and Turkish. In Greek, it is also used to express transportation in a vehicle. Thus, there is no obligatory contrast corresponding to that between *gå* and *åka*, even if specific verbs describing motion in a vehicle exist and are used to various degrees. (The Swedish verb *resa* is translated with such a verb except in Turkish.) In addition, all four languages have a separate verb that can be used when the subject refers to the driver operating the vehicle. In general, it can be concluded that verbs describing motion in a vehicle have many language-specific (but not necessarily unique) features in Swedish. In the following, these verbs will be looked at one by one in greater detail.

	Self-propelled motion on foot
Swedish:	gå
English:	go/walk
Greek:	páo
Rumanian:	a merge
Turkish:	gitmek/yürümek
	Motion in vehicle – being transported
Swedish:	åka/fara, resa
English:	go, travel
Greek:	páo, taksiðévo
Rumanian:	a merge, a călători
Turkish:	Gitmek
	Motion in vehicle – operate vehicle
Swedish:	köra
English:	drive
Greek:	oðiγó
Rumanian:	a conduce
Turkish:	sürmek
	Transportation in vehicle
Swedish:	köra
English:	drive
Greek:	páo
Rumanian:	a conduce
Turkish:	sürmek/götürmek

Table 2. The basic verbs for expressing self-propropelled motion and motion in a vehicle in Swedish and four other languages (based on Viberg et al 1984-, §15.5)

2 From Horse to Automobile: The Verb 'Ride'

The verb *ride* is a common Germanic verb with cognates in all the major Germanic languages. In Swedish, the verb has the form *rida*. The meaning 'move sitting on top a horse and controlling its movement' is still the dominant meaning in Swedish, where *rida* has not been extended to cover motion in a vehicle such as a car or a bicycle. These meanings are primarily covered by *åka*, whose meaning to a great extent overlaps with the present-day extended use of *ride* and its cognates, in particular in Dutch, as we have seen, but also to a certain extent in English. Since there are many interesting parallels to Swedish *åka*, a sketch of the meaning patterns of English *ride* is given in figure 1. This figure, like all the following figures of the same type, primarily shows sense relations in present-day English (or Swedish in sections below). Diachronic relationships are reflected only to the extent that

they are stated explicitly in the text. (Unlike the following figures, this one is not based on any systematic corpus study.)

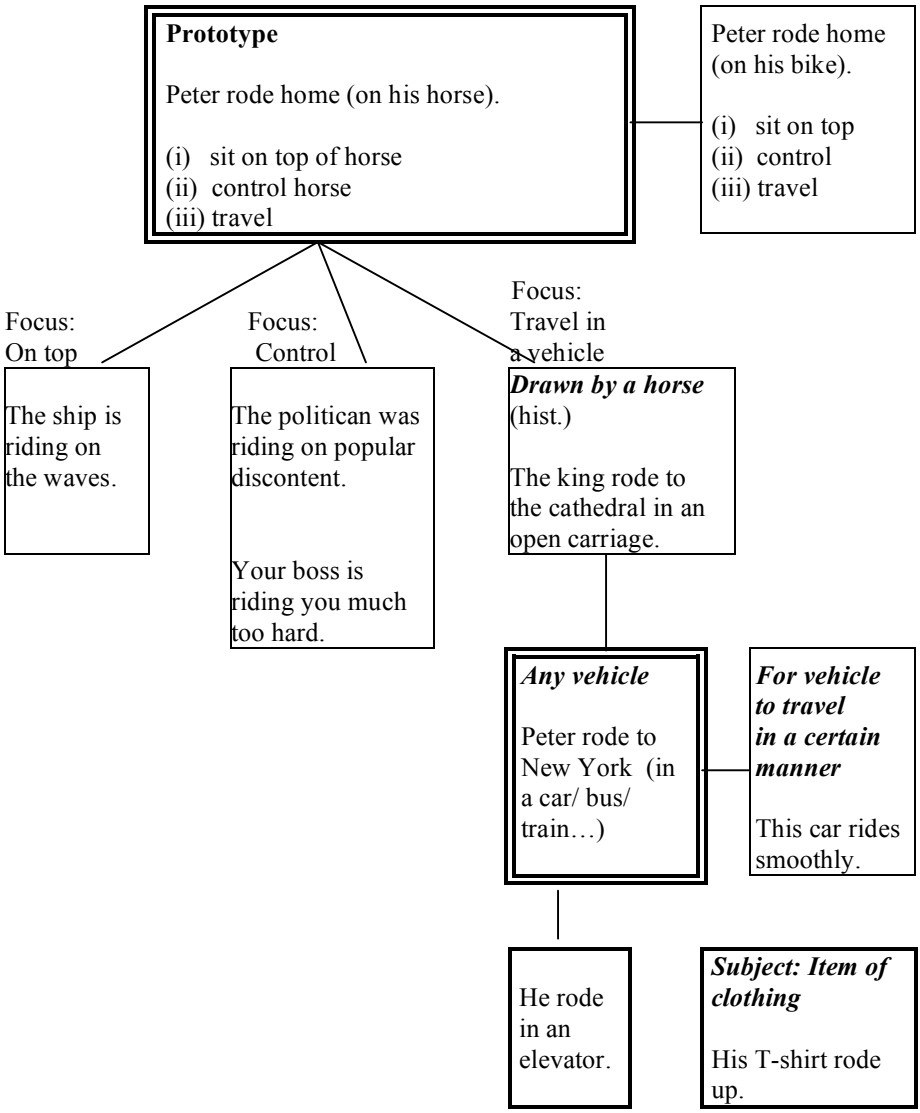

Figure 1: The meaning patterns of English *ride* (somewhat sketchy).

The common Germanic prototype of 'ride' seems to combine three major components (i) sitting on top a horse (ii) controlling the movement of the animal (iii) displacing oneself in this manner. Various secondary meanings focus parts of the prototypical meaning. With a vessel as subject, the notion

'on top of (the waves)' is focused, and in this case *rida* can be used in Swedish as in (7).

(7) the boat ***riding*** a run of wave prow up. DT
eftersom båten ***red*** på en våg med stäven i vädret,

Another case, where sitting on top is rather prominent is *ride on a bike*, but in this case *rida* cannot be used in Swedish but only *cykla* 'cycle' or *åka cykel* (see 8).

(8) Her upper lip was puffy, like the kind of scrape children get falling off bicycles when they first learn to ***ride***. SG
Hennes överläpp var svullen.Det liknade den sorts blessyr små barn brukar få då de ramlar omkull när de lär sig ***cykla***.

The notion of control, which is closely related to 'on top' (CONTROL IS UP), is focused in a number of metaphorical uses such as *The politician tried to ride on popular discontent*, which has a close Swedish equivalent. There is also a verb such as *topprida* 'ride on top', which is used metaphorically (*Så länge valuta- och räntemarknaderna topprider den svenska demokratin* P95). The most important set of extensions focuses on the fact of traveling, keeping the manner component in the background. Already in Middle English, *ride* could take on the extended meaning 'to be conveyed in a wheeled or other vehicle drawn by a horse' (OED). From this, there was a short step to riding in a train or a car, once these means of conveyance had been invented. With this extension, the meaning of *ride* is approaching the central meaning of the Swedish verb *åka*, non-self-propelled motion. In most cases where *ride* is used with reference to traveling in a conveyance, *åka* is used in Swedish as illustrated in (9) and (10).

(9) The dogs ***ride***, the poor walk, or go by bus. FW
Hundarna ***åker bil***; de fattiga går eller åker buss.
(10) He refused to speak or even look Andrew in the eye as they ***rode*** to the fourth floor. AH
Han vägrade säga något eller ens möta Andrews blick när de ***åkte upp*** till fjärde våningen.

The use in (11), which is restricted to items of clothing according to Cambridge International Dictionary of English(CIDE), also has a parallel in the use of Swedish *åka* (my translation):

(11) His T-shirt ***rode up***, when he bent over. CIDE
Hans T-shirt ***åkte upp***, när han böjde sig framåt.

If the interpretation sketched here is correct, there are two prototypical clusters in the meaning pattern of English *ride*. One is centered on the original prototype 'ride on a horse', and the other is emerging around the meaning in present-day English 'ride in a conveyance'. This latter meaning

has no counterpart in Swedish, but as we have seen *rijden* 'ride' has extended even further in Dutch.

3 Driving

According to Buck (1949), many of the words for 'drive' in Indo-European languages were originally used in the context of driving cattle (in front of oneself, as opposed to 'lead'). Another source is words with the opposite meaning, 'lead', which according to Buck applies to Romance words for driving such as Rumanian *conduce*, French *conduire*, Spanish *conducir*. The human agent in this situation acts as an outside force causing the cattle to move. Thus not only the idea of riding in a car as a passenger, but also the notion of controlling and driving a car were originally modeled on the guidance of domesticated animals. Other words for 'drive' have developed from 'strike' or 'push', both of which are centered on Force (see Viberg 1999b for striking). As will be demonstrated in this section, Force or Power/Control is also central in the meaning of verbs used about driving a car or other vehicle in modern Swedish and English.

3.1 The Swedish Verb *Köra*: 'Drive (a Vehicle)'

When the operation of a vehicle (typically a car) is profiled *köra* is used in Swedish. The profiling is particularly clear when a vehicle appears as direct object as in (12).

(12) Swedish: Ska du ***köra*** ambulans om du inte är nykter? PCJ
English: "Do you ***drive*** an ambulance even when you're not sober?"
German: „Willst du denn den Krankenwagen ***fahren***, wenn du nicht nüchtern bist?"
French: –Tu vas ***conduire*** ton ambulance même après avoir bu ?
Finnish: –Voitko ***ajaa*** ambulanssia, vaikka et ole selvä?

English and French in this case have special verbs (*drive*, *conduire*), whereas German and Finnish has one and the same verb corresponding to *köra* 'drive' and *åka* 'ride in a car etc.'. The contrast can be signaled by using different constructions. To profile 'driving', the vehicle appears as direct object as in (12) marked with the accusative in German and with the partitive in Finnish (alternating with the accusative in other examples). When the operation of the vehicle is not profiled as in (13), *åka* can be used even when the subject actually refers to the driver. In German and Finnish, the vehicle is marked as an instrumental (prep. *mit* 'with' in German and adessive case *–lla*(*-llä*) in Finnish.) If the subject of the verb refers to the driver of the vehicle, the choice between *köra* and *åka* is a question of profiling, since the driver simultaneously is being transported in the vehicle (in most cases).

(13) Swedish: Hon hade ***åkt*** **bil** här massor av gånger LM
English: She had ***driven*** past here many times
German: Unzählige Male war sie hier **mit dem Auto** entlang***gefahren***
French: Elle était ***passée*** par là **en voiture** à maintes reprises,
Finnish: Hän oli ***ajanut*** täällä lukemattomia kertoja **autolla**

As illustrated earlier in (5), *köra* can also be used as a verb of transport describing the movement of the object by driving a car or other vehicle. In this case, *köra* alternates with *skjutsa* which usually refers to human passengers and *frakta* which tends to refer to goods of various types. Examples (14) and (15) are from the ESPC. There are several further types of transportation verbs in Swedish referring to specific types of cargo or vehicles (see Viberg 1981, 59-62).

(14) Gammlundström brukade ***skjutsa*** henne, TL
Old Lundström used to ***drive*** her there,
(15) De stora spanska skeppen ***fraktad***e handelsgods från Amerika, BTC
Great Spanish ships ***carried*** freight from trade with America

The uses of *köra* involving a vehicle are the most frequent ones in present-day Swedish. The verb can, however, be used with reference to other types of forced motion. As is indicated in figure 2, the notion of causing something to move with force is the general (schematic) meaning of *köra*, which is shared by most uses. One set of uses describes how something held in the hand is set into motion, usually in a forceful way.

(16) Swedish: Björne ***körde*** en knytnäve i magen på honom. KE
English: Björne ***drove*** his fist into his stomach,
German: Björne ***jagte*** ihm die Faust in den Bauch.
French: Björne lui ***enfonca*** un poing dans le ventre.
Finnish: Björne ***iski*** häntä nyrkillä vatsaan.

The verb *köra* can also be used to describe how people or animals are driven away, usually verbally with a sharp order or (in the case of cats and similar animals) with a specialized interjection such as *shoo!* (Swedish *schas!*, which can be derived as a verb *schasa* 'shoo' vb.). (17) and (18) are taken from the ESPC.

(17) Bara öronen viftar till då och då som för att ***köra bort*** en fluga. PCJ
Only his ears move now and then, as if ***shooing away*** a fly.
(18) De kom alltid omgående och ***körde bort*** de som störde honom. HM
They always came right away and ***drove off*** anyone who was disturbing him.

The dominant meaning of *köra* in present-day Swedish is related to operating and traveling in a vehicle. However, the verb has a number of other uses that share a schematic meaning which can be expressed

approximately as “Forcefully cause to move”. In a way, the meanings related to operating a vehicle are set off from the rest of the meanings.

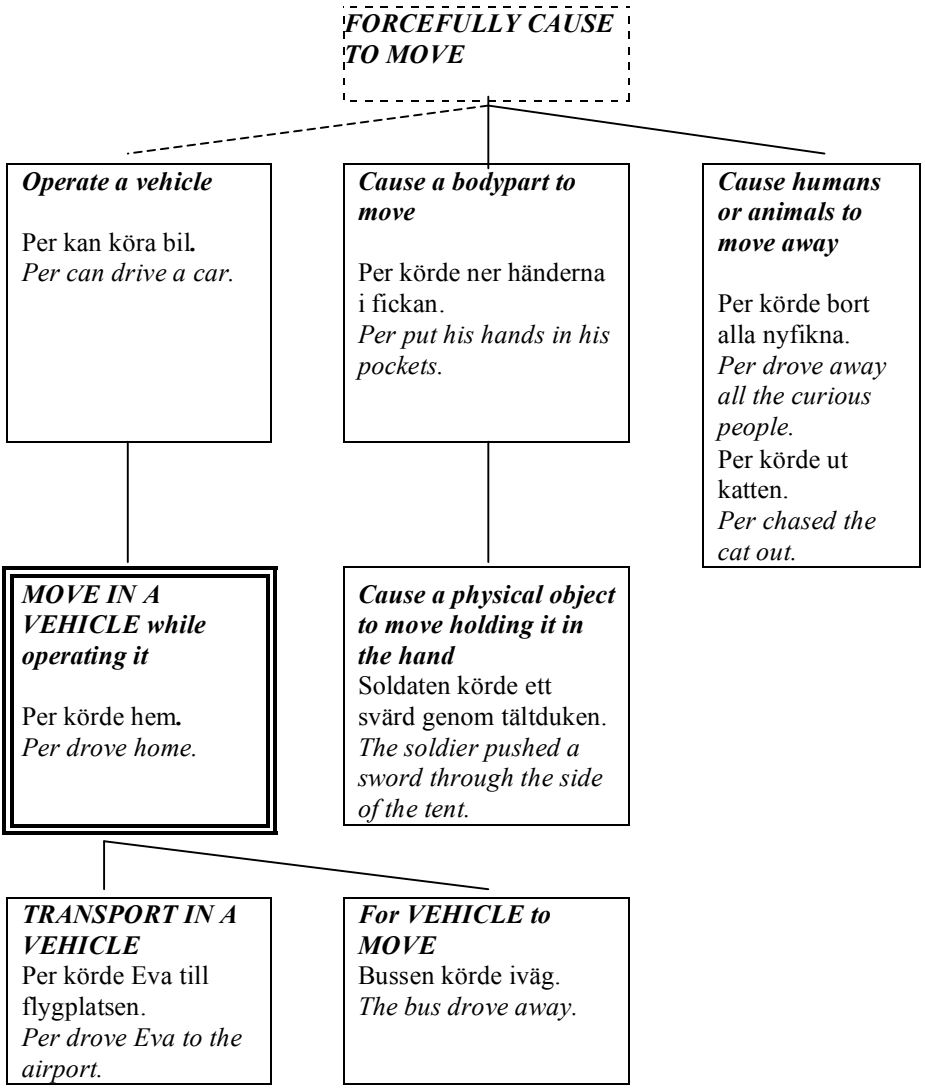

Figure 2: Meaning patterns of the verb *köra*.

3.2 The Cognates English *Drive* and Swedish *Driva*

As shown above, *drive* is the most frequent English translation of *köra*, which in its turn is the most frequent translation of *drive* in the ESPC. However, the relationship is not completely symmetrical. Even if *köra* is the most frequent equivalent of *drive*, there is a second equivalent that is relatively prominent, namely the Swedish cognate *driva*. In the ESPC, *köra*

accounts for 102 (53%) and *driva* for 35 (18%) of the equivalents of the 192 occurrences of *drive* in the English texts. Swedish *driva* expresses the application of force in various ways. The following example with a human object is parallel to example (18) with *köra*:

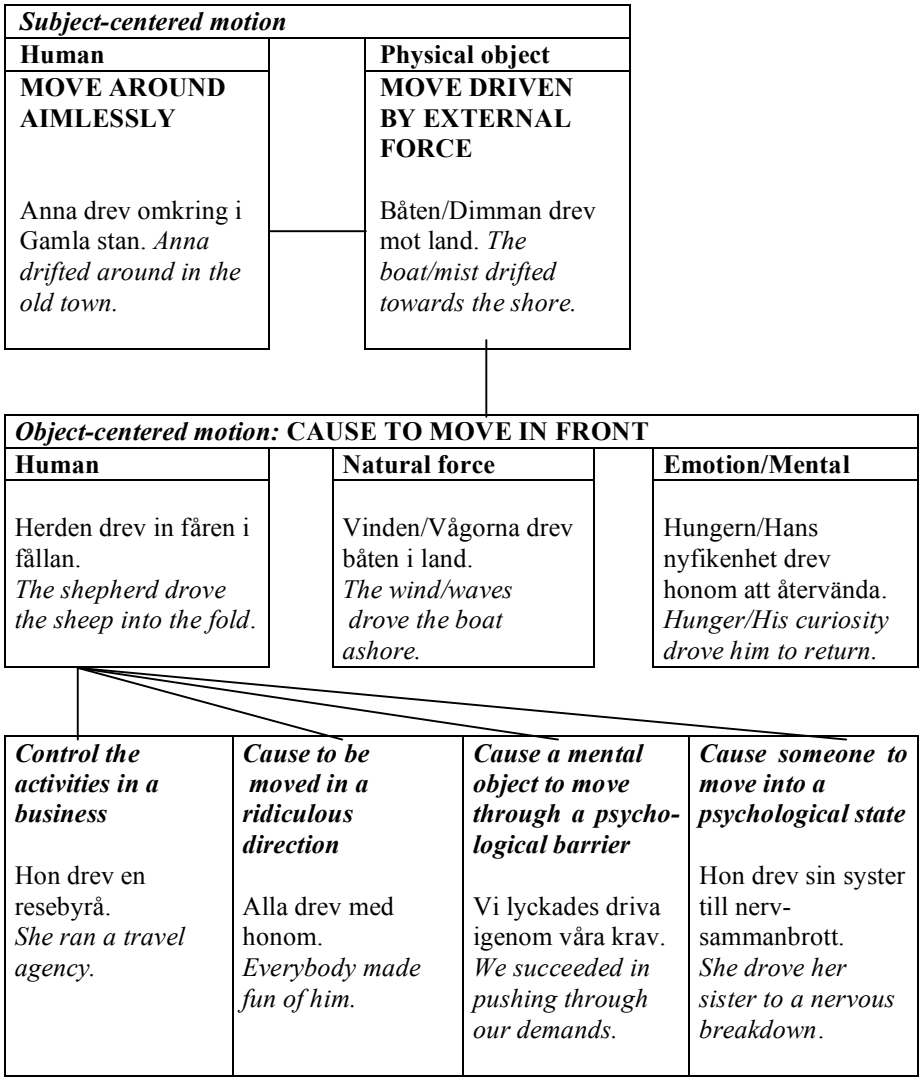

Figure 3: The polysemy of Swedish *driva.*

(19) Swedish: Utan förskoning ***drev*** hon ut dem i snön, KE
English: She ***drove*** them out into the snow without mercy,
German: Ohne Erbarmen ***trieb*** sie alle hinaus in den Schnee
French: Sans se laisser apitoyer, elle les ***fit sortir*** dans la neige
Finnish: Loviisa ***ajoi*** ilman armoa ryövärit hankeen,

Swedish *driva* can be used with both human and more abstract subjects as demonstrated in figure 3, which accounts for the major meanings of *driva*. A relatively common subject of *driva* is a Natural force such as Wind or Waves, which typically can drive a ship or other Natural phenomena such as clouds and mist. In Novels 81, examples are found such as (somewhat shortened): *Vinden drev båten långsamt framåt* 'The wind drove the boat slowly forwards' and *Vinden drev dammet längs gatorna* 'The wind drove the dust along the streets'. Related to this use is (20) from a popular science text (ESPC):

(20) Denna självreglerande process sker aktivt och ***drivs*** av den fria energi som kommer via solljuset.
The self-regulation of the system is an active process ***driven*** by the free energy available from sunlight. JL

Both English *drive* and its Swedish cognate can also be used about psychological forces causing mental processes as in (21).

(21) Vad jag tror är att Byrons inre främling var lynnigare än min och att den ***drev*** honom till större överdrifter och större ånger. SCO
What I think is that Byron's inner stranger was more capricious than mine, and ***drove*** him to greater exaggerations and greater regrets.

With a human subject, *driva* can be used with both concrete and abstract objects, for example a business or enterprise as in (22)

(22) Tillsammans med min hustru Louise ***driver*** jag Åkerbloms Fastighetsförmedling. HM2
I ***run*** Åkerblom's Real Estate with my wife.

Swedish *driva* can also be used intransitively with the moving Theme as subject as in (23) and (24). In this use, *driva* describes non-self propelled motion or motion without a clear aim. The most direct equivalent appears to be *drift* in English. Similar to many of the motion-in-vehicle verbs, *driva* alternates its meaning from exerting force (in its transitive uses) to lack of control (in its intransitive uses).

(23) Jag hade drömt att Rustica ***drev*** redlös med tidvattnet mot Stromas klippor. BL
I had dreamt that Rustica, disabled, was ***drifting*** with the tide towards the cliffs of Stroma.

(24) Zigenarna, som de kallar 'los egipcianos', är förbjudna att i fortsättningen ***driva*** omkring i riket. BTC
The gypsies, whom they call "los egipcianos", will in the future be forbidden to ***wander about*** in the kingdom.

It has only been possible to exemplify a few of the many different uses of *driva*. Figure 3 does not provide a complete account either but should suffice to show that the various meanings are interrelated in a systematic (but not completely predictable) way.

4 The Swedish Verbs of Travelling: *Åka*, *Fara*, *Resa*

As noted above, the Swedish verb *gå* cannot be used with a human subject when a vehicle is involved, since *gå* implies that the motion takes place on foot. There are basically three alternatives apart from verbs that incorporate a specific vehicle, namely the verbs *åka*, *fara* and *resa* which will be considered in this section. The major translations of these verbs in the Multilingual Pilot Corpus (MPC) are shown in table 3, which will be commented on below.

4.1 The Verb *Åka* and Non-Self-Propelled Motion

According to Buck (1949), Swedish *åka* is a late reflex of an Indo-European root *aĝ- with the primary meaning 'drive', which is reflected in Greek αγω 'lead' and Latin *agere* 'drive, carry on, act, do'. In Medieval Swedish, the verb *aka* covered both driving and traveling (Söderwall gives two senses: 1) *köra, föra på vagn eller i släde;* 2) *åka*). In present-day Swedish the verb is restricted to traveling. Thus, there is a development from 'driving' to 'traveling'.

Today, *åka* is the dominant expression for 'traveling in a vehicle' in Swedish. The MPC languages all have a verb with a corresponding meaning but the degree to which such a verb is used varies a great deal, as can be observed in table 3. In English, the most frequent translation is the nuclear motion verb *go*, whereas the most frequent verb indicating the means of transportation is *drive*, which more directly corresponds to *köra*.

The English verb that most directly expresses the meaning that the subject is being transported by some means of transportation is *ride* as in examples (9) and (10) in section 2. One characteristic of *åka* is that it can be used as a verb of departure as in (25).

(25) Swedish: Gudrun hade satt fram kaffebröd innan hon ***åkte***. KE
English: Gudrun had put out bunloaf before she had ***left***.
German: Gudrun hatte Kaffeegebäck hingestellt, bevor sie ***gefahren war***.
French: Gudrun avait sorti des petits gâteaux avant de ***partir***.
Finnish: Gudrun oli pannut pöydälle kahvileipää ennen kuin ***lähti***.

In (25), *hon åkte* 'she traveled', which does not have any explicit specification of Place, implicitly indicates 'from here' (i.e. the focus place) similar to expressions like *she left* (or *she went*). The other languages (except German) in this case use a verb of departure. As can be observed in table 3, this tendency is most pronounced in Finnish, where *lähteä* 'leave' is actually

The prototypical meaning: **TRAVEL IN A VEHICLE**								
Translation into	*Type of motion verb used as translation*							
	Nuclear 'go'/'come'		***Departure*** 'leave'		***Directional***		***Means of transportation***	
åka N								73
English	go come	31 4	leave	10	return	1	drive travel	11 2
German	gehen komme n	2 1					fahren fliegen reisen	57 1 1
French	aller venir	9 5	partir s'en aller	15 4	rentrer retourner descendre passer	6 3 3 3	conduire voyager	1 1
Finnish	mennä tulla	9 2	lähteä	32			ajaa matkustaa	15 7
fara N								33
English	go	11	leave	3	cross	2	drive travel	11 3
German							fahren reisen	27 1
French			partir	5	Various	8	voyager	1
Finnish	mennä	1	lähteä	6	ylittää	1	ajaa matkustaa	12 4
resa N								18
English	go	6	leave	2			travel drive	7 1
German							reisen fahren fliegen	10 4 1
French	venir	1	partir s'en aller	8 2	parcourir	1	voyager	3
Finnish			lähteä	1			matkustaa	9

Table 3: Major translations of the Swedish verbs *åka*, *fara* and *resa* in the Multilingual Pilot Corpus (MPC) used as verbs of traveling.

the most frequent translation of *åka*. Particularly in French, there is a tendency to use a directional verb unmarked for any type of manner such as *descendre* 'move down' and *passer* 'pass'. In spite of the fact that there exist verbs indicating means of transportation in all the MPC languages, there are remarkable differences with respect to the frequency with which they are used as translations of *åka*. French only in two cases uses such a verb. German, on the other hand, with very few exceptions uses *fahren* as a translation of *åka*. Finnish uses *ajaa* 'drive, travel' to a certain extent but the verb of departure *lähteä* 'leave' is actually the most frequent translation. In English, the most frequent translation is the nuclear verb *go*. The nuclear verb meaning 'go' is used relatively frequently as a translation also in French and Finnish but not to the same extent. As shown in Viberg (2006a), the use of *åka* with reference to motion in a vehicle is clearly dominant in Swedish, accounting for close to 90% of the occurrences in novels, for example. Still, *åka* has an interesting pattern of polysemy which is shown in figure 4. All the uses of *åka* share the schematic meaning non-self-propelled motion. This is true even when the subject is non-human as in (26). A verb in the passive or (in French) reflexive form appears as a translation.

(26) Swedish: fönstren åker upp, MA
English: the windows are flung open
German: die Fenster werden aufgerissen
French: les fenêtres s'ouvrent,
Finnish: ikkuna avataan

Many of the uses of *åka* imply a lack of control on the part of the subject, which is closely related to the notion non-self-propelled motion. This is evident when the subject refers to a body part as in (27).

(27) Swedish: Ibland gapade han lätt och ögonlocken ***åkte ner***. KE
English: his mouth occasionally dropping open and his eyelids ***drooping***.
German: Manchmal gähnte er flüchtig, und die Augendeckel ***fielen*** ihm ***zu***.
French: Par moments, il bâillait un peu et ses paupières ***se fermaient***.
Finnish: Häntä haukotutti ja hänen silmä-luomensa ***lurpahtelivat***

The notion of lack of control is even more prominent in a set of partly idiomatic phrases sharing the meaning 'End up in an unpleasant situation' such as *åka dit* 'be caught (by the police)' shown in (28).

(28) Swedish: Vi skulle ***åka dit*** direkt. MA
English: We'd be ***locked*** right ***up***.
German: Wir sollten direkt ***hinfahren***.
French: On serait ***coincées*** illico.
Finnish: Me ***jäisimme*** heti ***kiinni***.

There does not seem to be any general equivalent to *åka* used in this sense. These uses of *åka* are very colorful and it seems that translators, correctly, concentrate on rendering the correct associations of such expressive uses of words. (See Viberg 2006a for more examples of English translations in ESPC) .

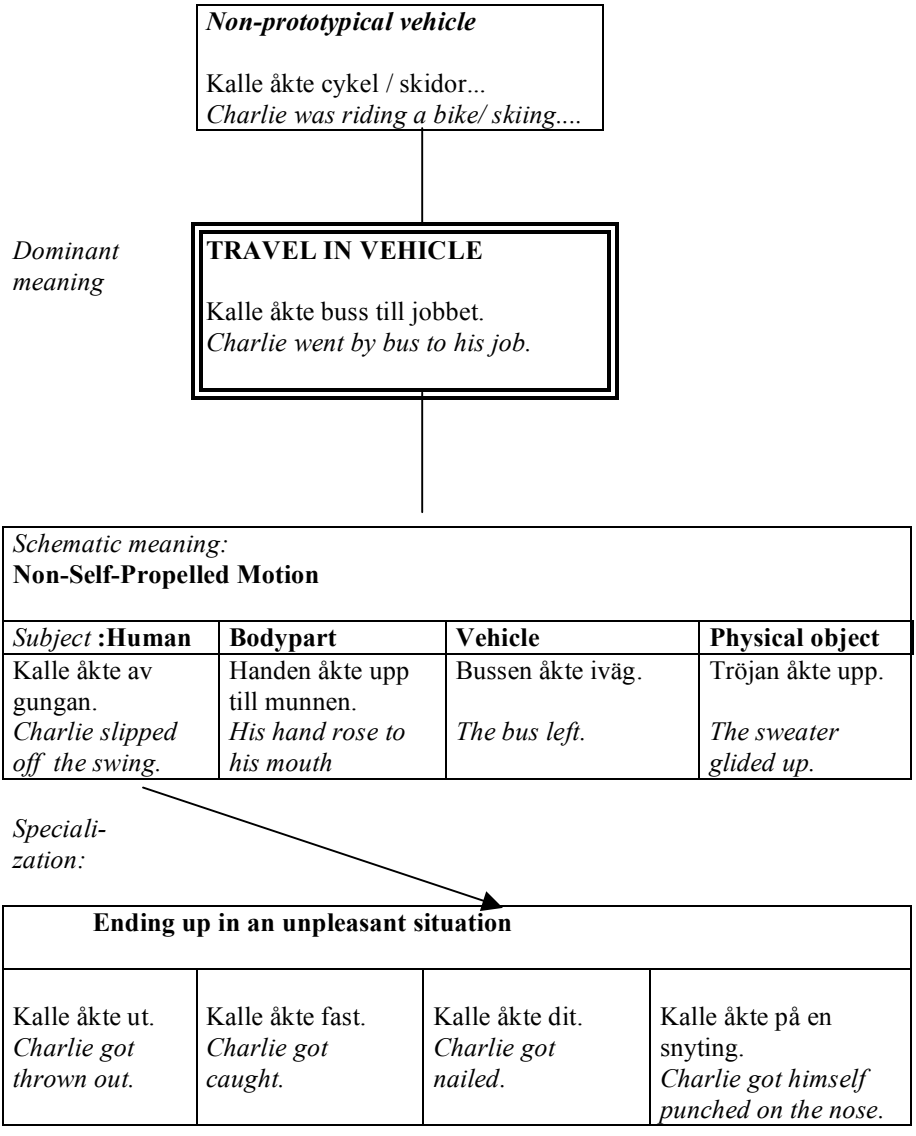

Figure 4: The meaning pattern of *åka.*

The schematic meaning non-self-propelled motion is shared by practically all uses of *åka*. The exceptions are expressions such as *åka skidor* 'ski' and *åka skridskor* 'skate', where the means of transportation require bodily

activity. However, the sense of lack of control is totally absent when *åka* refers to motion in a vehicle. In the majority of cases, traveling in a vehicle is an intentional act. In this respect, the more specialized uses represent a further step away from the historically more active basic meaning ('drive' etc.)

4.2 The Verb *Fara* and Rapid, Violent Motion

The Swedish verb *fara* is a verb with cognates in most of the modern Germanic languages, which originally had a very general meaning as a motion verb. The verb *fara* is still the most frequent motion verb in modern Icelandic (Pind 1991) and the same applies to one the most conservative Swedish dialects Elfdalian (Älvdalska), where *fårå* 'go, travel, move' is the most frequent motion verb, slightly more frequent than *kumå* 'come' (Steensland 1986). In modern standard Swedish, it can be used as an alternative to *åka* but is less frequent in this use and is perceived as somewhat formal and/or old-fashioned by many speakers, although there is variation on this point since the verb appears to be in more general use in certain regional varieties of Swedish. The following account is based on printed texts. In its use as a motion-in-vehicle verb as in (29) it has a translation pattern that is similar to that of *åka* as can be observed in table 3, which accounts only for this use of *fara*.

(29) Swedish: Wallander ***for*** hem. HM
English: Wallander ***drove*** home.
German: Wallander ***fuhr*** nach Hause.
French: Puis il reprit sa voiture et ***rentra chez lui***,
Finnish: Wallander ***ajoi*** kotiin.

In addition to this use, *fara* has a number of other uses which form a rather complex pattern of polysemy shown in figure 5. Several of these specialized meanings are realized as phraseological units of various types with idiosyncratic semantic and/or formal properties which are more or less lexicalized and will not be described in detail but they all share the schematic meaning 'travel fast and/or violently and without (full) control'. With a human subject, the motion may be self-propelled as in (30) and (31), but lack (or low degree) of control is usually prominent.

(30) Swedish: Jävlar! Han ***for*** upp ur stolen, PCJ
English: Heck! He ***leaped*** out of the chair –
German: Verdammt! Er ***sprang*** auf,
French: Merde! Il ***sauta*** hors de la chaise.
Finnish: Hitto! Hän ***hyppäsi*** tuolista,

(31) Swedish: Runt i förrådet ***for*** han och hotade, MN
English: He ***charged*** round and round the shed, growling out threats:
German: Er ***lief*** im Schuppen herum und stieß wilde Drohungen aus,

French: Il ***parcourait*** la réserve en proférant des menaces,
Finnish: Hän ***kolusi*** varaston laidasta laitaan ja uhosi,

With a bodypart as subject, the use of *fara* describes an uncontrolled and rapid movement of the limb:

(32) ”Ärrnäsan” ryckte till som om han ertappats med något olagligt och högerhanden ***for*** in under slängkappan i en van gest. ARP Fastighetsförmedling. HM2
‘Scarnose’ jumped as if he had been caught out doing something illegal and, out of habit, his right hand ***dived*** in under the cloak.

A variety of non-concrete subjects are also allowed. However, the progression from concrete subjects that can be perceived by the external senses to more abstract subjects is continuous and does not justify the postulation of metaphorical shifts between incompatible domains. Certain emotive, physiological reactions can be clearly localized spatially as in (33).

(33) En rysning ***for*** längs ryggraden.
A shudder ***ran*** down his backbone.

Subjects belonging to the semantic class Verbal communication such as ‘word’, direct quotes etc can be conceived of both as mental objects and as sounds which at least have sources (and partly also targets or receivers) that can be located in physical space. All the MPC languages use motion verbs as translations of fara (34) but the basic meaning varies a great deal.

(34) Swedish: Svordomarna ***for*** ur mun på henne, MF
English: oaths ***pouring*** out of her mouth,
German: Sie ***stieß*** Flüche aus,
French: Les jurons ***s'envolaient*** de sa bouche,
Finnish: Hänen suustaan ***pulppusi*** kirosanoja,

The step is therefore rather short to purely mental subjects such as in (35).

(35) Swedish: och för ett ögonblick ***for*** tanken att han kunde ha rätt genom hennes huvud. MF
English: and the thought that he might be right ***raced*** through her mind.
German: und für einen Augenblick ***schoß*** ihr der Gedanke durch den Kopf, daß er recht haben könnte.
French: espace d'un instant, la pensée qu'il pouvait avoir raison lui ***traversa*** l'esprit.
Finnish: ja hetken Hanna ***ajatteli,*** että hän saattoi olla oikeassa.

In metaphorical uses such as in (35), where the subject refers to a mental concept such as ‘thought’, the association to violence or rapidity is still

present and it is also reflected in the translations except for in Finnish, where *ajatella* 'think' is used.

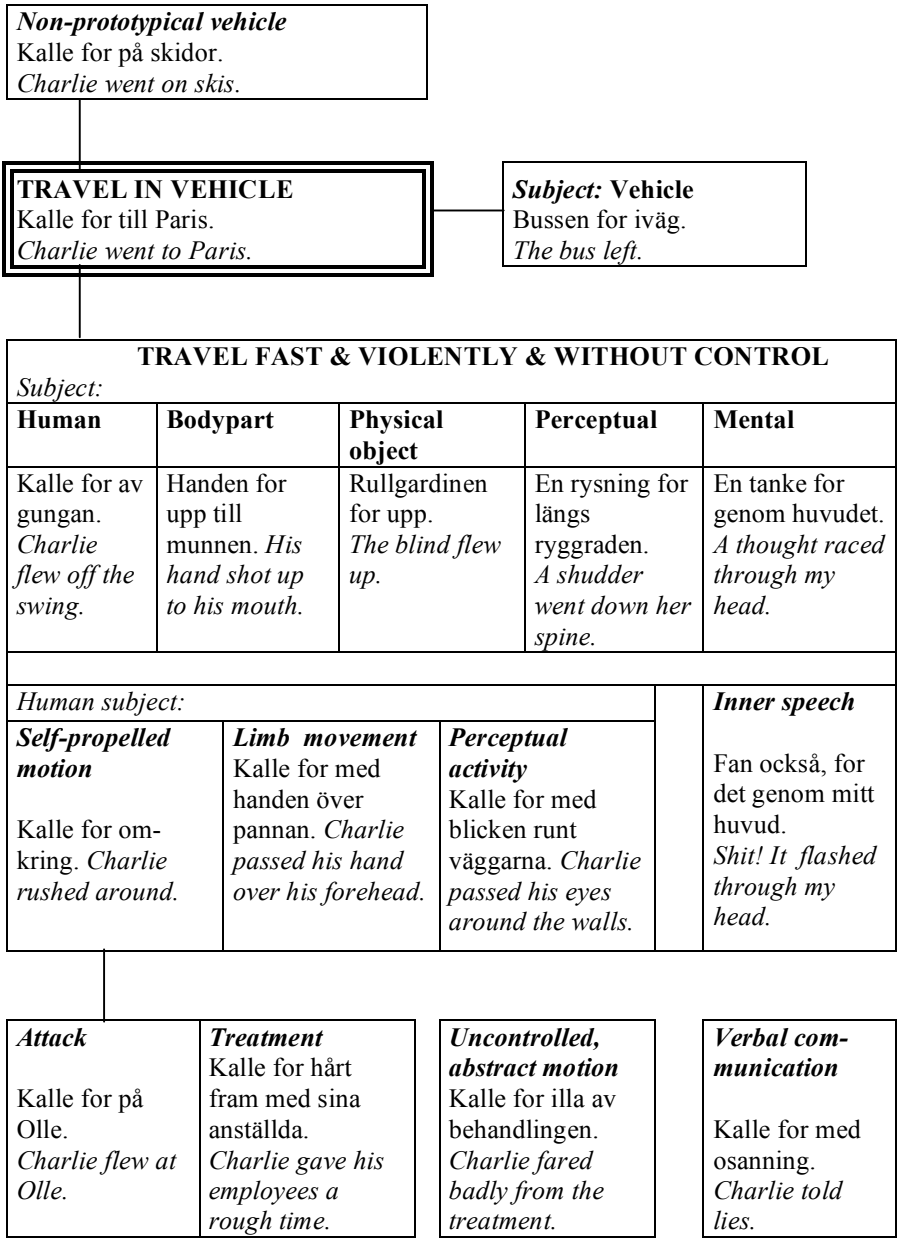

Figure 5: The meaning pattern of *fara*.

With a human subject, the verb has a number of specialized meanings which are given at the bottom of the figure (without any intention of showing in

detail how these meanings are interrelated.) The majority of these meanings refer to various types of self-propelled but relatively uncontrolled and violent actions such as: *Kalle for på Olle*. 'Charlie flew at Olle' and somewhat more abstractly: *Kalle for hårt fram med sina anställda*. 'Charlie gave his employees a rough time.' There is also a metaphorically based expression shown in (36) where the human subject is completely out of control: *fara väl/illa av*. This is one of the few cases where English can use the cognate verb *fare well/badly from*, although this is a slightly archaic use in English.

(36) Då begrep Mattis att den stackarn inte ***for väl av*** för mycket larm, och han teg motvilligt. AL
At last Matt realized that too much noise would not ***do*** the poor fellow any ***good***, and he fell into unwilling silence.

The metaphor is a variation on the theme LIFE IS A JOURNEY and the event structure metaphor (Lakoff 1993). What the phrase *fara väl/illa* conjures up is someone drifting through life meeting good or bad things without being able to control the flow of events. In the translation, the human being is even formally the object that 'the noise' would 'do' something to.

The frequencies of the various uses of *fara* in the tagged version of the Parole corpus are shown in table 4. Travel in Vehicle is dominant and accounts for roughly 55%. This is parallel to *åka* except that the dominance was more pronounced for that verb, 89%. The verb *fara* actually has a more complex meaning pattern than *åka* and the analysis presented here is not exhaustive.

Historically, the meaning pattern of *fara* in present-day Swedish is the result of specialization. Once *fara* had a more general meaning as a motion verb, but today the meaning 'travel in a vehicle' is clearly dominant. The rest of the uses of the verb to a great extent form a large set of phraseological units that share a schematic meaning 'travel fast & violently & without control' which is also metaphorically extended to various mental and other abstract phenomena.

Human subject:		
Travel in Vehicle	592	55%
Other Human Motion	86	
Verbal communication	17	
*Fara illa(/väl)**	100	
Non-human subject		
Bodypart	18	
Vehicle as Subject	36	
Other Concrete Subject	76	
Mental subject	32	
Various Other Cases	110	
TOTAL	1067	

Table 4: Uses of *fara* in the Swedish Parole corpus. (*A few of these have a non-human subject.)

4.3 The Verb *Resa*

The verb *resa* is primarily used about a journey covering a relatively long distance and/or requiring special preparations. Its most direct equivalent in English is the verb *travel*, but the nuclear verb *go* is also quite frequent as a translation for this meaning. As can be observed in (37), all the MPC languages have at least one special verb that corresponds relatively directly to *resa* and this is the most frequent translation of *resa* as a verb of traveling except in French according to table 3. In French, verbs of departure dominate as translations. (There are, however, only 20 occurrences of *resa* as a verb of traveling in the MPC.)

(37) Swedish: Han ***reser*** runt i världen HM
English: He ***travels*** all over the world
German: Er ***reist*** in der ganzen Welt umher
French: Il ***voyage*** dans le monde entier
Finnish: Hän ***matkustelee*** ympäri maailmaa

What is notable about *resa* is that it has two prototypes that synchronically seem to be completely unrelated apart from referring to motion. As an intransitive verb it means 'travel' as we have just seen. However, used as a transitive verb, *resa* means 'raise', 'cause to go up', 'cause to stand up'.

(38) Sedan tog han Robert Åkerblom i armen och ***reste honom upp***. HM2
Then he took Robert Åkerblom by the arm and ***helped him to his feet***.

This meaning is closely related to the meaning of the reflexive form of this verb *resa sig*, which means 'stand up', 'rise' and is very frequent as a

postural verb. In many cases it is interchangeable with *ställa sig up* 'stand up' but it may also be used about rising from a lying to a sitting position as an alternative to Swedish *sätta sig upp* 'sit up'. The reflexive forms of *resa* are usually translated by one of the expressions *rise*, *get up* or *stand up* in English and to the corresponding expressions in the other languages.

(39) Swedish: Johan ***reste sig***. KE
English: Johan ***got up***.
German: Johan ***stand auf***.
French: Johan ***se leva***.
Finnish: Johan ***nousi***.

The use as a postural verb is actually the most frequent use in present-day Swedish, much more frequent than the transitive form and than *resa* as a verb of traveling. The postural meaning can be extended to inanimate subjects and then describes apparent motion as in (40).

(40) Swedish: Det väldiga matsalsbordet ***reser sig*** över mitt huvud, IB
English: The huge dining-room table ***towers*** above me
German: Der gewaltige Eßzimmertisch ***erhebt sich*** über meinem Kopf,
French: L'immense table de la salle à manger ***s'élève*** au-dessus de ma tête,
Finnish: Ruokasalin valtava pöytä ***kohoaa*** pääni yläpuolelle,

Transitive *resa* can be extended to building (*resa ett tempel* 'erect a temple') and related types of production as in (41).

(41) Vid Gamla Uppsala och Vendel ***restes*** högar över bygdekungar, medan andra jordades i sina skepp.AA
At Gamla Uppsala and Vendel mighty burial mounds ***were raised*** over the local kings, while others were interred in their ships.

No closer semantic relationship, however, is perceived between *resa* as a verb of traveling and the other meanings of *resa*. One possibility would be to regard this as a case of homonymy, but there are no formal signs of this since all the inflected forms are shared. Whatever solution is chosen with respect to homonymy, it is justified (at least synchronically) to count two unrelated prototypes for the meaning of *resa* as in figure 6.

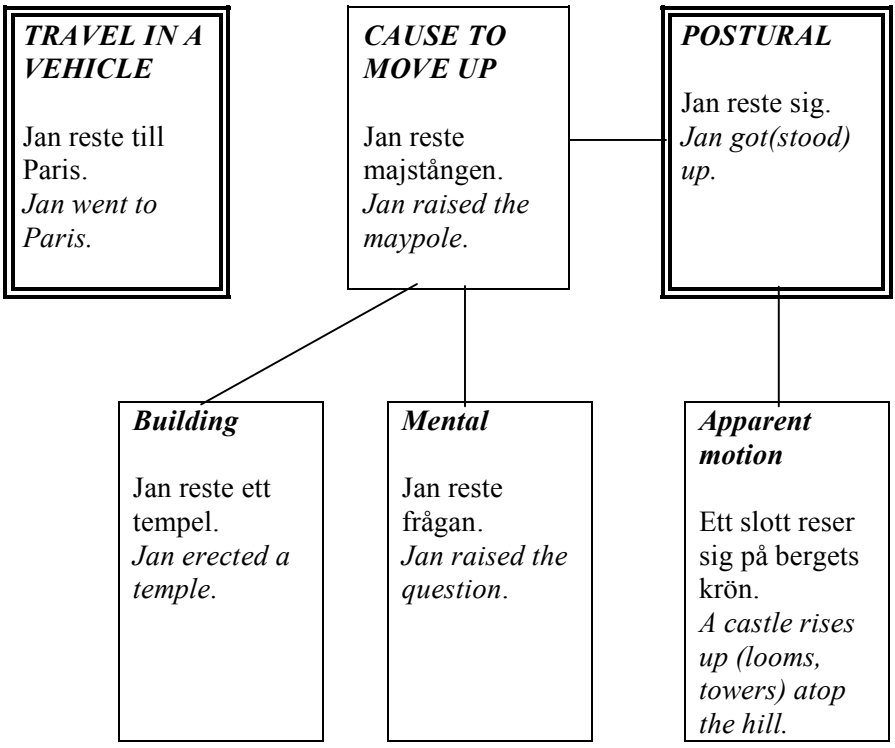

Figure 6: Split prototypes. Relationships between the major meanings of *resa* 'travel; and 'raise, rise'.

The transitive and reflexive forms of the verb have a number of secondary meanings only some of which are represented in the figure, since it would take us too far from the topic of this paper to account for these meanings in detail. With certain mental nouns as object, the meaning of the verb is 'move a mental object up to a conscious level' (a variation on the metaphor CONSCIOUS IS UP. See Lakoff & Johnson 1980, 15)

(42) Mannen, en framgångsrik, betrodd och uppmärksammad Herrens tjänare, ***reste*** onaturliga
The man, a successful and trusted servant of the Lord who people made much of, ***placed***

(43) När man letar utmärkande drag hos ett helt folk hittar man idel motsägelser, varje påstående ***reser opposition***.IU ***Lit.*** 'raises opposition'
Fumbling after characteristics
of a whole people one finds nothing but contradictions, every
statement ***calls up its opposite***.

Thus, as was the case with the other verbs of traveling *åka* and *fara*, the use of *resa* as a verb of traveling is isolated from the rest of the uses of the verb which form a semantic network. In this case, however, there is a complete split.

5 The Motion-in-Vehicle Verbs in Swedish FrameNet

The account so far has been focused on individual verbs. In this section, the place of the motion-in-vehicle verbs in a general model of the lexicon will be discussed based on FrameNet. Unless otherwise stated, only the meanings related to the use of the verbs as motion-in-vehicle verbs will be considered. According to the approach to semantics known as frame semantics (Fillmore 1985), verbs (and other relational words) evoke frames – schematic structures of recurring situations. This theory is the foundation of FrameNet, a comprehensive lexical semantic database of English (see Fillmore et al 2003 for a general description). The following is based on material from: http://framenet.icsi.berkeley.edu. The motion frame is defined in the following way: Some entity (Theme) starts out in one place (Source) and ends up in some other place (Goal), having covered some space between the two (Path). A simple example would be: *Peter* (Theme) *fell off the roof* (Source). The frames that inherit the general Motion frame add some elaboration to this simple idea. Inheriting frames can add Goal-profiling (*arrive, reach*), Source-profiling (*leave, depart*), or Path-profiling (*traverse, cross*), or aspects of the manner of motion (*run, jog*). Source-profiling is characteristic of what was referred to above as the verbs of departure.

Another elaboration is represented by the frame Self_motion in which the central frame element is the Self_mover , a living being which moves under its own power in a directed fashion, i.e. along what could be described as a Path , with no separate vehicle. A typical example in Swedish would be *Per* (Self-mover) *gick till stationen* (Goal) 'Per walked to the station'. In Swedish, the subject of *gå* in contrast to English *go* always is a Self-mover, if it refers to a human.

FrameNet provides an interface to the syntactic realization such as the argument structure of verbs. This is shown for *köra* 'drive' in table 5. In its basic meaning, *köra* evokes the frame Operate_vehicle. The words in this frame describe motion involving a Vehicle and someone who controls it, the Driver. Some words normally allow the Vehicle to be expressed as a separate constituent. Example: Tim [Driver] DROVE his car [Vehicle] all the way across North America [Path]. The syntactic realization of the frame elements can be described with respect to Phrase types (NP, PP...) and grammatical functions (subject, object, adjunct…). As can be observed in table 5, the subject slot can be filled by the frame elements Driver or and Vehicle. (The passive allows further options but these follow from general syntactic rules.) The object slot can be filled by Vehicle, Passenger or Cargo.

Grammatical Relations	Subject			Object	Adjuncts
Phrase Structure	NP	Verb	(Particle)	(NP)	(PP^n)
	Maria	körde	in	bilen	i garaget
	'Maria drove ('in') the car into the garage'				
Frame elements	Driver			Vehicle	Goal
	Maria	körde	hem	barnen	från skolan
	Lit. 'Maria drove home the kids from school''				
Frame elements	Driver			Passenger	Source
	Maria	körde	hem	möblerna	i sin volvo
	Lit. 'Maria drove home the furniture in her Volvo'				
Frame elements	Driver			Cargo	Vehicle
	Bilen	körde	ner		i diket.
	Lit. 'The car drove down into the ditch'				
	Vehicle				Goal

Table 5: Frame elements of *köra* and their syntactic realization.

Frame elements:

Driver: The being, typically human, that controls the Vehicle as it moves
Vehicle: The means of conveyance controlled by the Driver
Cargo/Passenger: The goods or people being moved by a Driver in a Vehicle

The frame elements evoked by the verbs *åka* and *resa* are shown in table 6. In FrameNet, there is a frame Ride_vehicle defined as "a Theme is moved by a Vehicle which is not directly under its power". When *åka* is used as a motion-in-vehicle verb, a human subject is normally interpreted as an intentional agent. For that reason, another frame element is preferred: Passenger defined as a person being moved by a Driver in a Vehicle. Theme is reserved for the case when the subject refers to a person who is totally out of control (or to an inanimate argument). A vehicle can be specified directly after the verb *åka* in Swedish. Since the vehicle in this position is realized as a bare noun, it can be regarded as a particle. When the noun is modified, it usually appears as an adjunct in a prepositional phrase (e.g. *i sin pappas bil* 'in his dad's car'). The verb *resa* evokes the Travel frame: „a Traveler goes on a journey, an activity, generally planned in advance". The vehicle can only be specified as an adjunct in a prepositional phrase with *med* 'with' (e.g. *med tåg* 'by train').

Phrase structure	NP	Verb	(Particle)	(PP^n)	
	Jan	åkte	bil	till skolan	
	Lit. 'Jan went (by) car to school''				
Frame elements	Passenger		Vehicle	Goal	
	Jan	åkte	av	gungan	
	'Jan flew off the swing'				
Frame elements	Theme			Source	
	Bilen	åkte	in	i tunneln	
	Lit. 'The car went in into the tunnel'				
Frame elements	Vehicle			Goal	
	Familjen	reste	upp	till fjällen	med buss.
	Lit. 'The family traveled up to the mountains by bus.'				
	Traveller			Goal	Vehicle

Table 6: Frame elements of *åka* and *resa* and their syntactic realization.

As shown in table 7, the verbs *åka*, *fara* and *resa* can all evoke the Departing frame defined as: „An object (the Theme) moves away from a Source".

NP	V	(Particle)
Subject		
Theme		
Ann	åkte	(iväg/bort)
	for	
	reste	
'Ann went (away)/left'		

Table 7: The Departing frame.

FrameNet is well suited to model the interface between semantic (conceptual) structure and syntax (in particular the argument structure). The account given in this section does not deal with phenomena that apply to motion verbs in general, in particular the wide range of frame elements evoked by the general motion frame such as Source, Path, Goal and Distance. In FrameNet such elements are inherited from the more general frame. Some general problems remain to be solved, in particular the

description of the interaction between motion verbs and spatial verbal particles (see Viberg 2007 for a brief discussion). As has been observed several times above, French tends to use directional verbs without any indication of manner, where Swedish uses a motion-in-vehicle verb in combination with a spatial particle.

6 Conclusion

The verbs describing motion in a vehicle form a (sub-)field in the larger semantic field of motion verbs. This paper has dealt with patterns of differentiation, i.e. field internal contrasts between the verbs in the field, and with patterns of polysemy of the individual verbs which extend into a number of other semantic fields.

The differentiation pattern was studied from a contrastive perspective. It is instructive to look at the differences that were found from two different angles: inventories and usage patterns. The comparison of the inventories describes the number of semantic contrasts that can be expressed. All the languages studied had verbs describing motion in a vehicle but varied with respect to the number of field internal contrasts that were lexicalized as different verbs. Thus, there were interesting differences even with respect to the inventories. The study of the usage patterns in the corpora, however, presented much more dramatic differences. As a translation of *åka* and *fara*, English tended to use the nuclear verb *go* which is completely unmarked for manner of motion, whereas Finnish favored a verb of departure (*lähteä* 'leave') and French primarily used directional verbs or verbs of departure. Such findings with respect to differences in usage patterns are among the key results of corpus-based contrastive studies. One limitation of the present study is that primarily one direction of translation was studied: from Swedish into other languages but it is not to be expected that these differences will completely disappear if the study is extended to include originals in the other languages translated into Swedish. (In other studies, this type of comparison has already been done especially for English/Swedish.)

The study of the patterns of polysemy of the Swedish motion-in-vehicle verbs pointed to an interesting general trend. In several cases, the semantic extension proceeded from power and control towards passivity and/or lack of control. Perhaps this finding can be related to the rapid change of the means of transportation that has taken place the last 150 years or so, from the control of domestic riding and draft animals with a will of their own to the use of more and more automated cars and modern passenger traffic. In old times, traveling tended to require greater amounts of control and effort (cf. French *travail* 'work', which is closer to the original meaning than its English cognate *travel*). More detailed comparisons with other languages are required to decide whether the development in Swedish represents a general trend or is more language specific. What is perfectly clear already from the present study is that there has been rapid semantic restructuring resulting in

differences between the closely related Germanic languages. Verbs like Dutch *rijden* and Danish *køre* still have a form that is very close to Swedish *rida* and *köra* in spite of their wide semantic extensions. One of the interesting results of the study of rather closely related languages is the observation that semantic change can proceed relatively rapidly.

References

Altenberg, B. and K. Aijmer (2000). The English-Swedish Parallel Corpus: A resource for contrastive research and translation studies. In C. Mair & M. Hundt (eds.) *Corpus Linguistics and Linguistic Theory*, 15–33. Rodopi.

Bohnemeyer, J. et al. (2007). Principles of event segmentation in language. The case of motion events. *Language* **83**, 495–532.

Buck, C. (1949). *A dictionary of selected synonyms in the principal Indo-European languages*. University of Chicago Press.

CIDE. *Cambridge International Dictionary of English*. Cambridge University Press. 1995.

Fillmore, C. (1985). Frames and the semantics of understanding. *Quaderni di semantica* **6**, 222–254.

Fillmore, C., C. R. Johnson and M. Petruck (2003). Background to FrameNet. In Th. Fontenelle (ed.) FrameNet and Frame Semantics. Special issue of *International Journal of Lexicography* **16**, 231–366.

Koptjevskaja-Tamm, M., D. Divjak, and E. Rakhlina (forthc.) Aquamotion verbs in Slavic and Germanic: a case study in lexical typology.

Lakoff, G. (1993). The contemporary theory of metaphor. In A. Ortony (ed.) *Metaphor and Thought*. 2nd ed. Cambridge University Press.

Lakoff, G. and M. Johnson (1980). *Metaphors we live by*. University of Chicago Press.

Pind, J. (ed.) (1991). *Íslensk Orðtíðnibók*. Orðabók Háskólans.

Spelke, E., A. Phillips and A. Woodward (1995). Infants knowledge of object motion and human action. In D. Sperber, D. Premack and A. Premack (eds.) *Causal cognition. A multidisciplinary debate*. Oxford University Press.

Steensland, L. (1986). *Liten älvdalsk-svensk och svensk-älvdalsk ordbok*. Älvdalen.

Talmy, L. (1985). Lexicalization patterns: semantic structure in lexical forms. In T. Shopen (ed.), *Language typology and syntactic description III. Grammatical categories and the lexicon,* 57–149. Cambridge University Press.

Viberg, Å. (1981). Svenska som främmande språk för vuxna. In K. Hyltenstam (ed.) *Språkmöte*, 21–65. Liber Läromedel.

Viberg, Å. (1992). Universellt och språkspecifikt i det svenska ordförrådets organisation. *Tijdschrift voor Skandinavistiek 13:2*, 17–58

Viberg, Å. (1999a). The polysemous cognates Swedish *gå* and English *go*. Universal and language-specific characteristics. *Languages in Contrast* **2**, 89–115.

Viberg, Å. (1999b). Polysemy and differentiation in the lexicon. Verbs of physical contact in Swedish. In J. Allwood, and P. Gärdenfors (eds.) *Cognitive semantics. Meaning and cognition,* 87–129. Benjamins,

Viberg, Å. (2006a). Crosslinguistic lexicology and the lexical profile of Swedish. In C. Bardel & J. Nystedt (eds) Progetto dizionario italiano-svedese. Atti del primo colloquio. *Acta Universitatis Stockholmiensis. Romanica Stockholmiensia* **22**, 79–118.

Viberg, Å. (2006b). Towards a lexical profile of the Swedish verb lexicon. In Viberg, Å. (guest ed.) The Typological Profile of Swedish. Thematic issue of *Sprachtypologie und Universalienforschung. Vol.* **59***:1*, 103–129.

Viberg, Å. (2007). Wordnets, Framenets and Corpus-based Contrastive Lexicology. In P. Nugues and R. Johansson (eds.) *Frame 2007: Building Frame Semantics Resources for Scandinavian and Baltic Languages*, 1–10.

Viberg, Å., K. Ballardini & S. Stjernlöf (1984) *A concise Swedish grammar.* Stockholm: Natur & Kultur.

Electronic sources

ESPC. The English Swedish Parallel Corpus. For a description, see: http://www.englund.lu.se/content/view/66/127/

Söderwall. Dictionary of Swedish medieval language. (See next source!)

Swedish Language Bank (Språkbanken) http://spraakbanken.gu.se/.

OED. *Oxford English Dictionary*. Electronic version: http://www.oed.com/

The Automatic Translation of Film Subtitles
A Machine Translation Success Story?

Martin Volk

Stockholm University
Department of Linguistics

University of Zurich
Institute of Computational Linguistics

1 Introduction

Every so often one hears the complaint that 50 years of research in Machine Translation (MT) has not resulted in much progress, and that current MT systems are still unsatisfactory. A closer look reveals that web-based general-purpose MT systems are used by thousands of users every day. And, on the other hand, special-purpose MT systems have been in long-standing use and work successfully in particular domains or for specific companies.

This paper investigates whether the automatic translation of film subtitles can be considered a machine translation success story. We describe various projects on MT of film subtitles and contrast them to our own project in this area. We argue that the text genre "film subtitles" is well suited for MT, in particular for Statistical MT. But before we look at the translation of film subtitles let us retrace some other MT success stories.

Hutchins (1999) lists a number of successful MT systems. Amongst them is Météo, a system for translating Canadian weather reports between English and French which is probably the most quoted MT system in practical use. References to Météo usually remind us that this is a "highly constrained sub-language system". On the other hand there are general purpose but customer-specific MT systems like the English to Spanish MT system at the Pan American Health Organization or the PaTrans system which Hutchins (1999) calls "possibly the best known success story for custom-built MT". PaTrans was developed for LingTech A/S to translate English patents into Danish.

Earlier Whitelock and Kilby (1995) had called the METAL system "a success story in the development of MT" (p. 198). METAL is mentioned as "successfully used at a number of European companies" (by that time this meant a few dozen installations in industry, trade or banking). During the same time the European Union has been successfully using a customized version of Systran for its translation service but also later for online access by all its employees.

Broad coverage systems like METAL and Systran have always resulted in a translation quality that required post-editing before publications.

Attempts to curb the post-editing by pre-editing or constraining the source documents have gone under the name of controlled language MT. Hutchins (1999) mentions controlled language MT (e.g. at the Caterpillar company) as an example of successful employment of MT. This is an area where part of the pioneering work was done at Uppsala University by Anna Sågvall Hein and her group (Almqvist and Sågvall Hein, 1996), including the development of controlled Swedish for the automobile industry. This research subsequently led to a competitive MT system for translating from Swedish to English (Sågvall Hein et al., 2002).

The claim that web-based machine translation is a success is based on the fact that it is used by large numbers of users. Critics do not subscribe to this argument as long as the translation quality is questionable. Still, popular services including Systran (www.systran.co.uk with 14 source languages) and Google (www.google.com/translate_t with 21 language pairs) cover major Western languages like English, Spanish and French, but also Arabic and Chinese. On the other hand there are providers that have successfully occupied niche language pairs like Danish to English (Bick, 2007).

So we see that MT success stories vary considerably. We regard the following criteria as the main indicators of success:

1. A large user base (this criterion is used in web-based MT services for the general public)

2. Customer satisfaction (this criterion is used in customer-specific MT systems and usually based on improved productivity and return on investment)

3. Long-term usage of the MT system

We will check which of these criteria apply to the automatic translation of film subtitles.

2 Characteristics of Film Subtitles

When films are shown to audiences in language environments that differ from the language spoken in the film, then some form of translation is required. Larger markets like Germany and France typically use dubbing of foreign films so that it seems that the actors are speaking the local language. Smaller countries often use subtitles. Pedersen (2007) discusses the advantages and drawbacks of both methods.

Foreign films and series shown in Scandinavian TV are usually subtitled rather than dubbed. Therefore the demand for Swedish, Danish, Norwegian and Finnish subtitles is high. These subtitles are meant for the general public

in contrast to subtitles that are specific for the hearing impaired which often include descriptions of sounds, noises and music. Subtitles also differ with respect to whether they are produced online (e.g. in live talkshows or sport reports) or offline (e.g. for pre-produced series). This paper focuses on general public subtitles that are produced offline.

In our machine translation project, we use a parallel corpus of Swedish, Danish and Norwegian subtitles. The subtitles in this corpus are limited to 37 characters per line and usually to two lines.[1] Depending on their length, they are shown on screen between 2 and 8 seconds. Subtitles typically consist of one or two short sentences with an average number of 10 tokens per subtitle in our corpus. Sometimes a sentence spans more than one subtitle. It is then ended with a hyphen and resumed with a hyphen at the beginning of the next subtitle. This occurs about 35.7 times for each 1000 subtitles in our corpus.

Example 1 shows a human-translated pair of subtitles that are close translation correspondences although the Danish translator has decided to break the two sentences of the Swedish subtitle into three sentences.[2]

(1) SV: Det är slut, vi hade förfest här. Jätten drack upp allt.
DA: Den er væk. Vi holdt en forfest. Kæmpen drak alt.
EN: *It is gone. We had a pre-party here. The giant drank it all.*

In contrast, the pair in 2 exemplifies a slightly different wording chosen by the Danish translator.

(2) SV: Där ser man vad framgång kan göra med en ung person.
DA: Der ser man, hvordan succes ødelægger et ungt menneske.
EN: *There you see, what success can do to a young person / how success destroys a young person.*

This paper can only give a rough characterization of subtitles. A more comprehensive description of the linguistic properties of subtitles can be found in (de Linde and Kay, 1999). Gottlieb (2001) and Pedersen (2007) describe the peculiarities of subtitling in Scandinavia.

3 Approaches to the Automatic Translation of Film Subtitles

In this section we describe other projects on the automatic translation of subtitles. We distinguish between rule-based, example-based, and statistical approaches.

[1] Although we are working on both Swedish to Danish and Swedish to Norwegian MT of subtitles, this paper focuses on translation from Swedish to Danish. The issues for Swedish to Norwegian are the same to a large extent.

[2] In this example and in all subsequent subtitle examples the English translations were added by the author.

3.1 Rule-based MT of Film Subtitles

Popowich et al. (2000) provide a detailed account of a MT system tailored towards the translation of English subtitles into Spanish. Their approach is based on a MT paradigm which relies heavily on lexical resources but is otherwise similar to the transfer-based approach. A unification-based parser analyzes the input sentence (including proper-name recognition), followed by the lexical transfer which provides the input for the generation process in the target language (including word selection and correct inflection).

Popowich et al. (2000) mention that the subtitle domain has certain advantages for MT. According to them it is advantageous that output subtitles can and should be grammatical even if the input sometimes is not. They argue that subtitle readers have only a limited time to perceive and understand a given subtitle and that therefore grammatical output is essential. And they follow the strategy that "it is preferable to drop elements from the output instead of translating them incorrectly" (p. 331). This is debateable and opens the door for incomplete output.

Although Popowich et al. (2000) call their system "a hybrid of both statistical and symbolic approaches" (p. 333), it is a symbolic system by today's standards. The statistics are only used for efficiency improvements but are not at the core of the methodology. The paper was published before automatic evaluation methods were invented. Instead Popowich et al. (2000) used the classical evaluation method where native speakers were asked to judge the grammaticality and fidelity of the system. These experiments resulted in "70% of the translations ... be ranked as correct or acceptable, with 41% being correct" which is an impressive result. Whether this project can be regarded as a MT success story depends on whether the system was actually employed in production. This information is not provided in the paper.

Melero et al. (2006) combined Translation Memory technology with Machine Translation, which looks interesting at first sight. But then it turns out that their Translation Memories for the language pairs Catalan-Spanish and Spanish-English were not filled with subtitles but rather with newspaper articles and UN texts. They don't give any motivation for this. And disappointingly they did not train their own MT system but rather worked only with free-access web-based MT systems (which we assume are rule-based systems).

They showed that a combination of Translation Memory with such web-based MT systems works better than the web-based MT systems alone. For English to Spanish translation this resulted in an improvement of around 7 points in BLEU scores (Papineni et al., 2001) but hardly any improvement at all for English to Czech.

3.2 Example-based MT of Film Subtitles

Armstrong et al. (2006) "ripped" subtitles (40,000 sentences) German and English as training material for their Example-based MT system and compared the performance to the same amount of Europarl sentences (which have more than three times as many tokens!). Training on the subtitles gave slightly better results when evaluating against subtitles, compared to training on Europarl and evaluating against subtitles. This is not surprising, although the authors point out that this contradicts some earlier findings that have shown that heterogeneous training material works better.

They do not discuss the quality of the ripped translations nor the quality of the alignments (which we found to be a major problem when we did similar experiments with freely available English-Swedish subtitles).

The BLEU scores are on the order of 11 to 13 for German to English (and worse for the opposite direction). These are very low scores. They also conducted user evaluations with 4-point scales for intelligibility and accuracy. They asked 5 people per language pair to rate a random set of 200 sentences of system output. The judges rated English to German translations higher than the opposite direction (which contradicts the BLEU scores). Owing to the small scale of the evaluation, however, it seems premature to draw any conclusions.

3.3 Statistical MT of Film Subtitles

Descriptions of Statistical MT systems for subtitles are practically non-existent probably due to the lack of freely available training corpora (i.e. collections of human-translated subtitles). Both Tiedemann (2007) and Lavecchia et al. (2007) report on efforts to build such corpora with alignment on the subtitles.

Tiedemann (2007) works with a huge collection of subtitle files that are available on the internet at www.opensubtitles.org. These subtitles have been produced by volunteers in a great variety of languages. But the volunteer effort also results in subtitles of often dubious quality (they include timing, formatting, and linguistic errors). The hope is that the enormous size of the corpus will supersede the noise in practical applications. The first step then is to align the files across languages on the subtitle level. The time codes alone are not sufficient as different (amateur) subtitlers have worked with different time offsets and sometimes even different versions of the same film. Still, Tiedemann (2007) shows that an alignment approach based on time overlap combined with cognate recognition is clearly superior to pure length-based alignment. He has evaluated his approach on English, German and Dutch. His results of 82.5% correct alignments for Dutch-English and 78.1% correct alignments for Dutch-German show how difficult the alignment task is. And a rate of around 20% incorrect alignments will certainly be problematic when training a Statistical MT system on these data.

Lavecchia et al. (2007) also work with subtitles obtained from the internet. They work on French-English subtitles and use a method which they call Dynamic Time Warping for aligning the files across the languages. This method requires access to a bilingual dictionary to compute subtitle correspondences. They compiled a small test corpus consisting of 40 subtitle files, randomly selecting around 1300 subtitles from these files for manual inspection. Their evaluation focused on precision while sacrificing recall. They report on 94% correct alignments when turning recall down to 66%. They then go on to use the aligned corpus to extract a bilingual dictionary and to integrate this dictionary in a Statistical MT system. They claim that this improves the MT system with 2 points BLEU score (though it is not clear which corpus they have used for evaluating the MT system).

This summary indicates that most work on the automatic translation of film subtitles with Statistical MT is still in its infancy. Our own efforts are larger and have resulted in a mature MT system. We will report on them in the following section.

4 The Stockholm MT System for Film Subtitles

We are building a Machine Translation system for translating film subtitles from Swedish to Danish (and Swedish to Norwegian) in a commercial setting. Some of this work has been described earlier by Volk and Harder (2007).

Most films are originally in English and receive Swedish subtitles based on the English video and audio (sometimes accompanied by an English manuscript). The creation of the Swedish subtitle is a manual process done by specially trained subtitlers following company-specific guidelines. In particular, the subtitlers set the time codes (beginning and end time) for each subtitle. They use an in-house tool which allows them to attach the subtitle to specific frames in the video.

The Danish translator subsequently has access to the original English video and audio but also to the Swedish subtitles and the time codes. In most cases the translator will reuse the time codes and insert the Danish subtitle. She can, on occasion, change the time codes if she deems them inappropriate for the Danish text.

Our task is to produce Danish and Norwegian draft translations to speed up the translators' work. This project of automatically translating subtitles from Swedish to Danish and Norwegian benefits from three favorable conditions:

1. Subtitles are short textual units with little internal complexity (as described in section 2).

2. Swedish, Danish and Norwegian are closely related languages.

3. We have access to large numbers of Swedish subtitles and human-translated Danish and Norwegian subtitles. Their correspondence can easily

be established via the time codes which leads to an alignment on the subtitle level.

But there are also aspects of the task that are less favorable. Subtitles are not transcriptions, but written representations of spoken language. As a result the linguistic structure of subtitles is closer to written language than the original (English) speech, and the original spoken content usually has to be condensed by the Swedish subtitler.

The task of translating subtitles also differs from most other machine translation applications in that we are dealing with creative language, and thus we are closer to literary translation than technical translation. This is obvious in cases where rhyming song-lyrics or puns are involved, but also when the subtitler applies his linguistic intuitions to achieve a natural and appropriate wording which blends into the video without disturbing. Finally, the language of subtitling covers a broad variety of domains from educational programs on any conceivable topic to exaggerated modern youth language.

We have decided to build a statistical MT (SMT) system in order to shorten the development time (compared to a rule-based system) and in order to best exploit the existing translations. We have trained our SMT system by using GIZA++ (Och and Ney, 2004)[3] for the alignment, Thot (Ortiz-Martínez et al., 2005)[4] for phrase-based SMT, and Phramer[5] as the decoder.

We will first present our setting and our approach for training the SMT system and then describe the evaluation results.

4.1 Swedish and Danish in Comparison

Swedish and Danish are closely related Germanic languages. Vocabulary and grammar are similar, however orthography differs considerably, word order differs somewhat and, of course, pragmatics avoids some constructions in one language that the other language prefers. This is especially the case in the contemporary spoken language, which accounts for the bulk of subtitles.

One of the relevant differences for our project concerns word order. In Swedish the verb takes non-nominal complements before nominal ones, where in Danish it is the other way round. The core problem can be seen in example 3 where the verb particle *ut* immediately follows the verb in Swedish but is moved to the end of the clause in Danish.

(3) SV: Du häller ut krutet.
DA: Du hælder krudtet ud.
EN: *You are pouring out the gunpowder.*

A similar word order difference occurs in positioning the negation adverb (SV: *inte*, DA: *ikke*). Furthermore, Danish distinguishes between the use of

[3]GIZA++ is accessible at http://www.fjoch.com/ GIZA++.html
[4]Thot is available at http://thot.sourceforge.net/
[5]Phramer was written by Marian Olteanu and is available at http://www.olteanu.info/

der (EN: *there*) and *det* (EN: *it*) but Swedish does not. Both Swedish and Danish mark definiteness with a suffix on nouns, but Danish does not have the double definiteness marking of Swedish.

4.2 Our Subtitle Corpus

Our corpus consists of TV subtitles from soap operas (like daily hospital series), detective series, animation series, comedies, documentaries, feature films etc. In total we have access to more than 14,000 subtitle files (= single TV programmes) in each language, corresponding to more than 5 million subtitles (equalling more than 50 million words).

When we compiled our corpus we included only subtitles with matching time codes. If the Swedish and Danish time codes differed more than a threshold of 15 TV-frames (0.6 seconds) in either start or end-time, we suspected that they were not good translation equivalents and excluded them from the subtitle corpus. In this way we were able to avoid complicated alignment techniques. Most of the resulting subtitle pairs are high-quality translations of one another thanks to the controlled workflow in the commercial setting.

In a first profiling step we investigated the vocabulary size of the corpus. After removing all punctuation symbols and numbers we counted all word form types. We found that the Swedish subtitles amounted to around 360,000 word form types. Interestingly, the number of Danish word form types is about 5.5% lower, although the Danish subtitles have around 1.5% more tokens. We believe that this difference may be an artifact of the translation direction from Swedish to Danish which may lead the translator to a restrictive Danish word choice.

Another interesting profiling feature is the repetitiveness of the subtitles. We found that 28% of all Swedish subtitles in our training corpus occur more than once. Half of these recurring subtitles have exactly one Danish translation. The other half have two or more different Danish translations which are due to context differences combined with the high context dependency of short utterances and the Danish translators choosing less compact representations.

From our subtitle corpus we chose a random selection of files for training the translation model and the language model. We currently use 4 million subtitles for training. From the remaining part of the corpus, we selected 24 files (approximately 10,000 subtitles) representing the diversity of the corpus from which a random selection of 1000 subtitles was taken for our test set. Before the training we tokenized the subtitles (e.g. separating punctuation symbols from words), converting all uppercase words into lower case, and normalizing punctuation symbols, numbers and hyphenated words.

4.3 Unknown Words

Although we have a large training corpus, there are still unknown words (words not seen in the training data) in the evaluation data. They comprise proper names of people or products, rare word forms, compounds, spelling deviations and foreign words. Proper names need not concern us in this context since the system will copy unseen proper names (like all other unknown words) into the Danish output, which in almost all cases is correct.

Rare word forms and compounds are more serious problems. Hardly ever do all forms of a Swedish verb occur in our training corpus (regular verbs have 7 forms). So even if 6 forms of a Swedish verb have been seen frequently with clear Danish translations, the 7th will be regarded as an unknown if it is missing in the training data.

Both Swedish and Danish are compounding languages which means that compounds are spelled as orthographic units and that new compounds are dynamically created. This results in unseen Swedish compounds when translating new subtitles, although often the parts of the compounds were present in the training data. We therefore generate a translation suggestion for an unseen Swedish compound by combining the Danish translations of its parts.

Variation in graphical formatting also poses problems. Consider spell-outs, where spaces, commas, hyphens or even full stops are used between the letters of a word, like "I will n o t do it", "Seinfeld" spelled "S, e, i, n, f, e, l , d" or "W E L C O M E T O L A S V E G A S", or spelling variations like *ä-ä-älskar* or *abso-jävla-lut* which could be rendered in English as *lo-o-ove* or *abso-damned-lutely*. Subtitlers introduce such deviations to emphasize a word or to mimic a certain pronunciation. We handle some of these phenomena in pre-processing, but, of course, we cannot catch all of them due to their great variability.

Foreign words are a problem when they are homographic with words in the source language Swedish (e.g. when the English word *semester* = "university term" interferes with the Swedish word *semester* which means "vacation"). Example 4 shows how different languages (here Swedish and English) are sometimes intertwined in subtitles.

(4) SV: Hon gick ut Boston University's School of the Performing Arts-
-och hon fick en dubbelroll som halvsystrarna i "As the World Turns".
EN: *She left Boston University's School of the Performing Arts and she got a double role as half sisters in "As the World Turns".*

4.4 Evaluating the Performance of the Stockholm MT System

We first evaluated the MT output against a left-aside set of previous human translations. We computed BLEU scores of around 57 in these experiments. In addition we computed the percentage of exactly matching subtitles against a previous human translation (How often does our system produce the exact

	Exact matches	Levenshtein-5 matches	BLEU
Crime series	15.0%	35.3%	63.9
Comedy series	9.1%	30.6%	54.4
Car documentary	3.2%	22.8%	53.6
Average	9.1%	21.6%	57.3

Table 1: Evaluation Results against a Prior Human Translation.

same subtitle as the human translator?), and we computed the percentage of subtitles with a Levenshtein distance of up to 5 which means that the system output has an editing distance of at most 5 basic character operations (deletions, insertions, substitutions) from the human translation.

We decided to use a Levenshtein distance of 5 as a threshold value as we consider translations at this edit distance from the reference text still to be "good" translations. Such a small difference between the system output and the human reference translation can be due to punctuation, to inflectional suffixes (e.g. the plural -s in example 5 with MT being our Danish system output and HT the human translation) or to incorrect pronoun choices.

(5) MT: Det gør ikke noget. Jeg prøver gerne hotdog med kalkun -
HT: Det gør ikke noget. Jeg prøver gerne hotdog**s** med kalkun**,** -
EN: *That does not matter. I like to try hotdog(s) with turkey.*

Table 1 shows the results for three files (selected from different genres), for which we have prior translations (done independently of our system). We observe between 3.2% and 15% exactly matching subtitles, and between 22.8% and 35.3% subtitles with a Levenshtein distance of up to 5. Note that the percentage of Levenshtein matches includes the exact matches (which correspond to a Levenshtein distance of 0).

On manual inspection, however, many automatically produced subtitles that were more than 5 keystrokes away from the human translations still looked like good translations. Therefore we conducted another series of evaluations with translators who were asked to post-edit the system output rather than to translate from scratch. We made sure that the translators had not translated the same file before.

Table 2 shows the results for the same three files for which we have one prior translation. We gave our system output to six translators and obtained six post-edited versions. Some translators were more generous than others, and therefore we averaged their scores. When using post-editing, the evaluation figures are 13.2 percentage points higher for exact matches and 19.5 percentage points higher for Levenshtein-5 matches. It becomes also clear that the

	Exact matches	Levenshtein-5 matches	BLEU
Crime series	27.7%	47.6%	69.9
Comedy series	26.0%	45.7%	67.7
Car documentary	13.2%	35.9%	59.8
Average	22.3%	43.1%	65.8

Table 2: Evaluation Results averaged over 6 Post-editors.

translation quality varies considerably across film genres. The crime series file scored consistently higher than the comedy file which in turn was clearly better than the car documentary.

There are only few other projects on Swedish to Danish Machine Translation (and we have not found a single one on Swedish to Norwegian). Koehn (2005) trained his system on a parallel corpus of more than 20 million words from the European parliament. In fact he trained on all combinations of the 11 languages in the Europarl corpus. Koehn (2005) reports a BLEU score of 30.3 for Swedish to Danish translation which ranks somewhere in the middle when compared to other language pairs from the Europarl corpus. The worst score was for Dutch to Finnish (10.3) and the best for Spanish to French translations (40.2). The fact that our BLEU scores are much higher even when we evaluate against prior translations (cf. the average of 57.3 in table 1) is probably due to the fact that subtitles are shorter than Europarl sentences and perhaps also due to our larger training corpus.

5 Conclusions

We have sketched the text genre characteristics of film subtitles and shown that Statistical MT of subtitles leads to good quality when the input is a large high-quality parallel corpus. We are working on Machine Translation systems for translating Swedish film subtitles to Danish and Norwegian with very good results (in fact the results for Swedish to Norwegian are slightly better than for Swedish to Danish).

We have shown that evaluating the system against independent translations does not give a true picture of the translation quality and thus of the usefulness of the system. Evaluation BLEU scores were about 8.5 points higher when we compared our system output against post-edited translations averaged over six translators. Exact matches and Levenshtein 5 scores were also clearly higher.

We are dealing with a customer-specific MT system covering a broad set of textual domains. The customer is satisfied and has recently started to employ our MT system in large scale production. It is too early to advertise this as

an MT success story as the overall productivity increase has not yet been determined. But we believe that our evaluation results are promising and hope that a future assessment will prove that the deployment of our MT system is profitable.

6 Acknowledgements

We would like to thank Jörgen Aasa, Søren Harder and Christian Hardmeier for sharing their expertise, providing evaluation figures and commenting on an earlier version of the paper.

References

Almqvist, I. and A. Sågvall Hein (1996). Defining ScaniaSwedish – a controlled language for truck maintenance. In *Proceedings of the First International Workshop on Controlled Language Applications*, Katholieke Universiteit Leuven.

Armstrong, S., A. Way, C. Caffrey, M. Flanagan, D. Kenny, and M. O'Hagan (2006). Improving the quality of automated DVD subtitles via example-based machine translation. In *Proceedings of Translating and the Computer 28*, London. Aslib.

Bick, E. (2007). Dan2eng: Wide-coverage danish-english machine translation. In *Proceedings of Machine Translation Summit XI*, Copenhagen.

de Linde, Z. and N. Kay (1999). *The Semiotics of Subtitling*. Manchester: St. Jerome Publishing.

Gottlieb, H. (2001). Texts, translation and subtitling - in theory, and in Denmark. In H. Holmboe and S. Isager (Eds.), *Translators and Translations*, pp. 149–192. Aarhus University Press. The Danish Institute at Athens.

Hutchins, J. (1999). The development and use of machine translation systems and computer-based translation tools. In *Proc. of International Symposium on Machine Translation and Computer Language Information Processing*, Beijing.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT-Summit*, Phuket.

Lavecchia, C., K. Smaili, and D. Langlois (2007). Machine translation of movie subtitles. In *Proceedings of Translating and the Computer 29*, London. Aslib.

Melero, M., A. Oliver, and T. Badia (2006). Automatic multilingual subtitling in the eTITLE project. In *Proceedings of Translating and the Computer 28*, London. Aslib.

Och, F. J. and H. Ney (2004). The alignment template approach to statistical machine translation. *Computational Linguistics 30*(4), 417–449.

Ortiz-Martínez, D., I. García-Varea, and F. Casacuberta (2005). Thot: A toolkit to train phrase-based statistical translation models. In *Tenth Machine Translation Summit*, Phuket. AAMT.

Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu (2001). Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center, Almaden.

Pedersen, J. (2007). *Scandinavian Subtitles. A Comparative Study of Subtitling Norms in Sweden and Denmark with a Focus on Extralinguistic Cultural References*. Ph. D. thesis, Stockholm University. Department of English.

Popowich, F., P. McFetridge, D. Turcato, and J. Toole (2000). Machine translation of closed captions. *Machine Translation 15*, 311–341.

Sågvall Hein, A., E. Forsbom, J. Tiedemann, P. Weijnitz, I. Almqvist, L.-J. Olsson, and S. Thaning (2002). Scaling up an MT prototype for industrial use – databases and data flow. In *Proceedings of LREC 2002. Third International Conference on Language Resources and Evaluation*, Las Palmas, pp. 1759–1766.

Tiedemann, J. (2007). Improved sentence alignment for movie subtitles. In *Proceedings of RANLP*, Borovets, Bulgaria.

Volk, M. and S. Harder (2007). Evaluating MT with translations or translators. What is the difference? In *Machine Translation Summit XI Proceedings*, Copenhagen.

Whitelock, P. and K. Kilby (1995). *Linguistic and Computational Techniques in Machine Translation System Design* (2 ed.). Studies in Computational Linguistics. London: UCL Press.