DEVELOPING AN ICELANDIC TO ENGLISH SHALLOW TRANSFER MACHINE TRANSLATION SYSTEM

Martha Dís Brandt
Master of Science
Language Technology
January 2011
School of Computer Science
Reykjavík University

M.Sc. RESEARCH THESIS

ISSN 1670-8539

Developing an Icelandic to English Shallow Transfer Machine Translation System

by

Martha Dís Brandt

Research thesis submitted to the School of Computer Science
at Reykjavík University in partial fulfillment of
the requirements for the degree of
Master of Science in **Language Technology**

January 2011

Research Thesis Committee:

Dr. Hrafn Loftsson, Supervisor
Assistant Professor, Reykjavík University

Eiríkur Rögnvaldsson
Professor, University of Iceland

Dr. Hannes Högni Vilhjálmsson
Associate Professor, Reykjavík University

Copyright
Martha Dís Brandt
January 2011

Developing an Icelandic to English Shallow Transfer Machine Translation System

Martha Dís Brandt

January 2011

Abstract

This thesis describes the author's contribution to creating a prototype of a machine translation system that translates from Icelandic to English. The prototype is an open source shallow-transfer machine translation system based on the Apertium platform with existing Icelandic Language Technology (LT) tools from the IceNLP toolkit integrated into it.

The purpose of integrating existing Icelandic LT tools into the Apertium platform was to avoid re-inventing the wheel and in the hopes that the prototype would produce better translation quality with state-of-the-art modules.

The integration itself was successful, even though the presupposition that Apertium modules could be easily substituted with IceNLP modules turned out to be overly optimistic. Evaluation of the prototype's word error rate (WER) and position-independent error rate (PER) is 50.60% and 40.78% respectively, which is higher than other systems it was compared to.

Based on the high error rates, it was assumed that the prototype would also rank poorly in comparison to two other publicly available Icelandic-English MT systems if assessed subjectively. The assumption was indeed supported by the results of a subjective user survey presented to the student body of Reykjavík University.

Further work must be carried out in order to make this method of hybridization a feasible option for creating an Apertium-based Icelandic-English shallow-transfer MT system.

This research project was funded by a grant of excellence from the Icelandic Research Fund.

Þróun íslensk-ensks vélræns grófþýðingarkerfis

Martha Dís Brandt

janúar 2011

Útdráttur

Þessi ritgerð lýsir framlagi höfundar við að búa til frumgerð af vélrænu þýðingarkerfi sem þýðir frá íslensku yfir á ensku. Hugbúnaður frumgerðar grófþýðingarkerfisins er opinn og byggist á Apertium rammanum ásamt tiltækum opnum málvinnslutólum úr IceNLP málvinnslupakkanum sem voru samþættuð í þessa frumgerð þýðingarkerfisins.

Tilgangur þess að tengja tiltæk opin íslensk málvinnslutól inn í Apertium rammann var annars vegar sá að forðast það að vinna eitthvað frá grunni sem þegar væri til og hins vegar sá að vonast var til að frumgerðin myndi þá skila betri gæðum á þýðingunum.

Það tókst að samþætta þessi tól, þrátt fyrir það að hugmyndin um að það væri auðveldlega hægt að skipta út Apertium módúlum fyrir IceNLP módúla reyndist vera fullbjartsýn. Mat á hlutfalli villuorða (e. WER) og hlutfalli staðsetningarvillna (e. PER) leiddi í ljós að villuorðahlutfallið er 50.60% og staðsetningarvilluhlutfallið er 40.78%, sem er hærra en önnur þýðingarkerfi sem frumgerðin var borin saman við.

Þar sem villuhlutföllin voru svo há, þá var ályktað sem svo að frumgerðin myndi ekki vera hátt skrifuð í huglægu mati í samanburði við tvö önnur íslensk-ensk vélræn þýðingarkerfi sem eru fáanleg á opinberum markaði. Niðurstöður könnunar sem var lögð fyrir nemendur Háskólans í Reykjavík studdu einmitt þá ályktun.

Frekari vinnu verður að leggja í frumgerðina til þess að gera þesskonar samþættingu að vænlegum kosti við gerð vélræns grófþýðingarkerfis byggðu á Apertium rammanum fyrir íslensku-ensku.

Þetta rannsóknarverkefni var styrkt af öndvegisstyrk Rannís.

For my grandmothers Hilda and Alla, my mother Gugga, my numerous relatives in the field of computer science, including my father Rick and brother David, and especially for my son Daníel Friðgeir.

Acknowledgements

I would like to thank the following individuals and organizations: *Robert J. Chatfield*, who encouraged me to pursue the path that interested me instead of taking the "more sensible" one, which brought me to where I am today; *Dr. Hrafn Loftsson*, for his supervision, guidance, expertise and encouragement; *Francis M. Tyers* and *Hlynur Sigurþórsson*, for their assistance and contributions to the transfer rules and adaption of IceNLP, respectively; *Anton Karl Ingason*, for graciously allowing the integration of his bilingual wordlist into the prototype; and *my friends and family*, who were there when I needed them most.

Last but certainly not least, many thanks are due to the publishing company *Forlagið* for sharing a good portion of the bilingual data used in this project and *The Icelandic Research Fund*, for the grant of excellence awarded to the Icelandic Centre for Language Technology for the project *"Viable Language Technology Beyond English - Icelandic as a Test Case"* (no.209001), which funded this research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis describes the work involved in creating a prototype of a shallow-transfer machine translation system that translates between Icelandic and English. The prototype is an open source system based on the Apertium platform with existing Icelandic Language Technology (LT) tools from the IceNLP toolkit integrated into it.

The main goal for the quality of the prototype's output was for it to be understandable without demanding grammatical correctness, i.e. it was meant for assimilation. However, most machine translation evaluation methods consider the dissemination value of translations. See section 2.2 for further differentiation between assimilation and dissemination.

One of the main reasons for using the Apertium platform was because its modules connect into a kind of pipeline that can be substituted with external modules without breaking the flow. The purpose of integrating existing Icelandic LT tools into the Apertium platform was to avoid re-inventing the wheel and in the hopes that the prototype would produce better translation quality with state-of-the-art modules. This turned out to be not quite as straight-forward as was first believed.

Various people have contributed to this project. Dr. Hrafn Loftsson and Hlynur Sigurþórsson adapted some of the IceNLP modules to make them compatible with the Apertium platform and modified the whole IceNLP toolkit to be open-source. Hlynur also transformed the toolkit into a daemonized version, meaning that it can now run in the background on a computer waiting for input (see section 4.3 for further details). Francis Tyers constructed the first set-up of the three dictionaries (see sections 3.3 and 4.1), and also

contributed to the creation of the transfer rules (see section 4.2). Francis also graciously used his own previously applied methods to produce a dump of the Icelandic Wikipedia website for evaluation purposes.

My contribution consisted of manually reviewing each of the approximately 5,000 entries in the bilingual dictionary, making corrections and modifications where necessary (see section 4.1); pre-processing and manually reviewing two large additional data sets, taking approximately 24,000 SL words and adding 19,400 entries to the bilingual dictionary (see section 4.1.1); adding some transfer rules (see section 4.2); and adding multiword expressions to the prototype (see section 4.4).

I generated a target language corpus from the approximately 188,000 lines from the Icelandic Wikipedia source, then randomly selected 1,000 sentences which were pruned down to 397 to use for evaluation of the Apertium-IceNLP prototype and other Icelandic-English MT systems (see section 5.1.1) and performed the evaluation itself (see section 5.2).

Evaluation of the prototype's word error rate (WER) and position-independent error rate (PER) is 50.60% and 40.78% respectively which are higher than the scores of other publicly available Icelandic-English MT systems that were measured for comparison: WER and PER for Google translate (http://translate.google.com/#is|en|) measured 33.63% and 22.15% respectively, and for Tungutorg (http://www.tungutorg.is/) 45.98% and 28.87% respectively.

The next logical step was to try to improve the performance of the prototype, and towards that end I performed an analysis. First, I collected a development data set from a different source than the evaluation data, i.e. from the online newspaper 'mbl.is' (see section 5.3.1); then I randomly selected and manually reviewed 50 files from the development data and identified 18 error categories (see section 5.3.2). I considered the relation between those error categories and grouped them into six meta-categories for further analysis, and made suggestions of areas to concentrate efforts for improvement of the prototype (see sections 5.3.2 and 5.5).

I also created a user survey based on the evaluation data with 40 randomly selected SL sentences and the TL sentences of the prototype and two other comparable MT systems,

for the user to rank subjectively (see section 5.4). The subjective user survey was presented to the student body of Reykjavík University and supports the assumption that the prototype overall ranks worst compared to Google translate and Tungutorg.

Further work must be carried out in order to make this method of hybridization a feasible option for creating an Icelandic-English MT system based on the Apertium platform. This research project was funded by a grant of excellence from the Icelandic Research Fund.

1.1 Language Technology (LT)

This section provides some background information on the scientific field of Language Technology (LT), the history of the field in general and how it pertains to Iceland and Icelandic.

1.1.1 What is LT?

The purpose of LT is to apply technology to language such that the outcome may further the knowledge base of the user, whether that is by providing understanding of a non-native language, perfecting the grammatical use of a written language, encompassing auditory usage or purely quizzical in nature.

LT is not exactly the same as Natural Language Processing (NLP), although many people use the two terms interchangeably. NLP concentrates mainly on the structure and semantics of natural languages, i.e. on the one hand taking a natural language and analyzing its structure and semantics, and on the other hand generating a language from some structural and semantic rules, while LT covers a range of scientific fields including machine translation, grammar checking, information retrieval and information extraction, question and answering systems, speech recognition and speech synthesis.

In many LT applications, one of the fundamental properties for their functionality is to be able to identify which word category the input belongs to. Word category disambiguation is called part-of-speech (POS) tagging which uses a list of possible linguistic characterizations, called a tagset, for the language in question. This tagging is usually performed

by a program called a POS tagger. Tagsets can vary greatly regarding how much linguistic detail they represent, e.g. the Penn-Treebank tagset for English consists of 36[1] tags while the Icelandic tagset consists of approximately 700 tags (Santorini, 1995; Pind, Magnússon, & Briem, 1991).

1.1.2 History of LT in Iceland

Compared to other countries, LT is a relatively young field in Iceland, and subsequently, so is Machine Translation (see section 2).

The Icelandic Ministry of Education, Science and Culture recognized in 1996 the importance of strengthening the use of Icelandic in information technology, which led to the translation of Microsoft's operating system into Icelandic. Furthermore, the Minister of Education appointed a committee to delve into LT and the status thereof in Iceland. This committee published its report in April 1999 (Ólafsson, Rögnvaldsson, & Sigurðsson, 1999).

The report revealed that experts in the field of LT were non-existent in Iceland, although there were many who had expertise in some areas of LT and interest in other areas. The committee pointed out that despite the fact that there were not many who spoke Icelandic, around 300 thousand, that the language was used on a daily basis in all of the nation's local communications and business interactions. Furthermore, that while information technology was well developed and widespread in the country, user interfaces were rarely in Icelandic.

This presented a new, unprecedented dilemma in the history of the Icelandic language, namely an important part of daily life where Icelanders could not use their mother tongue. The committee emphasized that this dilemma was actually threefold; *i)* it was an important communication factor, *ii)* it was a part of daily life and *iii)* it applied to the public, not just some scientists in a narrow field - and reasoned that the language might be able to resist a combination of two of these factors, but when all three factors came together, the language could be in danger of expiring.

[1] 45 including tags for punctuation marks.

The report concluded that the government needed to put considerable funds and effort into four areas in order to bring Icelandic LT up to speed:

1. Accessible data to create LT products
2. Viable research of LT
3. Development of LT products
4. Education in the fields of LT and Computational Linguistics

The estimated total cost for these four areas was 225-250 million ISK per year. The committee also cautioned that action needed to be taken swiftly or else risk that the necessary knowledge to address LT issues for Icelandic may never reach the country.

One of the first things any language needs in order to do any kind of research in LT is accessible data to work from. Icelandic dictionaries have existed long before the term 'language technology' has been used, but one extremely important publication for the field of LT was the Icelandic Frequency Dictionary (IFD). The IFD, which is used as the gold standard for tagging Icelandic text, is a balanced corpus with approximately 600,000 tokens, first published in 1991 (Pind et al., 1991).

While working on the IFD, Stefán Briem wrote a program that tags Icelandic text with Icelandic POS tags to speed up the task of tagging the whole text (Pind et al., 1991; Briem, 2009). Later he improved this tagger and used it in his online machine translation system, Tungutorg, which will be discussed further in section 2.6.2.

Subsequently, after receiving the report in 1999, the Icelandic government started an LT program in 2000, which resulted in a number of LT resources (Rögnvaldsson et al., 2009):

- A full-form morphological database of Modern Icelandic inflections (Bjarnadóttir, 2004, 2005).
- A balanced morphosyntactically tagged 25 million word corpus (Helgadóttir, 2004).
- A training model for data-driven POS taggers (Helgadóttir, 2005, 2007).
- A text-to-speech system (Rögnvaldsson, Kristinsson, & Þorsteinsson, 2006).
- A speech recognizer (Rögnvaldsson, 2004; Waage, 2004).

- An improved spell-checker (Skúlason, 2004).

Then after the time-limited government-funded LT program expired, three research institutions combined forces and founded the *Icelandic Centre for Language Technology* (ICLT) in 2005: the University of Iceland, the Reykjavik University and the Árni Magnússon Institute for Icelandic Studies. The main purpose of ICLT is to foster and facilitate the advancement of LT for Icelandic in any way possible.

In recent years, a few Basic Language Research Kit (BLARK) modules have been developed for the Icelandic language, e.g. a POS tagger (Loftsson, 2008), a lemmatizer (A. Ingason, Helgadóttir, Loftsson, & Rögnvaldsson, 2008), a shallow parser (Loftsson & Rögnvaldsson, 2007b) and a context-sensitive spell checker (A. K. Ingason, Jóhannsson, Rögnvaldsson, Loftsson, & Helgadóttir, 2009). None of these would have been possible to produce without the IFD.

The author of this thesis humbly aspires to have the prototype machine translation system described here become a part of the Icelandic BLARK as well.

Chapter 2

Machine Translation (MT)

MT is one area of LT. It is the field of automatically translating some text from a source language (SL) to a target language (TL). This section provides background information regarding the field of Machine Translation (MT), its history in general and within Iceland. We also look at what MT is useful for (and what not), various MT methods, what difficulties MT faces and some evaluation methods for MT.

2.1 History of MT

The theory of MT pre-dates computers, with philosophers' Leibniz and Descartes' ideas of using code to relate words between languages in the seventeenth century (J. Hutchins, 1993).

The early 1930s saw the first patents for 'translating machines'. Georges Artsrouni was issued his patent in France in July 1933. He developed a device "*which [he] called a 'cerveau mécanique'*" (mechanical brain) that could translate between languages using four components: memory, a keyboard for input, a search method and an output mechanism. The search method was basically a dictionary look-up in the memory and therefore Hutchins is reluctant to call it a translation system. The proposal of Russian Petr Petrovich Troyanskii patented in September 1933 even bears a resemblance to the Apertium system (see 3.2.2), using a bilingual dictionary and a three-staged process, i.e. first a native-speaking human editor of the SL pre-processed the text, then the machine performed the translation, and finally a native-speaking human editor of the TL post-edited the text (J.

Hutchins, 1993; W. J. Hutchins & Lovtskii, 2000).

After the birth of computers, (J. Hutchins, 2005a) divides the early years of MT into *'the pioneers'* from 1947-1954 ending with the first public demonstration of MT in the Georgetown-IBM experiment which proved deceptionally promising, encouraging financing of further research in the field; *'the decade of optimism'* from 1954-1966 where "*the many predictions of imminent 'breakthroughs' [were thwarted] as researchers encountered 'semantic barriers' for which they saw no straightforward solutions*" (J. Hutchins, 2005a, p. 2), culminating in the creation of the Automated Language Processing Advisory Committee (ALPAC); and *'the aftermath of the ALPAC report'* from 1966-1980 which brought MT research to its knees, suspending virtually all research in the USA while some research continued in Canada, France and Germany.

MT took off with the Georgetown-IBM experiment, where over 60 Russian sentences were "*translated smoothly*" into English using 6 rules and a bilingual dictionary consisting of 250 Russian words, with rule-signs assigned to words with more than one meaning. Although Professor Leon Dostert cautioned that this experimental demonstration was only a scientific sample, or "*a Kitty Hawk*[1] *of electronic translation*", the successfully translated test sentences and his prediction that in "*five, perhaps three years hence, interlingual meaning conversion by electronic process in important functional areas of several languages may well be an accomplished fact.*" (IBM, 1954) fueled optimism and attracted funding to the research field.

Then followed the notorious ALPAC report's crippling effect on the advancement of MT, which stemmed from the committee's belief that they could not find any "*pressing need for MT*" (ALPAC, 1966, p. 24) nor "*an unfulfilled need for translation [in general]*" (ALPAC, 1966, p. 11), that MT "*serves no useful purpose without postediting*" (ALPAC, 1966, p. 24), and that although "*a flawless and polished translation for a user-limited readership is wasteful of both time and money*" the committee also believed that "*production of an inferior translation [...] is even more wasteful of resources.*" (ALPAC, 1966, p. 16).

The killing blow is not in the committee's final chapter of recommendations, but the previous one where they first state that "*there is no immediate or predictable prospect of useful machine translation*" (ALPAC, 1966, p. 32) and then conclude the chapter with

[1] Kitty Hawk, North Carolina, USA was the site for the world's first successful powered human flight by the Wright brothers. "Kitty Hawk" references generally meant a break-through success in its early stages.

"*the total annual expenditure for research and development toward improving translation [...] should be spent hardheadedly toward important, realistic, and relatively short-range goals.*" (ALPAC, 1966, p. 33)

The implications of the report were grave enough that Harvey Brooks, chairman of the Committee on Science and Public Policy, was "*prompted by fear that the [ALPAC] committee report, read in isolation, might result in termination of research support for computational linguistics as well as in the recommended reduction of support aimed at relatively short-term goals in translation,*" to request that John R. Pierce, chairman of ALPAC, "*prepare a brief statement of the support needs for computational linguistics, as distinct from automatic language translation.*" (ALPAC, 1966, p. iv)

A wide variety of MT systems emerged after 1980 from various countries while research continued on more advanced methods and techniques. Those systems mostly comprised of indirect translations or used an 'interlingua' as its intermediate. In the 1990s statistical machine translation (SMT) and what is now known as example-based machine translation (EBMT) saw the light of day. At this time the focus of MT began to shift somewhat from pure research to practical application. Moving towards the change of the millennium, MT became more readily available to individuals via online services and software for their personal computers (PCs). More details regarding different MT methods are in section 2.3.

2.2 Uses for MT

There are two main usages for machine translation; *assimilation* and *dissemination*:

1. *Assimilation* is about enabling readers to understand some text, i.e. to grasp the general meaning of it while it may not be grammatically correct. This is for example used when translating e-mails, websites for reading, etc.
2. *Dissemination* is when some text needs to be publishable after translation, i.e. as close to grammatically correct as is possible so that it requires less effort to post-edit than translating from scratch. The point is not to replace human translators but to make their job easier.

2.3 Various MT Methods

MT methods can be categorized into rule-based machine translation (RBMT), statistical machine translation (SMT) and example-based machine translation (EBMT). RBMT methods can furthermore be categorized into dictionary-based, interlingual and transfer-based methods. More about these categories follow below:

- A *dictionary-based* method or *direct approach* (W. J. Hutchins & Somers, 1992) is known as the first generation of MT. It performs word-for-word look-up in a bilingual dictionary with some local word order adjustment and ignores "*grammatical relationships between the principal parts of the sentences*".

- The *interlingual approach* (W. J. Hutchins & Somers, 1992) involves converting the SL into an abstract language-independent representation before converting the abstraction to the TL. This method is most attractive for multilingual translation systems, as neither the analysis of a SL nor the generation of a TL affect the structure of the intermediate abstraction. Thus, on the one hand, it appears to be an easy task to add language pairs to the translation system, but on the other hand, the strict separation of the analysis and generation also cause disadvantages, e.g. analysis should not be oriented towards a particular TL nor is it possible to reference the SL when generating the TL.

- The *transfer-based* translation method (W. J. Hutchins & Somers, 1992) uses linguistic rules to perform morphological analysis on the SL text, next it will generally perform lexical categorization (disambiguation) and/or lexical transfer (similar to a dictionary look-up) and structural transfer rearranges the text into a representation like that of the TL. Finally, morphological generation is performed to create the correct surface forms of the words. The difference between a direct approach and a transfer approach is that the transfer approach has a language-dependent intermediate layer. Each language pair will have its own intermediate transfer module for that specific language pair.

  The disadvantage for using the transfer-based approach for a multilingual MT system is that the number of transfer modules needed will be $n(n-1)$ in addition to the n analysis and n generation modules (where n is the number of languages in the MT system), whereas an interlingual approach will only need $2n+1$ modules, i.e.

the n number of analysis and n number of generation modules plus the interlingua abstraction module.

Transfer-based translation can be further categorized into shallow and deep transfer methods: **Deep-transfer machine translation** requires full parsing and disambiguation of whole sentences, while **shallow-transfer machine translation** (STMT) works on partial sentence chunks.

- The *statistical machine translation* approach (J. Hutchins, 2005b) uses statistical models to calculate the most likely translation of the SL text into the TL text. A bilingual corpus is aligned on the sentence level first and then on the word level. Based on these alignments, a 'translation model of SL-TL frequencies' and a 'language model of TL word sequences' are derived, which are then used to calculate, in advance of the translation process, the most probable TL output for each SL input.

- The *example-based* approach to MT (J. Hutchins, 2005b) is somewhere between RBMT and SMT. According to J. Hutchins, the definition of EBMT and its boundaries have been difficult to define as it often bears resemblances to RBMT or SMT systems. In particular, SMT systems no longer focus solely on the word level, as phrase-based and syntax-based parsing is being used to improve translation quality, making the distinction between SMT and EBMT less clear. Essentially, J. Hutchins concludes that the distinguishable characteristic feature of EBMT is that it uses "'analogues' (similar in meaning and form) of SL sentences in existing TL texts", whereas SMT uses statistical models derived from aligned bilingual corpora and RBMT uses representations of equivalent meanings.

2.4 Difficulties with MT

Arnold (2003) maintains that one of the reasons translation is difficult for computers is that it is also difficult for humans. Human translators are expected to deliver an equally "good" text in the TL as the original SL, which largely depends on i) the context, ii) whether there exists a precise equivalent term in the TL and sometimes iii) on cultural knowledge; thus rendering the task somewhat difficult.

The problem here is determining what can be perceived as a "good" translation; should the result be "clear and unambiguous", "elegant and poetic", "persuasive and gripping", or even all of the above? The difficulty with this lies in the word "perception", for it implies that the translator understands the text, and by applying their own world-knowledge they are able to evaluate and modify their translation to give the desired result.

Which brings us to certain limitations of computers, as described by Arnold (2003):

- **Computers are unable to perform vaguely specified tasks.** Any variable that triggers an operation in a software program must not be vaguely specified if the program is to be expected to perform reasonably. An example of a vague task is e.g. "open the window if the room temperature is *too warm*", as this does not specify what the threshold measurement needs to be to signify when the temperature variable *too warm* becomes true. Likewise, any program must have a finite set of operations in order for it to complete, or it will otherwise remain stuck in an endless loop, if it runs at all. The common instructions on the back of a shampoo bottle are an example of an infinite set of operations with a missing stopping condition: "Lather, rinse, repeat."

- **Computers are unable to learn things.** Although Arnold allowed that there existed learning algorithms for certain tasks, his view was that they did not produce the kinds of knowledge required for MT, and therefore listed this point as one of the 4 major computer limitations that cause difficulties within MT. However, it can no longer be maintained today that computers are **unable** to learn things as can be seen from the ongoing research project "NELL: Never-Ending Language Learning" led by Tom Mitchell at Carnegie Mellon University (Carlson et al., 2010), which has accumulated approx. 480 thousand beliefs in its knowledge base since January 2010, that were derived automatically and for the most part without human supervision.

- **Computers are unable to perform common-sense reasoning.** Here he states that *'common-sense reasoning involves literally millions of facts about the world (water is wet, men don't get pregnant, most people have two feet, sheep are larger than fountain pens, if B has been put into A then A contains B [...])'*, furthermore maintaining that such a task as transforming these facts into code would be daunting and thus *'most of what we understand by "common-sense reasoning" is far beyond the reach of modern computers.'* As pointed out in

the previous point, "NELL" is currently learning and building a knowledge base using what could be called common-sense reasoning, which is really nothing more than application of logical deductions. Therefore this particular limitation is no longer as great an obstacle as it was previously considered.

- **Computers are unable to deal with combinatorially explosive problems.** The more variables that need to be considered in combination with each other, the longer it will take to reach a conclusion, and each time another variable is added to the mix the number of combinations to consider expands exponentially. In relation to MT, let us look at a short Icelandic sentence that has more than one English translation:

"*Bóndinn greiddi konunni.*"

The noun *bóndi* ($word_1$) has 3 translations in this prototype's bilingual dictionary: 'farmer', 'husband', and 'master'. The verb *greiða* ($word_2$) has 2 translations: 'pay' and 'comb'. Finally, the noun *kona* ($word_3$) has 3 translations: 'woman', 'wife' and 'lady'.

Thus, there are 18 ($3 \times 2 \times 3$) possible translations for this short sentence:

1) 1.1.1. The farmer paid the woman.
2) 1.1.2. The farmer paid the wife.
3) 1.1.3. The farmer paid the lady.

4) 1.2.1. The farmer combed the woman.
5) 1.2.2. The farmer combed the wife.
6) 1.2.3. The farmer combed the lady.

7) 2.1.1. The husband paid the woman.
8) 2.1.2. The husband paid the wife.
9) 2.1.3. The husband paid the lady.

10) 2.2.1. The husband combed the woman.
11) 2.2.2. The husband combed the wife.
12) 2.2.3. The husband combed the lady.

13) 3.1.1. The master paid the woman.
14) 3.1.2. The master paid the wife.
15) 3.1.3. The master paid the lady.

16) 3.2.1. The master combed the woman.
17) 3.2.2. The master combed the wife.
18) 3.2.3. The master combed the lady.

The computational limitations are slowly being overcome, but the biggest problem with MT remains: ambiguity, which can further be categorized into various types, so let us look at some of those types:

- **Lexical ambiguity.** *"The post has arrived."* Is the meaning that 'the mail' has been delivered or a 'piece of wood'?
- **Structural ambiguity.** *"The minister stated that the proposal was rejected yesterday."* What occurred yesterday, the rejection, or the minister's statement?
- **Anaphoric expressions.** Anaphora is an instance of an expression referring to another. *"The dog ate the bird and **it** died."* Did the dog or the bird die?

In most cases a discourse model might resolve the ambiguity issue. A discourse model involves analysis of relations between words, creating a model of those relationships from some discourse (a form of communication) for back-tracking references. Other cases can be solved with lexical selection, where syntactical patterns in a sentence are used to choose between multiple translations of a single word or phrase.

2.5 Evaluation Methods for MT

In this section we will look at various methods to evaluate machine translation systems.

2.5.1 F-score

F-score is a statistical measurement of a test's accuracy and requires both the precision p and the recall r of the test to calculate the accuracy. Precision is the number of correct results divided by the total number of retrieved results. Similarly, recall also uses the number of correct results, but this time it is divided by the total number of relevant results.

$$P = \frac{tp}{tp + fp}$$

$$R = \frac{tp}{tp + fn}$$

Where tp, fp, fn and tn are defined as *true positive*, *false positive*, *false negative* and *true negative*, such that tp is a correct word in the correct position, fp is a correct word in the wrong position, fn is a wrong word in the correct position (think of it as a placeholder), and tn is a wrong word in the wrong position (see table 2.1):

		Correct result	
		relevant	irrelevant
Test result	retrieved	tp	fp
	not retrieved	fn	tn

Table 2.1: *Classification context table.*

Thus, the traditional F-score is the harmonic mean[2] of precision, P, and recall, R:

$$F = \frac{2 \times P \times R}{P + R}$$

2.5.2 WER and PER

WER stands for Word Error Rate and is a common measurement tool for MT systems. WER is based on the Levenshtein distance[3], but is applied on whole words instead of individual characters.

[2] The harmonic mean for two numbers is calculated by multiplying them together, doubling that number and then dividing the result with the sum of the same two original numbers.

[3] The Levenshtein distance (Levenshtein, 1966) is calculated by adding the number of insertions, deletions and/or substitutions needed to change one word into another. For example, the Levenshtein distance between stapler and papers is 3 (from stapler: delete s, substitute t for p, delete l and insert s at the end).

If the TL sentence does not match the reference sentence[4], then the difference between the two sentences is eradicated by making substitutions, deletions and insertions until the TL sentence matches the reference sentence. These actions are used to calculate the WER, with S as the number of substituted words in the TL sentence, D as the number of deleted words from the TL sentence, I the number of inserted words into the TL sentence and N is the total number of words in the reference sentence:

$$\text{WER} = \frac{S + D + I}{N}$$

PER, Position-independent Error Rate, is related to WER and calculates an error rate based on the number of correct words regardless of where they are positioned within the sentence. Again, N is the total number of words in the reference sentence (same as in the WER formula), T stands for the total number of words in the TL sentence and C is the total number of correct words in the TL sentence.

$$\text{PER} = 1 - \left(\frac{C - \max(0, (T - N))}{N}\right)$$

First, the difference between T and N is found. If that result is a positive number then it is subtracted from C, otherwise nothing (0) is subtracted from C. This number is then divided with N, and that result is subtracted from one (1) to get the PER score.

For example, if $T = 6042$, $N = 6374$ and $C = 3775$:

$$\text{PER} = 1 - \left(\frac{3775 - \max(0, (6042 - 6374))}{6374}\right)$$

$$= 1 - \left(\frac{3775 - \max(0, (-332))}{6374}\right)$$

$$= 1 - \left(\frac{3775 - 0}{6374}\right)$$

$$= 1 - \left(\frac{3775}{6374}\right)$$

[4] A reference sentence is the human translation or gold standard that another sentence is compared to.

$$= 1 - 0.5922$$

$$= 0.4078$$

2.5.3 BLEU

BLEU stands for Bilingual Evaluation Understudy. BLEU is based on WER with allowances for multiple reference translations, use of synonyms and alternative word order (Papineni, Roukos, Ward, & Zhu, 2002). It assumes that there are multiple reference translation sentences (which were translated directly by multiple humans) to compare the machine translated sentence with. They suggest that it can also work with a single source reference translation corpus, '*provided that the translations are not all from the same translator*'.

The BLEU method provides a score between 0 and 1, which is a scale indicating how similar the TL text is to the reference texts, where 1 is the perfect score. This score is calculated through a series of steps. BLEU's baseline metric is a 'modified n-gram precision' that calculates precision using the maximum number of words in the TL found in any of the reference sentences, divided by the total number of words in the TL sentence.

Papineni et al. (2002) demonstrate the calculation of a modified unigram precision with table 2.2:

Candidate sentence (TL)	the	the	the	the	the	the	the
Reference sentence one	the	cat	is	on	the	mat	
Reference sentence two	there	is	a	cat	on	the	mat

Table 2.2: *Example of one (improbable) TL sentence and two reference sentences.*

There are 7 words in the TL sentence. Reference sentence one has 2 words from the TL sentence, while reference sentence two only has 1 word, therefore the maximum number of words in the TL which were found in any of the reference sentences is 2. Thus, the modified unigram precision p_1 is $2/7$ instead of $7/7$ if it were calculated with normal unigram precision.

Using this method to calculate modified precision, they proceed to calculate the modified precision for all n-grams up to length N. Then they calculate the average logarithm with uniform weights[5], w_n, of all of the p_n. BLEU introduces a multiplicative brevity penalty, BP, but in order to compensate for harsh penalties of short sentences' deviations in length, the brevity penalty is only calculated over the whole corpus '*to allow some freedom at the sentence level*'(Papineni et al., 2002, p.315).

To calculate the brevity penalty, let c be the length of the candidate translation and r be the effective reference corpus length:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}.$$

Now that those values have been computed, BLEU is calculated like so for the whole corpus:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right).$$

[5] The average logarithm with uniform weights is equivalent to using the geometric mean, i.e. taking the nth root of the result of multiplying the relevant numbers up to the nth number.

2.6 Related work

The work described in this thesis is about a prototype of a machine translation system that translates from Icelandic to English, therefore we must look at other systems that also perform translations between the two aforementioned languages.

2.6.1 InterTran

Translation Experts Ltd. (http://www.tranexp.com/) have been providing a range of translation services, including software for machine translation such as InteractiveTran and NeuroTran, since 1992. In 2009 they added Icelandic to their language pool (TranslationExperts, 2010b).

The company's software "*translates sentence-by-sentence by using advanced artificial intelligence rules*" and by accessing a knowledgebase that exceeds 125 terabytes stored on their Translation Server through an internet connection. NeuroTran "*is smart and will enable users a much higher degree of accuracy during translation*" than of its predecessor InteractiveTran. NeuroTran is a "*hybrid system with a combination of linguistic rules, statistical methods and neural networks*" (TranslationExperts, 2010b). It also uses text analysis to determine the type of text to be translated, e.g. whether the context is technical, computer related or medical. The company claims that this kind of lexical selection is unique to their product (TranslationExperts, 2010a).

In addition to the software, the company also has a web-based product called *InterTran*, which utilizes NeuroTran, and can be found here:
http://www.tranexp.com:2000/Translate/result.shtml

The sample Icelandic sentence *Allir stóru strákarnir borðuðu góðu súpuna* (English literal translation: All big boys-the ate good soup-the) gave tragic results, yet still allowed for corrections (see figures 2.1 and 2.2).

Figure 2.1: *Sample sentence tested on InterTran.*

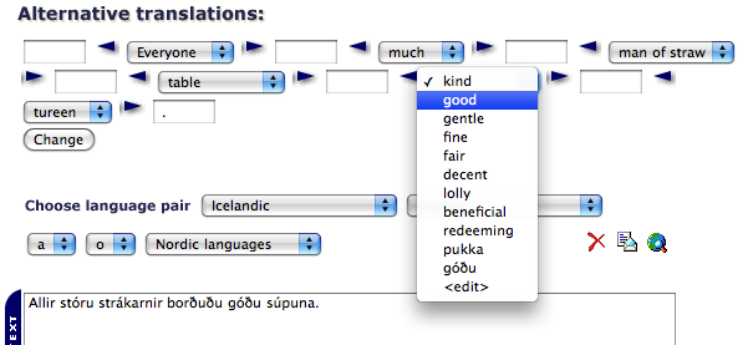

Figure 2.2: *InterTran allows for corrections.*

2.6.2 Tungutorg

Tungutorg (Briem, 2010) is a contribution to Icelandic LT by a highly motivated individual, Stefán Briem. He made his translation system publicly available as an HTTP web form in March 2008. He created a POS tagger for the Icelandic Frequency Dictionary in 1989 and made several improvements to it, which he then based his rule-based system on a few years later (Briem, 1990, 2009).

The sample Icelandic sentence *Allir stóru strákarnir borðuðu góðu súpuna* (English literal translation: All big boys-the ate good soup-the) gave almost perfect results on April 17th 2010 (see figure 2.3).

Figure 2.3: *Sample sentence tested on Tungutorg.*

All the words were correctly translated and all but two are in the correct position, i.e. if the definite article *the* and the adjective *all* exchange their places then the sentence would be perfectly translated. The sentence was re-tested on November 7th 2010, but the system still gave the same result as in April.

2.6.3 Google Translate

Google Translate (Google, 2010) is a web-based translation service provided by Google Inc. It is a statistical machine translation system, using very large corpora and parallel texts. They added Icelandic to their language pairs and launched it in August 2009 (mbl.is, 2010).

When the sample Icelandic sentence *Allir stóru strákarnir borðuðu góðu súpuna* (English literal translation: All big boys-the ate good soup-the) was put into the system on launch day, the result was *All major boys ate good soup*, while on April 17th 2010 the result was much better (see figure 2.4).

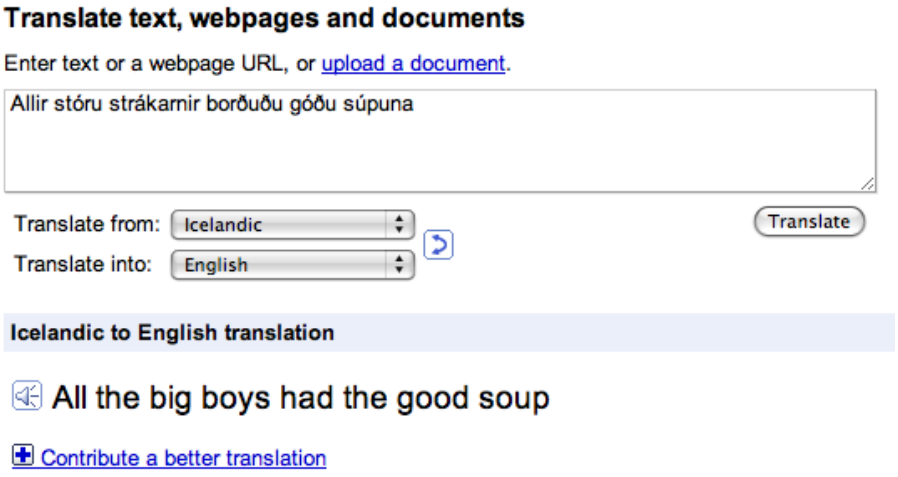

Figure 2.4: *Sample sentence tested on Google Translate.*

Although the translation shown in figure 2.4 could be considered adequate in the sense that when a person "**has** some food", that implies that the person has "eaten" it. However, this is not an accurate translation. The sentence was re-tested on November 7th 2010, but the system still gave the same result as in April.

2.7 Summary

The purpose of LT is to apply technology to language in order to increase the user's knowledge base. The area of LT that concerns this thesis is MT. The theory of MT actually predates computers as far back as the seventeenth century, and the first patents for 'translating

machines' appeared in the early 1930s. After computers came onto the scene, the success of the Georgetown-IBM experiment in 1954 was blown out of proportion causing high expectations that the challenge of automatic machine translation would soon be solved, attracting funds and scientists. As time revealed that the task was more complex than initial speculations suggested, the infamous ALPAC report in 1966 pronounced that MT was worse than useless: it was a waste of time and money, bringing nearly all work in MT to a halt.

As more efforts were put into MT, the scope of difficulties with MT became more clear. First of all, translating from one language to another is no trivial task for humans. Secondly, the quality requirements are hard to define, e.g. when is a translation considered 'equally good' as the source. And thirdly, there are certain limitations to the abilities of computers: they cannot perform vaguely specified tasks, they are unable to learn things, they cannot perform common-sense reasoning and are unable to deal with combinatorially explosive problems. Since one-to-one translations for all possible words between any two languages do not always exist, there arises the problem of ambiguity. The ambiguity may be lexical in nature, structural or due to anaphoric expressions. If the sentence is taken out of context, then neither human nor computer can select the correct translation. The computational limitations are slowly being overcome, but the issue of resolving ambiguity will remain the biggest problem with MT as long as humans continue to have difficulty with it themselves.

There are different uses for MT; assimilation is meant to provide a general, informal meaning of some text while dissemination is meant to be a more grammatically correct translation. Various methods have been developed within MT in the quest to produce the sought-after results; rule-based MT, statistical MT and example-based MT. Rule-based MT utilizes linguistic rules in its methods and can be further categorized into dictionary-based, interlingual and transfer-based methods. Dictionary-based methods perform word-for-word look-up, interlingual methods use an intermediate abstract language representation, and transfer-based methods generally perform morphological analysis, lexical transfer, structural transfer and morphological generation. Transfer-based methods are further categorized as deep-transfer, which requires full parsing and disambiguation, or shallow-transfer, which works on partial sentence chunks. Statistical based MT methods are just that: methods based on statistics, while example-based MT methods are not easily defined but are somewhere between RBMT and SMT.

There are also several methods available for evaluating MT systems. Described here were the F-score, BLEU, WER and PER. The F-score evaluation method is just as applicable as the WER and PER methods for evaluating MT systems and may even be better for measuring translation accuracy, providing a score between 0 and 1, but the WER and PER provide instantly recognizable scores as percentages, i.e. numbers of words per 100 words of running text. The BLEU method requires multiple reference translation sentences per each SL sentence directly translated by different sources without relying on the TL translations. Since BLEU's brevity penalty can prove extreme for short sentences, the method should be calculated over a whole corpus instead of sample sentences. Furthermore, since the language skill behind the reference translations influence the outcome of the score, e.g. comparing poor reference translations to poor TL translations which have similar output will provide a high score, this method is not reliable to measure improvement in translation quality of MT systems.

MT is a relatively young research field in Iceland. The first official steps were taken in 1999 towards bringing LT to the Icelandic language and within ten years a number of LT resources had been produced: a database of modern Icelandic inflections, a tagged 25 million word corpus, a training model for data-driven POS taggers, a text-to-speech system, a speech recognizer, an improved spell-checker, a POS tagger, a lemmatizer, a shallow parser and a context-sensitive spell-checker. Most of these resources would not have come into existence if not for the Icelandic Frequency Dictionary, a balanced corpus of roughly 600 thousand tokens which was first published in 1991. The first MT system for Icelandic, Tungutorg, is a rule-based MT system which became publicly available in March 2008. Google launched their Icelandic statistical MT system in August 2009 and InterTran published their hybrid system also in 2009.

This research project commenced in February 2009 with the aim of creating an Icelandic to English MT prototype which makes use of existing LT resources. The next chapter describes this project.

Chapter 3

Project description

This project aims to further the Icelandic BLARK with the creation of a prototypical MT module that translates from Icelandic to English. Such a module has not previously been present in said BLARK, and thus it is the intent that this prototype lend its weight to the advancement of Icelandic LT.

3.1 Goals

In the process of creating this prototype, we will seek to fulfill three main objectives:

1. To find the most economic methods for creating the rules and data needed for a successful implementation of a full STMT system (see section 2.3).
2. To find ways of incorporating existing LT tools into the Apertium platform.
3. To use these means to develop a prototype of an STMT system.

Additionally, the focus is on translating from a minor inflectional language to a (non-inflectional) major language, and so it is our hope that the work here will also be beneficial for other (inflectional) languages.

3.2 Existing LT Tools

Here we will look at the modules of the IceNLP (Loftsson & Rögnvaldsson, 2007a) and the Apertium (Armentano-Oller et al., 2005) platform, readily available LT tools. We will discuss which modules were used in this project, which were substituted and why in section 3.4.

3.2.1 IceNLP

Some of the current LT tools that exist for Icelandic are encompassed in IceNLP, the Icelandic Natural Language Processing toolkit. The IceNLP modules are six:

1. a tokenizer and sentence segmentizer, which transforms a stream of characters into linguistic units and groups tokens into sentences.

2. *IceMorphy* (Loftsson, 2008), a morphological analyzer, which guesses POS tags for unknown words. It automatically fills in tag profile gaps in the lexicon by performing morphological analysis, compound analysis and ending analysis.

3. *IceTagger* (Loftsson, 2008), which is a linguistic rule-based POS tagger that uses the IFD. It uses a small number of local elimination rules along with a global heuristics component that guesses the functional roles of the words in a sentence, marks prepositional phrases, and uses the acquired knowledge to force feature agreement where appropriate, e.g. gender of an adjective preceding a noun.

4. *TriTagger* (Loftsson, 2006),which is a data-driven re-implementation of the TnT trigram tagger (Brants, 2000) in Java, driven by the IFD.

5. *IceParser* (Loftsson & Rögnvaldsson, 2007b), which is a shallow parser[1] with incremental finite-state transducers[2]. The parser consists of a phrase structure module and a syntactic functions module. The two modules are comprised of a sequence of finite-state transducers, each of which adds syntactic information into substrings of

[1] A shallow parser analyzes sentence input to identify the linguistic constituents (noun phrase, verb phrase, etc) but does not carry out full parsing to finalize the whole sentence structure.

[2] A finite state transducer is a finite state machine (similar to a flowchart) that takes some input and generates output using a set of actions.

the input text.

6. *Lemmald* (A. Ingason et al., 2008), which is a mixed method lemmatizer. It uses a Hierarchy of Linguistic Identities (HOLI) approach, combining data-driven machine learning with linguistic insights, to determine a word's lemma[3]. Furthermore, it uses an add-on which connects to the Database of Modern Icelandic Inflections (Bjarnadóttir, 2005) to improve lemmatization accuracy.

Note that the latest version of IceTagger uses TriTagger to achieve higher accuracy, see (Loftsson, Kramarczyk, Helgadóttir, & Rögnvaldsson, 2009).

3.2.2 Apertium

Apertium (Armentano-Oller et al., 2005) is an open source Shallow Transfer Machine Translation (STMT) platform, created in 2005 by a team from the department of Languages and Information Systems in the University of Alicante, that was originally made for translating between closely related languages such as Spanish-Catalan.

It has since been developed to handle less related languages, and as of November 2010 has 25 language pairs in 40 directions that have been released[4] and 17 more that are in various stages of construction (Apertium, 2010b).

The modules of the Apertium pipeline are nine (Armentano-Oller et al., 2006):

1. a de-formatter, which encapsulates markup of websites and documents, e.g. HTML and RTF, in brackets to be ignored throughout the rest of the pipeline.
2. a morphological analyzer, which is all-in-one a sentence segmentizer, tokenizer, lemmatizer and retrieves all possible POS tags for each word. It tokenizes the surface forms[5] of the words in the SL text and delivers one or more lexical forms[6]. Each surface form is delimited with a caret '^' at the start of the string token and a dollar-sign '$' at the end, and if more than one lexical form is returned then they

[3] A lemma is the dictionary look-up form of a word.

[4] Not all language pairs can be translated in both directions, which is why the number of language pairs and language directions are both specified.

[5] The surface form of a word is how it appears in running text.

[6] The lexical form of a word consists of a lemma, word category and inflection information.

are delimited by a forward slash '/', all in one string token. The example sentence "*Bíllinn minn er rauður.*" ("*My car is red.*") looks like this:

```
^Bíllinn/bíll<n><m><sg><nom><def>\$
^minn/minn<prn><pos><m><sg><nom>\$
^er/vera<vbser><pri><p3><sg>\$
^rauður/rauður<adj><pst><m><sg><nom><sta>\$
^./.<sent>\$
```

3. a statistical POS tagger, used to resolve ambiguity, i.e. selects one tag for each word. The Apertium POS tagger is a first-order hidden Markov model (HMM), which has to be trained on representative SL texts before it can be used, either a large amount of untagged text processed by the morphological analyzer (millions of words) or a small amount of tagged text (tens of thousands of words).

4. lexical selection, this module is not fully developed yet, but the general idea is to select the correct translation based on sentence context. The Apertium group is developing this module with VISL-CG3, a constraint grammar tool created at the University of Southern Denmark (Didriksen, 2010). An attribute is added to the opening element tag (e.g. '<e slr="1">') in any bilingual dictionary entries that have more than one translation, so that when this new module searches an additional dictionary for lexical selection, it will find patterns that indicate which of the translations in the bilingual dictionary should be selected. Here is an example from such a dictionary:

   ```
   # "dalur"     : {0: "valley", 1: "dollar"};
   SUBSTITUTE ("dalur") ("dalur:1") ("dalur") + N (-1 Num);
   ```

   In the above example, the former line is a comment showing that the translation number 0 for "dalur" should be "valley" (zero is always the default), while an alternate translation (number 1) for "dalur" will be "dollar". The second line shows the pattern which will trigger the substitution: the first set of parentheses is the tag to be located, the second set of parentheses is the replacement tag to be inserted in place of the third parentheses where the rest of the line is the contextual pattern that must be matched in order for the rule to be triggered[7].

5. lexical transfer, is called by the structural transfer module and uses the disambiguated SL lemma to look up the translation in the bilingual dictionary (see section 4.1).

6. structural transfer, is used for moving chunks around in the sentence structure. It is compiled from the transfer rule files (see section 4.2) and uses finite-state pattern

[7] Substitute ("this") ("with this") ("here") if "here" is a noun and a number is in the next position before "here" in the sentence.

matching to detect fixed-length patterns of lexical forms that need special processing due to grammatical divergences between the SL and the TL.

7. a morphological generator, which will produce the appropriate surface form of the word in the TL by using the assigned tag. It uses the binary file compiled from the TL morphological dictionary, also called a monolingual dictionary.
8. a post-generator, which applies final touch ups, such as contractions and insertion of apostrophes, e.g. "can not" becomes "can't". This module is usually dormant and returns the input unchanged unless it encounters an alarm symbol ('<a/>') which performs the particular string transformation that matches one of the rules in one of the rule files.
9. a re-formatter, which removes the encapsulations surrounding the markup that the de-formatter applied.

Simplistically put, the only thing needed for the Apertium platform to become an MT system is some data, basically three dictionaries and some transfer rules (see 3.3).

3.3 Pure Apertium

As mentioned in section 3.2.2, the only thing needed to transform the Apertium platform (see figure 3.1) into an MT system is some data. This is a largely simplified statement as the task does involve considerable manual work to varying degrees of complexity. This section will discuss said data and its format. Furthermore, it will discuss the similarites with a pure Apertium system and the prototype Apertium-IceNLP hybrid system.

A pure Apertium system needs three dictionaries and some transfer rules to become operational as an MT system. Two monolingual dictionaries (sometimes referred to as a 'monodix') - one for the SL and another for the TL - and a bilingual dictionary, which we call a 'bidix'. The data must be in an XML based format defined through XML document type definitions (DTD) which are part of the Apertium package.

The monolingual dictionaries must contain (Armentano-Oller et al., 2006):

i) a definition of the language's alphabet, which is used by the tokenizer;

ii) a section containing definitions of the grammatical symbols used to represent concepts such as nouns, verbs, plural, masculine, indefinite, etc.

iii) a section defining paradigms, which describe reusable groups of correspondences between parts of surface forms and lexical forms. Paradigms represent regularities in an inflective language.

iv) one or more labelled dictionary sections with lists of surface form to lexical form correspondences for whole lexical units, including contiguous multi-word units.

The bidix has a similar structure to the monolingual dictionaries, except it contains pairs of SL lexical forms and TL lexical forms. Additionally, Apertium needs one or more transfer rule files which contain pattern-action rules describing what action happens if the pattern is matched. For example, a common pattern rule is the '*determiner-noun*' pattern, which ensures that the gender and number of the noun and its determiner are in agreement.

The Apertium-IceNLP hybrid prototype shares the bidix, the English TL monodix and the transfer rules with the pure Icelandic-English Apertium system. The pure Apertium system additionally uses an Icelandic SL monodix, which only exists because of the work of Francis Tyers, who is a non-native speaker of Icelandic. Therefore, I find it rather impressive at how far he has gotten with it.

3.4 Apertium-IceNLP

As was mentioned in section 3.1, one of the objectives of this research project was to use existing LT tools for Icelandic. The reasoning behind this was of course to avoid "re-inventing the wheel" and the belief was that tools that had been constructed for Icelandic, using Icelandic grammar rules, should theoretically perform better on Icelandic than other comparable tools, even if they were said to be language-independent (Loftsson et al., 2009; Kramarczyk, 2009).

With that in mind, the Icelandic Centre for Language Technology began collaborating with the University of Alicante to use their Apertium MT platform, because the platform is a pipeline of LT modules which are potentially interchangeable with comparable modules, thus allowing for the substitution and utilization of at least some of the Icelandic

LT tools.

The idea was to create a hybrid system by combining modules from IceNLP and Apertium, and despite our discovery that the substitution of modules was not quite as straightforward as it was first thought to be, we were still successful in integrating most of IceNLP's modules into the MT prototype. However, the four IceNLP modules that we used (tokenizer/sentence segmentizer, the morphological analyzer IceMorphy, the POS tagger IceTagger, and the lemmatizer Lemmald) actually only replaced two of the Apertium modules: the morphological analyzer and the POS tagger. In the early stages of this project, we skipped the Apertium de-formatter and re-formatter modules, as they were not needed with the standard IceNLP at that time. Later, when Hlynur Sigurþórsson modified IceNLP into a daemonized version (see section 4.3), those two modules came back into play: Hlynur rewrote the de-formatter to output the data such that IceNLP could receive it and the re-formatter was added to its original place at the end of the Apertium pipeline.

Thus, the Apertium-IceNLP hybrid prototype system consists of the following modules which are used in this order: first the IceNLP modules de-formatter, tokenizer/sentence segmentizer, IceMorphy, IceTagger, Lemmald are used, and then the system switches over to the Apertium lexical transfer, structural transfer, morphological generator, post-generator and re-formatter modules, see figure 3.2.

Although the pure Apertium MT system (see section 3.3) could translate in either direction, e.g. from Icelandic to English and from English to Icelandic (if provided with the system's three required dictionaries and transfer rules for the language pair), the prototype Apertium-IceNLP hybrid MT system can only translate from Icelandic to English.

3.5 Summary

The overall goal of this project was to provide a useful MT module for the Icelandic BLARK by way of three objectives: find economic methods for creating the necessary rules and data for a full STMT system, find ways of incorporating existing LT tools for Icelandic into the Apertium platform, and use those means to develop a prototypical

STMT system for translating from Icelandic to English.

Many state-of-the-art LT tools for processing Icelandic text are found in the IceNLP toolkit. These are a tokenizer and sentence segmentizer; a morphological analyzer called IceMorphy; a linguistic rule-based POS tagger called IceTagger; a data-driven re-implementation of the TnT trigram tagger, called TriTagger, which uses the Icelandic Frequency Dictionary; a shallow parser with incremental finite-state transducers called IceParser; and a mixed method lemmatizer called Lemmald.

The Apertium shallow transfer MT platform is an open source framework, where some of its modules are similar to those in IceNLP: an extended morphological analyzer which is also a sentence segmentizer, tokenizer, lemmatizer and it retrieves all possible POS tags for each word as well; and a statistical POS tagger. Apertium additionally contains these modules which are not present in IceNLP: a de-formatter; a prototypical lexical selection module; a lexical transfer module; a structural transfer module; a morphological generator; a post-generator; and a re-formatter. In addition to the Apertium platform modules, a pure Apertium MT system can translate in either direction provided it has the required three dictionary files, i.e. two monolingual and one bilingual, and some files with transfer rules.

As one of the objectives for this project was to incorporate existing Icelandic LT tools into the Apertium platform, the relevant available Icelandic modules were substituted for the comparable Apertium ones to create an Apertium-IceNLP hybrid system in the hope that using state-of-the-art modules for Icelandic would produce better results. The hybrid system also requires transfer rules, a bilingual dictionary, but only one monolingual dictionary, i.e. the English TL one, since the IceNLP modules do not require a dictionary to perform their jobs. This means that the hybrid prototype only translates from Icelandic to English. The next chapter looks at how the prototype system was developed.

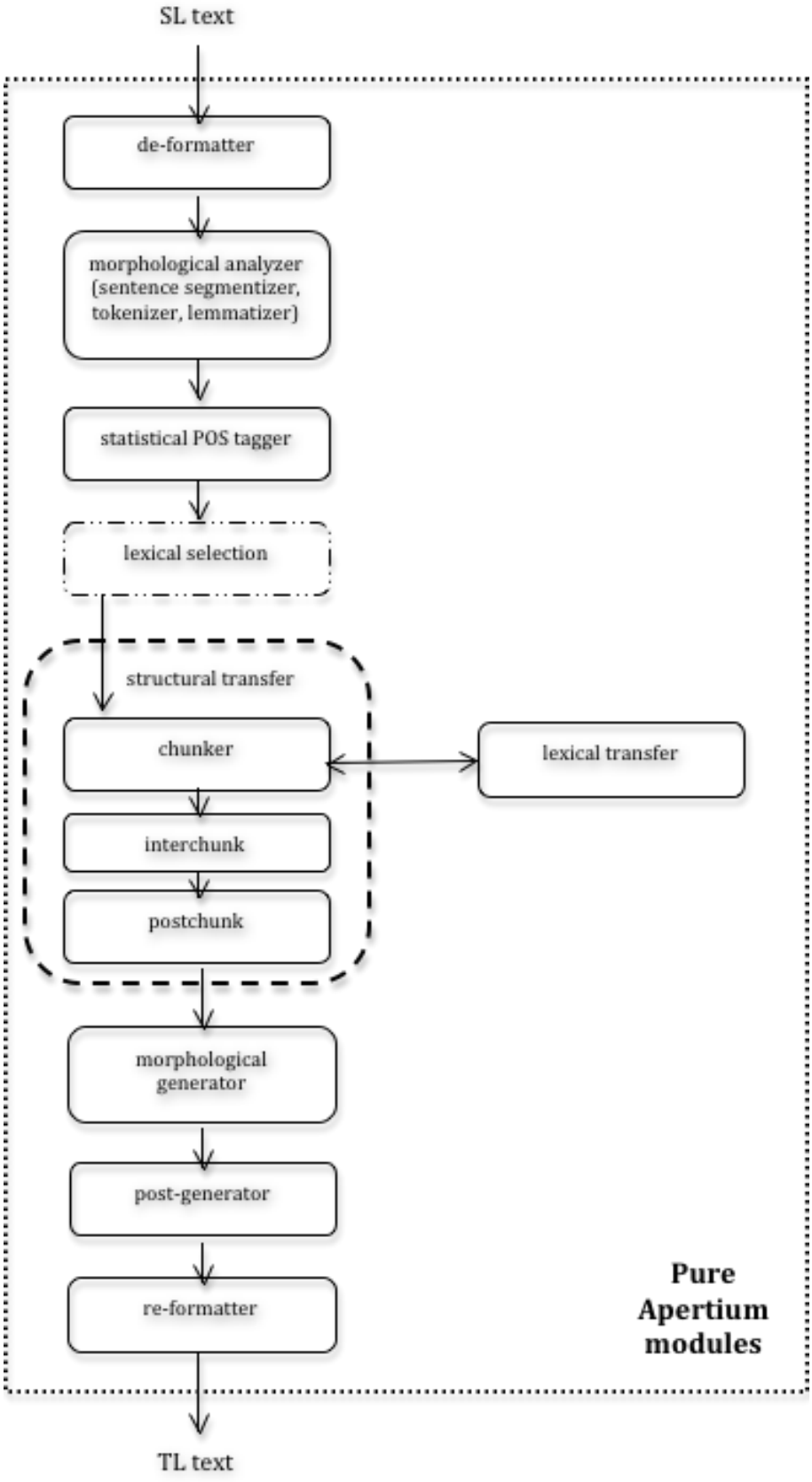

Figure 3.1: *The modules of a pure Apertium system.*

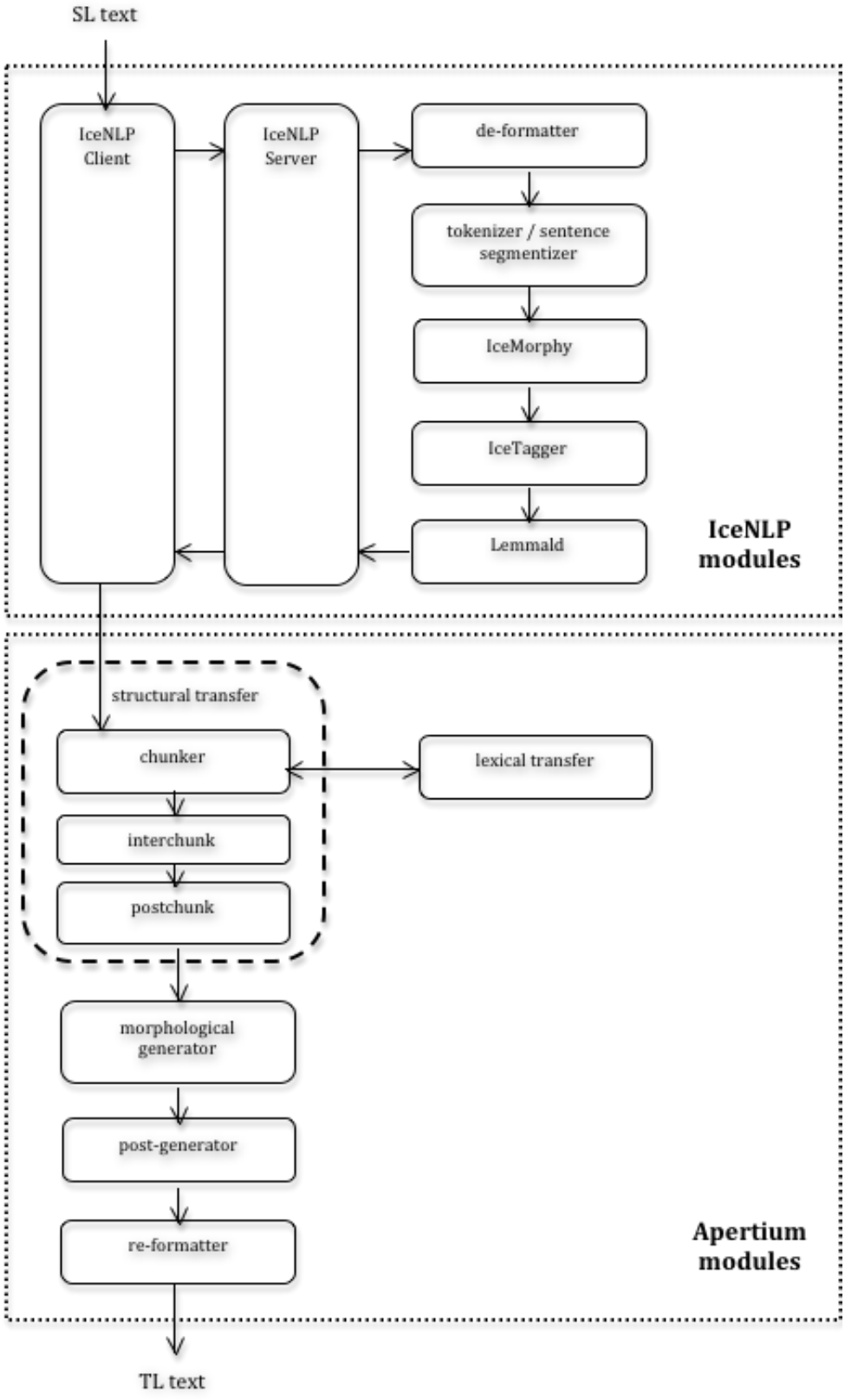

Figure 3.2: *The Apertium-IceNLP hybrid prototype.*

Chapter 4

System development

This chapter describes my contribution in the development of the Apertium-IceNLP prototype system as well as an introduction to some work contributed to the project by Francis Tyers, Hlynur Sigurþórsson and Dr. Hrafn Loftsson. Presented are descriptions of the initial creation of the bilingual Icelandic-English dictionary, the process of expanding it, an example of the transfer rules which are e.g. used for word re-ordering, and modifications that had to be made to fit IceNLP modules into the Apertium framework.

The largest contribution to this project in terms of time consumption was undoubtedly the work on the bilingual dictionary. This was mainly repetitive manual work, where I had to assess the correctness of the first set of Icelandic-English translations (see 4.1). I also added multiword expressions (MWEs) to the prototype (see section 4.4) and Francis helped me add transfer rules to the system. Next, additional data was acquired which I then had to pre-process - with some obstacles to overcome on the way - before converting it to the appropriate format and finally proofread as well (see section 4.1.1). It should be noted that all contributions to the dictionaries and transfer rules used in the Apertium-IceNLP hybrid prototype can also be used by the pure Apertium version.

I also performed evaluation to assess the status of the system and to attempt to identify the types of errors made by the system. The evaluation will be discussed in Chapter 5.

4.1 Bilingual dictionary

The bilingual dictionary for Icelandic-English was originally populated by Francis M. Tyers with entries spidered from the internet from Wikipedia, Wiktionary, Freelang, the Cleasby-Vigfusson Old Icelandic dictionary and the Icelandic Word Bank (Apertium, 2010a). This provided a starting point of over 5,000 entries in Apertium style XML format which needed to be checked manually for correctness. Also, since lexical selection was not an option in the early stages of the project, only one entry could be used. SL words that had multiple TL translations had to be commented out, based on which translation seemed the most likely option according to the author.

The manual revision of the bilingual dictionary entries took approximately 4 months, which primarily involved checking whether each translation was correct, but also e.g. whether the gender was correctly assigned. If an entry contained more than one TL translation, then that entry had to be copied and each resulting copy had to be modified to contain only one translation. Furthermore, since the system did not have the ability to carry out lexical selection it only allowed for one-to-one translations, therefore I had to assess which entry should be the default translation and comment out all additional entries.

Example of an entry in the Icelandic-English bilingual dictionary[1]:

```
        <e><p>
          <l>jafngilda<s n="vblex"/></l>
          <r>equal<s n="vblex"/></r>
        </p></e>
 <!--
        <e><p>
          <l>jafngilda<s n="vblex"/></l>
          <r>amount<g><b/>to</g><s n="vblex"/></r>
        </p></e>
        <e><p>
          <l>jafngilda<s n="vblex"/></l>
          <r>mean<s n="vblex"/></r>
        </p></e>
   -->
```

[1] Indenting used here for readability, as the dictionary entries (which are usually on one line) could not fit across the page.

The above example shows the entry in the bilingual dictionary for the Icelandic verb *jafngilda*, with the default English translation followed by two alternative translations commented out between '<!–' and '–>' tags. Each dictionary entry is encased in '<e>... </e>' tags. Within the entry tags is an SL-TL pair, encased in ' ... ' tags. There you will find the SL entry on the left side within '<l>...</l>' tags and the TL counterpart on the right side within '<r>...</r>' tags. In addition to the <e>, ,<l> and <r> tags which are universally used for all word categories, anywhere a blank space is needed in either an SL or TL entry it is represented with ''. The '<g>...</g>' tags contain the non-inflecting part of a multiword. Furthermore, each word category is represented with the 'n' attribute of the lexical symbol tag 's'; in this example "vblex" represents a standard verb.

4.1.1 Additional data

A bilingual Icelandic-English wordlist of approximately 6,000 SL words with wordclass and gender was acquired from an LT colleague, Anton Karl Ingason, which required some preprocessing before it could be added to the prototype's bilingual XML format dictionary.

The first step was to determine which of these new SL words did not already exist in the bilingual dictionary. In order to do that, I wrote some scripts and perl programs, and the result was a list of approximately 4,000 SL words that were transformed into the Apertium style XML format, one entry per SL word, sorted by wordclass. However, each SL word usually had more than one possible translation, thus I also had to manually review and modify these 4,000 entries, resulting in approximately 7,100 additional entries for the bilingual dictionary.

The format of Anton's data shown in table 4.1 is set up such that each line is made up of three parts. The first part is the SL token (word, number or punctuation mark) delimited by an underscore, the second part is the POS tag delimited by a semicomma, and the third part is the TL translation. The translation part may consist of one or more options, e.g. 'jealous', 'envious', 'green-eyed' and 'resentful'; it may indicate with an asterisk that the SL word is ungrammatical, e.g. '*Afghan' ('afgönskur' is not a valid Icelandic word); or it may contain multiword translations, e.g. 'Member of Parliament'.

af_aa;by,from
af_aþ;of,by
afar_aa;very
afbrigði_nh;variant,variation,variety,type,version
afbrot_nh;crime
afbrýðisamur_l;jealous,envious,green-eyed,resentful
afgönskur_l;*Afghan
alþingismaður_nk;congressman,Member of Parliament

Table 4.1: *Example of additional data from Anton Ingason.*

Later another Icelandic-English bilingual wordlist was acquired from the dictionary publishing company Forlagið, which was an excerpt of approximately 18,000 SL words from their online Icelandic-English dictionary (www.snara.is). This required similar handling to the previous wordlist, i.e. I had to preprocess the data since the format differed from that of the previous wordlist. First I purged out some unnecessary symbols in the text, e.g. HTML tags. Then in order to determine which words were missing from the prototype's bilingual dictionary, I extracted a list of just the SL words from Snara's list which I then ran through the prototype to identify the unknown words. Almost 5,000 of the input words already existed in the bilingual dictionary.

At this point I decided to start with words from open word categories (i.e. nouns, verbs, adjectives, adverbs) and so the 13,000 words got further reduced to about 12,000, shown here in table 4.2.

Interim word count	*Interim word category*
2,125	adjectives
177	adverbs
8,022	nouns
1,673	verbs
11,997	Total

Table 4.2: *Interim distribution of the Snara data, by word category.*

More problems were encountered and dealt with, e.g. I had to remove lines that did not contain a translation and lines that had missing closing brackets. These corrections and the closed word categories (that were not used) accounted for the further reduction to just about 11,200 words.

After investigating another set of problems, I had to make some more reductions due to multiple comments (in parenthesis), bringing the final number of SL words from Snara's

data to be added to the bilingual dictionary to approximately 10,800.

abbast ofl:v
að ofl:prep/adv ofl:prep towards, up to in
að ofl:conj that, so that
að to
aðalatriði ofl:n main point, main thing, main issue
aðaláhersla ofl:f main emphasis
aðalbláber ofl:n whortleberry
aðalbláberjalyng ofl:n whortleberry (bush)
aðdragandi ofl:m antecedents, events leading up to sth preparation
aðhyllast ofl:v subscribe to, endorse ([[a view]])
aðild ofl:f participation, membership, affiliation
aðili ofl:m party ([[in an issue]])
aðkeyptur ofl:adj bought elsewhere, brought in/hired from elsewhere
aðnjótandi
aðrar
aðrein ofl:f entrance ramp, ([[UK]]) slip road
aðsjálni tight-fistedness
afgreiðslumaður ofl:m salesman, (UK) (shop) assistant

Table 4.3: *Example of additional data from the online dictionary Snara.*

The format of Snara's data shown in table 4.3 is also set up with three parts on each line, similar to Anton's data. Some entries were incomplete, i.e. had fewer parts and were not useful, like 'aðnjótandi' and 'aðsjálni tight-fistedness'. Other entries contained superfluous characters, e.g. '[[' and '', that had to be removed to make the entry useful. The first part is the SL word delimited by a space, the second part is the word category suffixed to 'ofl:' and delimited by a space, and the third part is the TL translation.

The format of the translation part is multifaceted depending on whether the translation consists of:

- one or more options with the same meaning, e.g. 'antecedents' vs 'events leading up to sth';
- one or more alternative meanings, e.g. on the one hand 'antecedents, events leading up to sth' and on the other hand 'preparation';
- or optional explanations are included in parentheses, e.g. '(UK) (shop) assistant'.

The resulting difference of approximately 10,800 entries were saved to a new file and then I wrote another Perl program to convert those entries to the Apertium style XML format.

I had to yet again manually revise the results of that style change, which yielded approximately 12,300 additional entries to the prototype's bilingual dictionary - the expansion caused by splitting the multiple translations into separate entries (see table 4.4).

Final word count	*Final word category*
2,203	adjectives
154	adverbs
8,343	nouns
1,613	verbs
12,313	TOTAL

Table 4.4: *Final distribution of the Snara data, by word category.*

4.2 Transfer rules

The Apertium structural transfer module consists of transfer rules that are grouped into stages. Closely related languages may only require one stage of transfer rules while other language pairs usually have three or more stages. Our prototype has five stages. Two short examples of transfer rules are provided on the next pages, both have to do with handling the presentation of a reference to a year.

Transfer rules consist of a pattern section "<pattern>...</pattern>" and an action section "<action>...</action>". The pattern section indicates which words in what order will trigger the rule. The action section often starts by calling a macro with "<call-macro>...</call-macro>", which is a subroutine or rule that is carried out before anything that follows. Then a section to state some action with "<let>...</let>" and a section indicating the output of the rule is within the "<out>...</out>" tags.

Let us now look closer at the functionality of the inner workings of the first rule, let us call it "Rule A". The pattern matches on a preposition, the word "*árið*", followed by a number. At the beginning of each transfer rule file is a section containing the definitions of the lexical categories used to identify the rule patterns. Each definition (called a '<def-cat>') contains a list of categories that define it (called a '<cat-item>'). For the word "*árið*", the <cat-item> identifies the lemma "*ár*" in the neutral gender, singular, in any case and it must have the definite article:

```
<cat-item lemma="ár" tags="n.nt.sg.*.def"/>
```

Continuing down the rule, the macro "firstWord" uses the word in position one as its parameter and determines whether it is the first word in the sentence or not in order to use a capital letter if appropriate. The let action pastes together the lemma part of the TL such that the lexical unit in position 1 connects to position 3 with an underscore and 'drops' the one from position 2. The output is one chunk that contains the tag "ADV" followed by the whole first lexical unit (which was pasted together in the let section), then a space identified as being in space-position 1, followed by a second lexical unit which is comprised of the lemma head information, the number and the empty lemma queue information[2].

The following rule, **Rule A**, transforms for example "*fyrir árið 2010*" (which literally translated means "*before the year 2010*") into "*before 2010*":

```
<rule comment="REGLA: PREP árið NUM -> PREP NUM">
   <pattern>
     <pattern-item n="prep"/>
     <pattern-item n="árið"/>
     <pattern-item n="num"/>
   </pattern>
   <action>
     <call-macro n="firstWord">
       <with-param pos="1"/>
     </call-macro>
     <let>
       <var n="cur_adv_lema"/>
       <concat>
         <clip pos="1" side="tl" part="lem"/>
         <lit v="_"/>
         <clip pos="3" side="tl" part="lem"/>
       </concat>
     </let>
     <out>
       <chunk namefrom="cur_adv_lema" case="caseFirstWord">
         <tags>
           <tag><lit-tag v="ADV"/></tag>
          </tags>
```

[2] The lemma head and lemma queue are used to distinguish between lexical forms that inflect. The lemma queue is the invariable part, but if the lemma is not a split lemma then it will be sent forward empty. This way it is not necessary to define different rules for lexical forms with and without queue.

```
            <lu>
              <clip pos="1" side="tl" part="whole"/>
            </lu>
            <b pos="1"/>
            <lu>
              <clip pos="3" side="tl" part="lemh"/>
              <clip pos="3" side="tl" part="a_num"/>
              <clip pos="3" side="tl" part="lemq"/>
            </lu>
          </chunk>
        </out>
    </action>
</rule>
```

The functionality of the following second transfer rule, which we shall call **Rule B**, is similar to the one above, just replacing the word in position 1 with "in" instead. This rule is however more specific, as it only transforms the word "árið" followed by a number, e.g. 2010 (literally "*the year 2010*") into "*in 2010*".

```
<rule comment="REGLA: árið NUM -> in NUM">
   <pattern>
     <pattern-item n="árið"/>
     <pattern-item n="num"/>
   </pattern>
   <action>
     <call-macro n="firstWord">
       <with-param pos="1"/>
     </call-macro>
     <let>
       <var n="cur_adv_lema"/>
       <concat>
         <lit v="in_"/>
         <clip pos="2" side="tl" part="lem"/>
       </concat>
     </let>
     <out>
       <chunk namefrom="cur_adv_lema" case="caseFirstWord">
         <tags>
           <tag><lit-tag v="ADV"/></tag>
         </tags>
```

```
            <lu>
              <lit v="in"/>
              <lit-tag v="pr"/>
            </lu>
            <b pos="1"/>
            <lu>
              <clip pos="2" side="tl" part="lemh"/>
              <clip pos="2" side="tl" part="a_num"/>
              <clip pos="2" side="tl" part="lemq"/>
            </lu>
          </chunk>
        </out>
    </action>
</rule>
```

4.3 Adapting IceNLP

One of the goals of this project was to make use of the existing Icelandic LT tools and in order to do that, some of those modules had to be slightly modified. Most of the work here was done by Hlynur Sigurþórsson and Dr. Hrafn Loftsson.

Firstly, IceNLP had to be made open source to be used in conjunction with the Apertium platform as that was their condition for co-operation. Subsequently, the toolkit was transformed into a daemonized version such that it no longer had to start up every time an input string was sent to the system (Sigurþórsson, 2010). So now it has a server version running in the background and input is sent through client threads.

The IceNLP toolkit is now available under the LGPL license at
http://sourceforge.net/projects/icenlp/.

Other changes mostly involved analyzing the input and output of the Apertium modules and modifying the respective IceNLP modules to handle the same format. For example, the Icelandic tagset consists of approximately 700 POS tags where each character in the token string represents an identification element, whereas the Apertium POS tags have

each element enclosed within the symbols < and > (see table 4.5).

SL text	*IFD POS tag*	*Apertium POS tag*	*Description (IFD order)*	*TL text*
Allir	fokfn	<prn><qnt> <m><pl><nom>	Pronoun, quantative, masculine, plural, nominative case.	All
stóru	lkfnvf	<adj><pst><m> <pl><nom><vei>	Adjective, masculine, plural, nominative case, weak declension, positive degree.	big
strákarnir	nkfng	<n><m><pl> <nom><def>	Noun, masculine, plural, nominative case, definite article.	boys-the
borðuðu	sfg3fþ	<vblex><actv> <past><p3><pl>	Verb, indicative mood, active voice, 3rd person, plural, past tense.	ate
góðu	lveovf	<adj><pst><f> <sg><acc><vei>	Adjective, feminine, singular, accusative case, weak declension, positive degree.	good
súpuna	nveog	<n><f><sg> <acc><def>	Noun, feminine, singular, accusative case, definite article.	soup-the
.	.	<sent>	Punctuation mark.	.

Table 4.5: *Relationship between IFD POS tags, Apertium XML style POS tags and the tags' underlying meaning.*

4.4 Multiword Expressions (MWEs)

This section describes how MWEs[3] were added to the prototype system, as there were some discrepancies between the IceNLP modules and Apertium modules regarding MWEs, e.g. there were MWEs in the bilingual dictionary, but they were not recognized as single units in the IceNLP part of the prototype.

In order to be able to use these MWEs from the bilingual dictionary, they needed to be added to the otb.apertium.dict file, which IceTagger uses to map the IFD style POS tags to the Apertium XML style POS tags (this tag-mapping file did not originally have a section for MWEs). In order to be able to add them to the tag-mapping file, each MWE had to have a mapping between the set of individual words' POS tags and a single-unit POS tag for the MWE.

The bilingual dictionary MWEs already had single-unit POS tags, so for these entries I needed to find the set of individual words' POS tags for each MWE. First I extracted all MWEs from the bilingual dictionary and stripped them of their XML formatting, which produced 515 entries. Then I had to manually review and clean up that list, e.g. removing entries that had the MWE on the TL side, consisted of temporal phrases (such as "*klukkan átta*" ("eight o'clock")), or unconventional and/or unlikely MWEs (such as "*tónleikaferð um heiminn*" ("musical world tour")), which left 286 MWEs. These 286 entries were then tagged with IceTagger to get the POS tag for each individual word, which in turn had to be cross-referenced with the IFD gold standard to ensure consistency in the POS tagging. So, in order to identify MWEs in the IFD, I had to first transform the tagged IFD file, which had one token and its respective POS tag on each line, into one sentence per line with the POS tags still in their respective places.

Then I wrote a Perl program that took the 286 MWEs from the bilingual dictionary and compared them to the sentence-per-line version of the tagged IFD and wrote the matching lines to a file for manual reviewing. The manual reviewing of the MWEs resulted in 110 MWEs with unambiguous POS tags, of which 82 were copied to otb.apertium.dict tag-mapping file:

[3] Note that 'MWE' is used in this project in a very loose sense, i.e. it is used for any phrase that consists of more than one word. Here "*af hverju*" ("why") is considered a MWE.

- 25 MWEs had inconsistent tags (either IFD and IceTagger did not match, or sometimes IFD had more than one tagging result), and were therefore not used.
- 3 MWEs consisted of four or more words, which IceTagger currently does not handle, and were therefore not used.
- 18 MWEs were trigrams, i.e. consisted of three words, and were added to the file.
- 64 MWEs were bigrams, i.e. consisted of two words, and were added to the file.

Although IceParser was not incorporated into the hybrid prototype, it contains information regarding MWEs in an idioms dictionary file, and so I used that for reference and possible additions in this MWE creation phase of the prototype. Therefore, I used the code for MWE phrases in IceParser to create all possible MWEs known to the IceNLP toolkit, which were 124 and got the single POS tag for the MWEs at the same time.

Then I extracted all MWEs from the idioms dictionary, which were 346, and manually marked 89 non-MWE entries [4] to be excluded from further processing. These idioms had neither POS tags for the individual words nor for the whole units. Therefore, I tagged the remaining 257 entries with IceTagger to get the individual words' tags and similarly compared them with the tagged IFD sentence-per-line version as described above to ensure tagging consistency.

Next, the 257 idiom-MWEs were cross-referenced with the 124 MWE-phrases to facilitate assigning single-unit POS tags to each MWE so I could transform them into the Apertium XML style entries and finally add the TL translation to each entry before they were added to the bilingual dictionary.

4.5 Summary

The most time consuming task involved the dictionaries. The first round of data for the bidix consisted of approximately 5,000 entries and came from multiple sources. These entries had to be manually reviewed, firstly for correctness, and secondly to select a default translation since the lexical selection module did not exist in the beginning of this project. The development of the lexical selection modules was part of a parallel project

[4] Such as "*við hann og*" ('with him and') and "*við hlið hans*" ('next to him').

and modifications were added to some of the bilingual dictionary entries as part of that development phase.

The second round of data for the bidix consisted of approximately 6,000 words with one or more translations before processing them into the correct Apertium XML style format. After filtering, processing and manually reviewing the result was approximately 7,100 additional entries for the bidix. The third round of data for the bidix consisted of approximately 18,000 words with one or more translations before processing, and resulted in approximately 12,300 additional entries for the bidix.

The Apertium structural transfer module requires one or more stages of transfer rules depending on how closely related the SL and TL are. Closely related languages may only require one stage, but often there are three stages. The Apertium-IceNLP prototype has five stages. Transfer rules consist of patterns and actions, where certain actions are performed on the relevant patterns, usually reordering SL sentence chunks to conform to the linguistic rules of the TL.

Some of the existing Icelandic LT tools had to be slightly modified in order to be incorporated into the prototype. IceNLP was made open source and then transformed into a daemonized version so that it no longer needed to start up and load everything into memory each time an input string was sent to the system, but instead had a server running in the background waiting for client threads to deliver the input strings. Additional changes involved modifying IceNLP modules to handle input and output in the Apertium style format.

There were still differences between the way the pure Apertium system and the Apertium-IceNLP prototype handled MWEs. Basically, the pure system handled MWEs as single units while the prototype did not and the pure version could also handle MWEs with inflections while the prototype would have needed to hardcode that functionality for each instance. Changes were made so that the prototype could handle up to trigram MWEs, but the hardcoding solution to inflecting MWEs was considered impractical.

The next chapter discusses the results of initial evaluation, collection and processing of development data, the ensuing error analysis and the results of a subjective user survey.

Chapter 5

Evaluation

This chapter covers the evaluation set-up, how the evaluation data was acquired and processed, which methods were chosen to use for the evaluation and why. Next, statistics are presented from the evaluation of the prototype Apertium-IceNLP hybrid MT system and three other systems. Also presented here is a discussion of the collection and processing of development data, the resulting error analysis and user survey that was carried out based on said development data.

5.1 Evaluation set-up

The goal was to evaluate approximately 5,000 words, which corresponds to roughly 10 pages of text, and compare our results to two other publicly available Icelandic to English MT systems: Google Translate and Tungutorg.

5.1.1 Evaluation data

The evaluation needs a corpus of SL text which is run through the MT systems to be evaluated, producing the TL text which is then post-edited manually. The TL and post-edited files are then used as input for the evaluation method (see section 5.1.2). Francis Tyers procured a dump of the Icelandic Wikipedia on April 24th 2010, from which 187,906 lines of SL text were extracted for use as the evaluation data. The reason Wikipedia was used is because the IFD is only partially open-source, and therefore this evaluation could

not have been easily reproduced had the IFD been used.

I generated the TL translations by running the data through the Apertium-IceNLP prototype system and sequentially adding a unique number to each output line. Next, I randomly selected 1,000 lines from the test corpus by generating a file with numbers running from 1 to the maximum number of sentences (i.e. 187,906), scrambling the numerical order and selecting the first 1,000 numbers from the unordered file. Then I extracted the corresponding sentences with the matching numbers to a new evaluation data file. A quick examination of the resulting SL sample file revealed that there were some inconsistencies in the line format, e.g. more than one sentence per line, incomplete sentences, incoherent text or lists. Therefore, I deemed it necessary to prune the file so the evaluation would more accurately measure how the systems perform *dissemination* (see 2.2) and not how well they handle imperfect input.

First, I filtered out sentences with fewer than four words, as well as sentences with more than one lower-case unknown word[1], which left 509 sentences. The reason I left out sentences with more than one unknown word, was to test the performance of the rules of the system, not the coverage of the dictionaries. Then I filtered the remaining data manually, removing or modifying entries such that:

1. each line only had one sentence;
2. each line only had a complete sentence;
3. lines that were clearly metadata and traceable to individuals were removed, e.g. IP numbers or usernames;
4. lines that contained incoherent strings of numbers were removed, e.g. from a table entry;
5. lines containing non-Latin alphabet characters were removed, e.g. if they contained Greek or Arabic font;
6. lines that contained extremely domain specific and/or archaic words were removed, e.g. 'stúpan', 'pagóða', 'kjörfursti', 'ek' (as in 'út vil ek'); and
7. repetitive lines, e.g. multiple lines of the same format from a list, were removed.

[1] This did not filter out Icelandic proper nouns, as they all start with an upper-case letter.

After this filtering process, 397 sentences remained which I then ran through the three MT systems. In order to calculate the chosen evaluation metrics (see section 5.1.2), each of the three output files had to be post-edited. I reviewed each TL sentence, copied it and then made minimal corrections to the copied sentence so that it would be suitable for publication, see examples in table 5.1.

	SL sentence:	Einnig býr Finnland yfir fjölbreyttu og víðtæku dýralífi.
1)	TL sentence:	Finland also has the broad and diverse wildlife.
1)	Post-edited:	Finland also has a broad and diverse wildlife.
2)	TL sentence:	Moreover lives *Finnland over varied and widespread *dýralífi.[2]
2)	Post-edited:	Moreover Finland has a varied and widespread wildlife.
3)	TL sentence:	Also resides Finland over miscellaneous and an extensive animal life.
3)	Post-edited:	Also Finland has a miscellaneous and extensive animal life.

Table 5.1: *Example of an SL sentence, three TL versions and post-edited results.*

5.1.2 Selected evaluation methods

The translation quality was measured with the `apertium-eval-translator` tool[3], which uses WER and PER (see 2.5.2). Metrics based on word error rate were chosen so as to be able to compare the system against other systems based on similar technology, and to assess the usefulness of the system in a real setting, i.e. of translating for dissemination.

5.2 Evaluation results

Explanations for the statistics template in the next section:

- **Number of words in reference:** is the total number of words in the post-edited translation.
- **Number of words in test:** is the total number of words in the machine translated file.
- **Edit distance:** indicates how many substitutions, deletions and insertions were needed to get the machine translated sentence to match the post-edited one.

[3] http://sourceforge.net/projects/apertium/files/apertium-eval-translator/1.2/; Version 1.2.0, 4th November 2010.

- **Word error rate (WER):** is the percentage of the machine translated words that required correction.
- **Number of position-independent correct words:** is how many words were correct regardless of their position.
- **Position-independent word error rate (PER):** is the percentage of errors when the word order is disregarded.

To show how these statistics work, here is one SL sentence, its TL equivalent and the post-edited version, followed by the template frame with the appropriate statistics:

- **SL sentence:** Einnig býr Finnland yfir fjölbreyttu og víðtæku dýralífi.
- **TL sentence:** Also resides Finland over miscellaneous and an extensive animal life.
- **Post-edited:** Also Finland has a miscellaneous and extensive animal life.

- **Number of words in reference (post-edited):** 9 (the punctuation mark is not counted).
- **Number of words in test (TL sentence):** 10.
- **Edit distance:** is 4: 'resides' is deleted, 'over' is substituted for 'has', 'a' is inserted and 'an' is deleted.
- **Word error rate (WER):** $\frac{S+D+I}{N} = \frac{1+2+1}{9} = 0.444 = 44.4\%$.
- **Number of position-independent correct words:** 7, i.e. 'Also', 'Finland', 'miscellaneous', 'and', 'extensive', 'animal' and 'life'.
- **Position-independent word error rate (PER):** $1 - \left(\frac{C-\max(0,(T-N))}{N}\right) = 1 - \left(\frac{7-\max(0,(10-9))}{9}\right) = 1 - \left(\frac{7-1}{9}\right) = 1 - \left(\frac{6}{9}\right) = 1 - 0.667 = 0.333 = 33.3\%$.

Following are some statistics for the prototype, and for three other MT systems for comparison, taken in April 2010. A discussion of these statistics follow at the end of the section:

5.2.1 Apertium-IceNLP MT prototype

- Number of words in reference: 6374
- Number of words in test: 6042
- Edit distance: 3225
- Word error rate (WER): 50.60 %
- Number of position-independent correct words: 3775
- Position-independent word error rate (PER): 40.78 %

5.2.2 Pure Apertium MT

- Number of words in reference: 6353
- Number of words in test: 5867
- Edit distance: 2917
- Word error rate (WER): 45.92 %
- Number of position-independent correct words: 3927
- Position-independent word error rate (PER): 38.19 %

5.2.3 Tungutorg

- Number of words in reference: 6328
- Number of words in test: 6184
- Edit distance: 2811
- Word error rate (WER): 44.42 %
- Number of position-independent correct words: 4194
- Position-independent word error rate (PER): 33.72 %

5.2.4 Google Translate

- Number of words in reference: 6222
- Number of words in test: 5877
- Edit distance: 2271
- Word error rate (WER): 36.50 %
- Number of position-independent correct words: 4434
- Position-independent word error rate (PER): 28.74 %

Notice that with lower edit distances that the WER also goes down, and with higher numbers of position-independent correct words that the PER becomes lower. The scores are of course dependent upon the number of words in the references, but since the reference totals in these results are roughly in the same ball-park, the effect on the scores is more apparent. Table 5.2 brings together the WER and PER results from the above listings.

MT system	WER	PER
Apertium-IceNLP prototype	50.60 %	40.78 %
Pure Apertium	45.92 %	38.19 %
Tungutorg	44.42 %	33.72 %
Google Translate	36.50 %	28.74 %

Table 5.2: *Error rates for Icelandic-English MT systems.*

One possible explanation for the lower error rates for the pure Apertium version than the Apertium-IceNLP hybrid prototype is the handling of multiword expressions (MWEs). MWEs most often do not translate literally nor even to the same number of words, which can dramatically increase the error rate. The pure version translates unlimited lengths of MWEs as single units *and* can deal with MWEs that contain inflectional words, whereas the prototype could only handle unigrams (single words) at first, i.e. no MWE handling was present. Then, after some modifications were made so that the prototype could handle MWEs as single units, it can still only handle up to three-word expressions, and cannot deal with inflectional MWEs.

In order to bring the PER down to about 20 % for our prototype, it would require over 100 fewer post-edit corrections. The best way to do that is to try and identify a recurring pattern in the errors to minimize the amount of manual work needed. For example, the

indefinite article has not been fully dealt with, but it only needed to be inserted 14 times, i.e. less than 3 % of the errors. See more about error analysis in sections 5.3.2 and 5.5.

5.3 Development set-up

Although the first evaluation results showed that it was indeed possible to create an STMT system based on the Apertium framework with substituted modules, the results were rather disappointing. Therefore, efforts were made to analyze the current status of the system with the intent to identify major error categories in order to fix these categories without using the evaluation data.

5.3.1 Development data

I collected the development data from the largest Icelandic online newspaper '*mbl.is*'. The idea was to automate the process of retrieving the news articles and translating them with the prototype system. In order to do that I first needed to be able to read the RSS news feed and automatically save them in files.

There were some frustrating problems in the beginning, first to read Icelandic characters correctly and then to get the background scheduling (cron) to run the PHP program that extracted the RSS news feed. The problem seemed to be that I was attempting to '*jump through too many hoops at once*' so to speak, that is to say I wanted cron to start the PHP program which called a shell script that was a series of commands piped together using input from the PHP.

The solution entailed modifying the shell script to use full paths for everything and accepting an additional argument, then permanently setting various system environment variables inside cron.

Once I had verified that the automated process did in fact create what I wanted it to (an HTML file, a tagged SL file, a clean SL file and a TL file), I then scheduled it to run every 15 minutes of every day. Then I created more scripts and Perl programs to run weekly tests which produced a log of the evaluation statistics for the news articles during the

previous week.

5.3.2 Error analysis

After collecting development data for several weeks, I proceeded to analyze this new data. First I needed a random selection of files from the development data set. At the time of this selection, there were 1728 SL and TL files to select from. The random selection process used here was similar to that described in 5.1.1, with the exception that I created a single file with all of the 1728 filenames in it instead of individual sentences. In order to make the random selection of at least 50 files, I selected the first 100 numbers from the unordered list. As the pool of 1728 files contained both SL and TL files, there was some overlapping, just as I had anticipated. Thus, I skipped TL files if the matching SL file had already been selected, or, conversely, selected the matching SL file if only the TL file number was in the unordered list of 100, and finally I merged the corresponding HTML file, tagfile, SL file and TL file into one file for each of the remaining 82 files from the original unordered number list.

474	were proper nouns not found in the bidix: names of people, places or things
302	were compound words also not found in the bidix - nouns, verbs, adjectives
272	existed in the bidix - these require further analysis
138	were marked as generation errors
66	were compound proper nouns that were not found in the bidix, e.g. '*Reykjavíkurhöfn*' (Reykjavík harbor)
59	merely categorized as missing from the bidix, no further classification given
42	were due to multiword expressions with an inflecting verb
37	were due to standard multiword expressions
25	were combinations of numerals with punctuation marks, e.g. sports scores '*3:2*'
24	were abbreviations, either missing from the bidix or incorrectly classified
14	typing errors in the original SL news article
11	were split multiword expressions, usually with an inflecting verb as well
11	did not have an entry in the bidix for the appropriate word class
10	were proper nouns which should not have been translated
7	were proper noun single letter initials
5	were URLs
4	were marked as analysis errors
1	was an unresolved html tag that slipped through the PHP program
1502	**error-marked words in total**

Table 5.3: *List of all error-marked words in the development data.*

Next I manually reviewed each of the files and assigned an error category to each error (word marked with a symbol) in the file. I had to invent these error categories along the way and after completing the review of 50 files, the error categories were 18. The next step was to perform some data-mining on the error categories (see table 5.3).

Then I grouped the categories into meta-categories to identify where it would be most beneficial to make corrections so that the error rates might be lowered (see overview in table 5.4).

474 proper nouns;

66 compound proper nouns;

302 compound words - nouns, verbs, adjectives;

59 missing;

11 missing the appropriate word class;

912 Total words missing from the bilingual dictionary in some form or another.

272 existed in the bilingual dictionary;

138 were marked as generation errors;

4 were marked as analysis errors;

414 Total words that need further analysis as to why they were marked as errors.

37 standard multiword expressions;

42 multiword expressions with an inflecting verb;

11 split multiword expressions;

90 Total number of multiword expressions.[4]

[4] Note that the total number of MWEs is the number of expressions, not the total number of individual words. Each MWE contains one or more error-marked words and possibly one or more non-error-marked words as well.

24 abbreviations, either missing or incorrectly classified;

7 proper noun single letter initials;

31 Total number of abbreviations and initials.

25 various numerals;

5 URLs;

1 unresolved html tag;

31 Total number of errors that need more detailed pattern matching.

14 typing errors in the original SL news article;

10 were proper nouns which should not have been translated;

24 Total number of errors that cannot easily be dealt with.

	Number of errors	Percentage of all errors
Missing and compound words	912	60.7%
Need further analysis	414	**27.5%**
Multiword expressions	90	6.0%
Abbreviations and initials	31	2.1%
More sophisticated patterns	31	2.1%
Leftovers	24	1.6%
Total	1502	100%

Table 5.4: *Grouping of error categories into meta-categories.*

The reason I grouped compound words with the missing-words meta-category is because they were first and foremost missing words. However, since it is by far the largest error category, it makes sense to take a closer look at this particular meta-category (see table 5.5).

While I was handling the error categories from the development data, I noticed that some error-marked words seemed to appear more often than others, so I decided to examine the possibility that correcting a handful of error-marked words, regardless of their assigned error category, might dramatically lower the prototype's error rates if the occurrences of

	Number of errors	Percentage of all errors
Proper nouns	474	**31.6%**
Compound proper nouns	66	4.4%
Compound words	302	**20.1%**
Other missing words	70	4.6%
Subtotal	912	60.7%

Table 5.5: *More detailed classification of the missing-words meta-error-category.*

these words were high enough. Table 5.6 shows a list of the highest occurrences of the error-marked words, with a threshold set to five[5]

16	=genError=#to
9	=exists=*laxar
8	=genError=#much
8	=exists=*síðan
7	=missing=*sama
6	=exists=*um
6	=exists=*sér
6	=exists=*milljarðar
5	=sérnafn=*Örn
5	=genError=#why
5	=genError=most_#first

Table 5.6: *Error-marked words above threshold set at five, surface forms only.*

The number at the beginning of the line is the number of times that particular word, in this particular surface form, occurred marked as an error in the development data. The error category is pre-fixed to the error-marked word between two equal signs (=), and the asterisk (*) and hash (#) symbols are error symbols output from the prototype which are pre-fixed to the problematic word or words with an underscore (_) replacing blank spaces.

The specified surface forms of the eleven words in table 5.6 occurred eighty-one times in the development data. Some of these words also occurred as other surface forms and were therefore automatically counted separately, so I assumed that if all forms of the words' lemmas were counted collectively, it could affect whether a word rose above the threshold or not. For example, '*næsta*' and '*næstu*' (TL surface forms of the SL word '*next*') had four and three occurrences respectively and were not listed when the threshold was set to five unless they were counted together (see table 5.7).

[5] If the threshold is set to five, that means that there must be at least five occurrences of the error for it to be noteworthy.

16	=genError=#to
12	=genError=#much + #Much
11	=exists=*laxa + *laxar + *laxar.
10	=exists=*sig + *sér + *sín
10	=exists=*síðan + *síðan, + *síðan.
9	=exists=*annað + *annan + *annar + *Annar + *annarra + *öðrum + *önnur
8	=missing=*sama + *sömu
7	=exists=*næsta + *næstu
6	=sérnafn=*Ólafsfirði + *Ólafsfirði. + *Ólafsfjarðar
6	=sérnafn=*Fulham + *Fulham, + *Fulham:
6	=sérnafnComp=*Þjórsárver, + *Þjórsárvera" + *Þjórsárverum + *Þjórsárverum.
6	=missingComp=*viðræðna + *viðræður + *viðræður
6	=genError=#why + #Why
6	=genError=most_#first + Most_#first
6	=exists=*um
6	=exists=*milljarðar
5	=sérnafn=*Örn
5	=sérnafn=*Arion_knock + *Arion_knock, + *Arion_knocks
5	=missing=*tekjum + *tekjur
5	=missingComp=*greiðslustöðvun + *greiðslustöðvunar + *Greiðslustöðvunin

Table 5.7: *Error-marked words above threshold set at five, all word forms.*

I also found that when all forms of a word were considered one error, that roughly twice as many error instances emerged than when considering each surface form uniquely, given the same threshold (see tables 5.8 and 5.9). Therefore, setting the threshold to five and considering all forms of the error-marked words from their lemma, will account for 9.0% of all error-marked words (see table 5.8) but this percentage rises to 15.4% if the threshold is lowered to four (see table 5.9).

	Threshold >= 5		
	Unique instances	Error-marked words	Percentage of all errors
Surface form	11	81	5.4%
Lemma form	20	136	9.0%

Table 5.8: *Error instances per surface form and lemma with threshold set to five.*

I will speculate further on how this analysis may be of use to improve the prototype in section 5.5.

		Threshold >= 4	
	Unique instances	Error-marked words	Percentage of all errors
Surface form	21	121	8.0%
Lemma form	40	231	**15.4%**

Table 5.9: *Error instances per surface form and lemma with threshold set to four.*

5.4 User survey

I had the opportunity to informally introduce this research to the public on European Researcher's Night 2010, at the event's location in Reykjavík, so I thought I might be able to '*bring down two birds with one stone*'. On the one hand, I hoped to get people to actively participate by filling out a survey regarding the results of a small selection from the evaluation data, and on the other hand, by saving the survey answers I could get a subjective assessment of that set.

However, since there were not many that participated at the event (probably due to an overwhelming overload of the senses as the place was packed with people and activities) I followed up by sending invitations via email, and that was far more successful.

5.4.1 Survey set-up

The survey was based on the evaluation data, which consisted of 397 sentences (see section 5.1.1). I decided to request the users to rank three MT systems w.r.t. each other, based on how well they deemed the TL sentence corresponded to the SL sentence: the Apertium-IceNLP hybrid prototype, Google translate and Tungutorg.

I used the same process for selecting random sentences as described in 5.1.1 to select 40 random sentences from the 397. After consulting my supervisor, I decided not to use the pure-Apertium version since it is rather a by-product than a full-fledged MT system.

Since the survey was to be presented after an introduction to the Apertium-IceNLP web interface, which displays symbols in front of unknown words and other errors, I removed all symbols from the prototype's TL sentences before adding them to the survey. Otherwise the survey could no longer be considered "anonymous", i.e. the prototype's output

would be easily recognized since the other MT systems don't mark words with symbols.

The user was asked to rank at least 10 sentences with a maximum of 40 sentences available, but could quit at any point in the process. Each question consisted of the original SL sentence and the three TL results presented in random order, generated automatically by the survey program for each individual user. The online survey was sent to almost 3000 university students and over 500 replies for the first 10 sentences were received.

5.4.2 Survey results

The purpose of the survey was really only to establish a starting point for future references after improvements have been made to the Apertium-IceNLP system, so I was not surprised when the prototype received the worst subjective ranking, because that was what I expected.

Google translate received the highest ranking, Tungutorg came in second and the Apertium-IceNLP prototype came in third and last place. Of the ten questions, five had a very clear internal ranking, three were fairly clear-cut and two questions were very close. Those two questions placed Google clearly in 3rd place (the only times it received 3rd rank), but neither Apertium-IceNLP nor Tungutorg received over 50% for the other two places in these cases.

I looked at the overall ranking, the ranking results by gender, and by three age groups. The only deviations I found within each of these five groups (men, women, age groups 16-20 yrs, 21-40 yrs and 41-60 yrs old) from the overall ranking were 4 pairings out of the 50 (five groups of the ten questions). These occurred in two separate questions, the former had the age groups 16-20 and 41-60 place Apertium-IceNLP in 1st place in one of the close call questions; and the second occurrence was a very short sentence where the women and youngest age group disagreed with the overall ranking of Apertium-IceNLP in 1st place and Google in 2nd place and had those two reversed.

Here is the output from the survey software for each question. Note that the first two questions were regarding gender and age:

Question 3

Upprunalega setningin á íslensku er svohljóðandi:

"Einnig býr Finnland yfir fjölbreyttu og víðtæku dýralífi."

	1	2	3	Responses	Total
Moreover Finland lives over varied and widespread dýralífi.	5%	20%	75%	327	33%
Finland also has the broad and diverse wildlife.	84%	10%	5%	334	34%
Also resides Finland over miscellaneous and an extensive animal life.	11%	69%	20%	324	33%

Figure 5.1: *Question 3 of the user survey.*

Question 4

Upprunalega setningin á íslensku er svohljóðandi:

"Skammt innan við þorpið er Álfaborgin, sem samkvæmt þjóðsögum er bústaður huldufólks."

	1	2	3	Responses	Total
Short within with the village is Álfaborgin, that according to folktales residence of elves is.	45%	36%	19%	330	34%
Dose within the village is alpha city, which according to folklore, the residence of hidden people.	10%	24%	66%	323	33%
A portion within at the village is the Álfaborg, which according to legends is a residence of fairies.	46%	39%	16%	327	33%

Figure 5.2: *Question 4 of the user survey.*

Question 5

Upprunalega setningin á íslensku er svohljóðandi:

"Það hefur verið opinbert tungumál Færeyja frá 1937."

	1	2	3	Responses	Total
It has been opinbert language Færeyja from 1937.	4%	1%	95%	326	33%
It has been the official language of the Faroe Islands since 1937.	94%	2%	4%	332	34%
It has been public language of Faeroe Islands from 1937.	2%	97%	1%	321	33%

Figure 5.3: *Question 5 of the user survey.*

Question 6

Upprunalega setningin á íslensku er svohljóðandi:

"Þar bíður hún eftir bráð sinni."

	1	2	3	Responses	Total
There she waits after her victim.	51%	44%	5%	329	33%
As she waits for prey.	46%	45%	9%	328	33%
There waits she of a prey their.	4%	9%	87%	326	33%

Figure 5.4: *Question 6 of the user survey.*

Question 7

Upprunalega setningin á íslensku er svohljóðandi:

"Angurboða var gýgi í Jötunheimi í norrænni goðafræði."

	1	2	3	Responses	Total
Angurboða Was gýgi in Jötunheimi in Nordic mythology.	43%	46%	12%	320	34%
Angus Such a meeting was in Jötunheim giantess in Norse mythology.	9%	10%	81%	314	33%
Angurboður was a gýg in Jötunheimur in a Nordic mythology.	49%	43%	8%	320	34%

Figure 5.5: *Question 7 of the user survey.*

Question 8

Upprunalega setningin á íslensku er svohljóðandi:

"Eftir það ævintýri var hann svo keyptur til Milan liðsins."

	1	2	3	Responses	Total
After the one adventure he was then bought to Milan the team.	11%	82%	8%	319	33%
After that adventure, he was purchased for Milan team.	85%	10%	5%	324	34%
Thereafter an adventure was he so bought to the Milan of the team.	5%	7%	88%	315	33%

Figure 5.6: *Question 8 of the user survey.*

Question 9

Upprunalega setningin á íslensku er svohljóðandi:

"Svo virðist sem hann hafi fallist á allar kenningar stóuspekinnar fyrir utan að hann hafnaði því að heimurinn myndi farast í eldi."

	1	2	3	Responses	Total
then Respects as he has fallen on all the theories of the stoicism fyrir without that he rejected it that the world will go in fire.	4%	8%	88%	300	33%
It seems that he has agreed to all theories of Stoicism sciences except that he rejected the world would perish in the fire.	91%	6%	4%	307	34%
So seems which he has approved the all theories of the Stoicism beside to he refused it that the world would burn.	6%	86%	8%	299	33%

Figure 5.7: *Question 9 of the user survey.*

Question 10

Upprunalega setningin á íslensku er svohljóðandi:

"Tölvan notast við WiiConnect 24, sem leyfir notendum að ná í uppfærslur og að taka á móti og senda skilaboð í gegnum netið, og notar WiiConnect 24 afar lítið rafmagn."

	1	2	3	Responses	Total
The computer use with WiiConnect 24, that allows notendum to obtain in update and to take on meeting and send messages í through the net, and uses WiiConnect 24 grandfathers of small electricity.	5%	38%	57%	297	33%
Your computer uses WiiConnect 24, which allows users to download updates and to receive and send messages through the network, and uses 24 WiiConnect very little electricity.	89%	4%	7%	304	34%
The computer done with WiiConnect 24, which allows users to catch presentations and to receive and send messages per the network, and uses WiiConnect 24 grandfathers a little electricity.	5%	58%	37%	293	33%

Figure 5.8: *Question 10 of the user survey.*

Question 11

Upprunalega setningin á íslensku er svohljóðandi:

"Alexa Vega fæddist í Miami á Flórída."

	1	2	3	Responses	Total
Alexa Roads fed in Miami on Flórída.	4%	2%	94%	296	33%
Alexa Vega was born in Miami in Florida.	94%	2%	4%	305	34%
Alexes Vega was born in Miami on Flórída.	2%	96%	2%	296	33%

Figure 5.9: *Question 11 of the user survey.*

Question 12

Upprunalega setningin á íslensku er svohljóðandi:

"Haukadalur getur átt við eftirfarandi:"

	1	2	3	Responses	Total
Haukadalur Can own with following :	3%	24%	73%	300	33%
Haukadalur may refer to:	93%	3%	4%	307	34%
Haukadalur can alluded following:	4%	73%	23%	299	33%

Figure 5.10: *Question 12 of the user survey.*

5.5 Discussion

When comparing the prototype to other MT systems that use the Apertium platform, I noticed that the range of word error rate percentages is very wide, from 17% to 72% (Apertium, 2010c):

Language pair	WER	PER
Norwegian bokmål - Norwegian nynorsk	17.7 %	N/A
Swedish-Danish	30.3 %	27.7 %
Basque-Spanish	72.4 %	39.8 %
Welsh-English	55.7 %	30.5 %

Table 5.10: *Error rates of some other MT systems using the Apertium platform.*

The results in table 5.10 seem to indicate that the error rates increase in proportion with increased relational distance between the observed language pairs. For example, Swedish and Danish are both Mainland Scandinavian languages, which are closely related (see the dashed lines in figure 5.11), and the Norwegian languages are even closer (see the dashed-and-dotted lines in figure 5.11), while Welsh and English are distantly related (see the unbroken lines in figure 5.12). Note that the North Germanic language branch is identified with a box in both figures to show the relationship between the two.

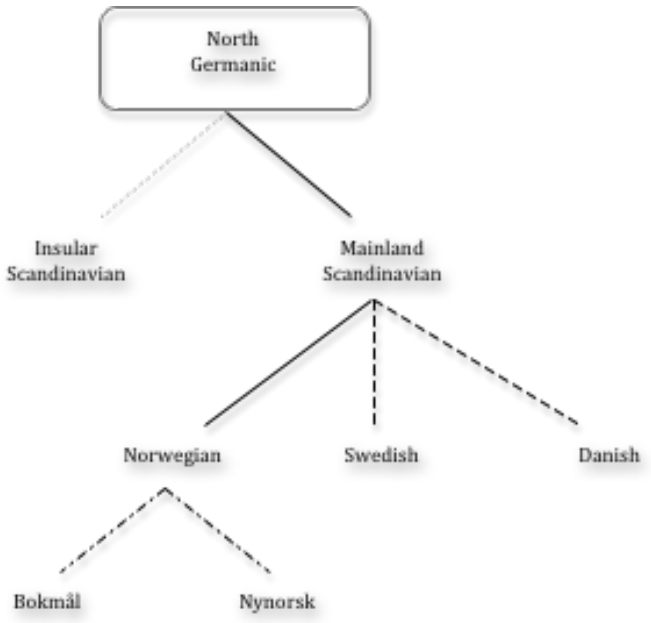

Figure 5.11: *The language family tree relationship between Norwegian bokmål, nynorsk, Swedish and Danish.*

Based on language family and branch distances between Icelandic and English (see the dotted lines in figure 5.12), I would expect an Apertium-based Icelandic-English MT system to have an error rate somewhere between the Apertium-based Swedish-Danish and Welsh-English error rates, i.e. WER between 30.3-55.7 % and PER between 27.7-30.5 %. The current WER for the Apertium-IceNLP hybrid prototype is within that range at 50.60 % but its PER is considerably higher, at 40.78 %.

The comparisions of the prototype to other MT systems shown in table 5.2 and the subjective user survey indications in section 5.4.2, suggest that this prototype is just barely comparable to other currently available MT systems and needs considerable work done to become competitive.

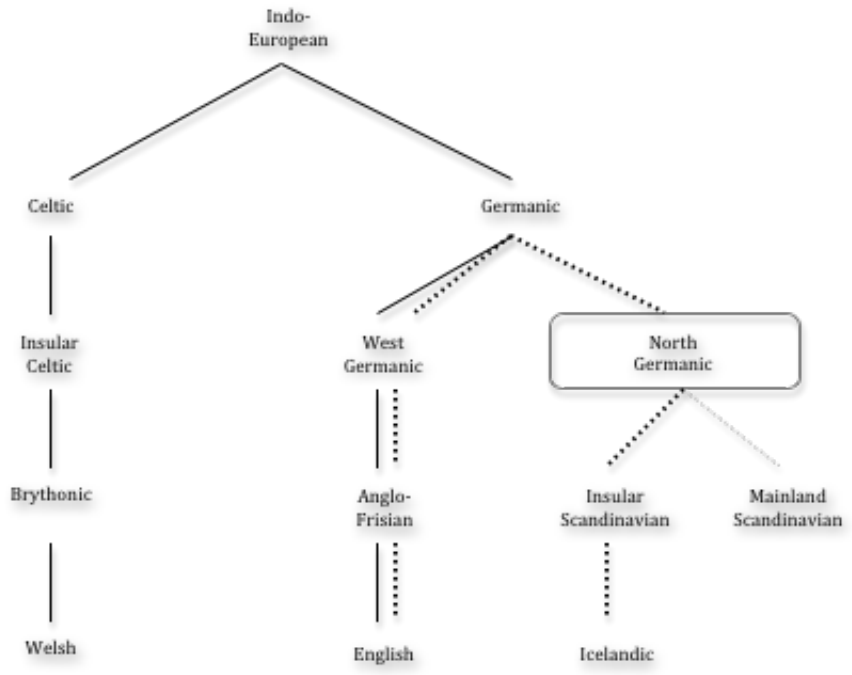

Figure 5.12: *The language family tree relationship between Welsh, English and Icelandic.*

The poor performance of the prototype on the evaluation data can mainly be attributed to the fact that the Icelandic state-of-the-art NLP tools did not improve translation quality as was hoped. This may be largely attributed to the incompatibility of the IceNLP modules with the Apertium modules, which required that the IceNLP modules be adapted to work in the Apertium pipeline.

Another large contributor to the poor performance may be because IceTagger can only handle a fraction of MWEs, i.e. only if the MWE does not contain a verb. Currently, the only way the prototype can handle MWEs with verbs is to hardcode them into the system, i.e. to add an entry for each MWE with every inflected form of the verb (or combination of verbs) to the tagmapping file and this is just not feasible. This may in particular attribute to the difference in the error rates between the Apertium-IceNLP hybrid prototype and the pure Apertium version, since the prototype only recognizes 98 MWEs of the 560 MWEs in the bidix, while the pure version can utilize all of them. If we look at the number of errors attributed to the MWE meta-category in table 5.4, we see that they account for 6.0% of all errors, and in table 5.2 we see that the difference between the WER scores for the prototype and pure version is 4.68%. One way to test this theory would be to make sure there are no MWEs in the data being evaluated. This could be done for example by splitting the current evaluation data set into sentences with and without MWEs and run these two subsets again through the prototype and pure Apertium version. If the difference in error rates between the prototype and pure Apertium for each subset don't show significantly different results, then it is unlikely that MWEs are the cause of the performance difference between these two systems.

On the other hand, the analysis of error categories on the development data (see section 5.3.2) indicate that improvement to the performance of the prototype can be achieved by concentrating on adding proper nouns to the bidix on the one hand and resolving compound words on the other. These errors consisted of over half of the error-marked words in the development data, i.e. proper nouns were 31.6% of all errors, compound words were 20.1% and compound proper nouns were 4.4%, coming to a total of 56.1% of all error-marked words in the development data set.

The proper noun errors could be reduced by translating gazetteers, i.e. lists of place names, but person names would still have to be added on a case-by-case basis. Since the error categorization did not differentiate between types of proper nouns, it cannot be estimated here how much that action would affect the ratio of errors.

The Icelandic language is rich with compound words, therefore it is foreseeable that the ability to decompose compound words into smaller words will be extremely useful, whether the functionality were to be added to one of the current IceNLP modules or as a new module. As was mentioned in section 3.4, the Apertium-IceNLP hybrid prototype does not currently utilize all of the IceNLP modules, and so it is also foreseeable

that integrating IceParser into the prototype could be beneficial, since it labels constituent structure and syntactic functions of the input text, which may be quite useful when writing transfer rules.

Improvement of performance could also be achieved by correcting all surface forms of the error instances above threshold four which accounted for 15.4% of all error-marked words in the development data set (see table 5.9). However, it has not been established which error categories these individual errors come from and the correction of these errors may prove arbitrary instead of contributing to the reduction of certain error categories. Furthermore, it will be necessary to perform additional analysis in order to determine the cause of 27.5% of the errors (see table 5.4).

Once some of these improvements have been implemented, it would be interesting to evaluate these same three Icelandic-English MT systems again with the same evaluation data set and compare the results to see if the Apertium-IceNLP hybrid translation system will indeed be a feasible option. However, the additional work required to get a better performance out of the Apertium-IceNLP hybrid system than a pure Apertium system raises the question as to whether "*less is more*", i.e. whether instead of trying to incorporate as much of the existing Icelandic NLP tools as possible into the Apertium-based hybrid, that it may produce better results to use only IceTagger for POS tagging, since that outperforms the Apertium tagger for Icelandic. In order to step back and only replace the POS tagger in the Apertium pipeline, some modifications will have to be made to IceTagger. In addition to the modifications that were already done to make IceTagger return output in Apertium style format, the tagger will also have to be able to take Apertium style formatted input. It will also be necessary to separate IceMorphy and Lemmald from IceTagger more clearly, so that instead of calling for them specifically from within IceTagger, it can receive the necessary information from Apertium sources.

5.6 Summary

Since the Icelandic Frequency Dictionary is only partially open-source, the Icelandic Wikipedia website was used as the corpus from which roughly ten pages of text, approximately 5,000 words, was selected in order to evaluate the performance of the prototype. The Wikipedia corpus consisted of 187,906 lines of SL text. In order to extract

a suitable test set 1,000 lines were randomly selected, from which 397 lines remained after automatic and manual filtering. This test set of 397 SL sentences was processed by the Apertium-IceNLP prototype and the resulting TL file was then post-edited into a third file. The TL and post-edited files were then used to calculate the prototype's WER and PER, which were 50.60 % and 40.78 % respectively. The WER and PER evaluation methods were chosen because they could assess the usefulness of the system in a real setting and they had been used to evaluate other Apertium systems. The same test set was used to evaluate two other publicly available Icelandic-English MT systems for comparison: Google translate and Tungutorg. Although the pure Apertium Icelandic-English MT system is a by-product of the prototype, it is also a fully functional MT system and was therefore tested as well. The pure Apertium system's WER and PER were lower than the prototype's, Tungutorg scored lower than the pure Apertium system and Google translate had the lowest error rates of these publicly available Icelandic-English MT systems.

A user survey was conducted based on a small random selection from the test set of 397 sentences to discern a subjective view of the translation quality of the three MT systems; Google translate, Tungutorg and the prototype. The survey was presented to almost 3,000 university students and over 500 replies were received. Participants were asked to blindly rank the translations of three MT systems internally. As was expected, the Apertium-IceNLP prototype received the worst subjective ranking. Furthermore, there was a unanimous consensus on almost all responses regardless of gender or age.

Even though the evaluation results were not as good as was hoped, they showed that it was indeed possible to create a shallow-transfer MT system based on the Apertium platform with substituted modules. So, in order to determine where to concentrate efforts towards improving the performance of the prototype some error analysis was carried out. In order to prevent influencing potential future evaluation results of the test set, a development data set was created for this purpose. This development data was collected from the largest Icelandic online newspaper into SL files, translated by the prototype into TL files and then 50 files from the pool of 1728 SL and TL files were randomly selected for manual review and categorization of errors in those 50 files. Analysis of the error categories showed that 60.7 % of the errors were due to words missing from the bidix, mostly proper nouns and compound words. The possibility that a high percentage of errors might be traceable back to a small set of words was explored using thresholds.

The WER and PER of a few other Apertium MT systems indicated that error rates increase in proportion with increased relational distance between the observed language pairs. Based on this observation and the language tree distance between Icelandic and English it could be expected that the WER for this language pair should be somewhere between 30.3 % and 55.7 % and that the PER should be somewhere between 27.7 % and 30.5 %. The actual WER of the prototype falls within that range, but the PER is considerably higher. This suggests that efforts towards improvement should be concentrated correcting sentence structure, possibly by adding more transfer rules.

Other conjectures regarding reasons for the prototype's poor performance are that the IceNLP modules required modification to be incorporated into the Apertium platform and poor handling of MWEs. Suggestions for addressing improving performance include translating gazetteers; decomposing compound words; integrating IceParser for constituent structure and syntactic functions; and performing further analysis on 27.5 % of the identified errors in the development data. Almost all work towards improving the prototype will also improve the pure Apertium by-product. It may be worth considering whether the required additional work to integrate all these IceNLP modules into the prototype is worth the effort, i.e. whether it might be better for the performance results to take a step back and perhaps just use IceTagger.

The final chapter of this thesis discusses conclusions and future work regarding this project.

Chapter 6

Conclusions and future work

This thesis has described how a prototype shallow transfer machine translation system for Icelandic to English was created. The intent was that the prototype would help advance the Icelandic BLARK, which currently does not have an MT system. Furthermore, this prototype translates from a minor inflectional language to a (non-inflectional) major language, and one hope was that this work might prove beneficial for other (inflectional) languages.

The prototype was created by integrating existing open source LT tools from the IceNLP toolkit into the Apertium machine translation platform. The hybrid Apertium-IceNLP prototype MT system can be accessed here: http://nlp.cs.ru.is/ApertiumISENWeb/.

This prototype is the first system which replaces the whole morphological and tagging modules of the Apertium machine translation platform with modules from an external system. The hope was that by using state-of-the-art Icelandic NLP modules that the translation quality would be better.

However, the results of evaluations of the prototype are not disastrous even though they are disappointing. The prototype comes in third place of the three Icelandic-English MT systems that were compared, both with statistical methods and in a subjective user survey. More work must be put into the system to deal with multiwords, compound words, lexical selection and adding more transfer rules to cover more patterns, before it will become apparent whether this kind of hybridization is feasible.

Bibliography

ALPAC. (1966). *Language and Machines: Computers in Translation and Linguistics. A report by the Automatic Language Processing Advisory Committee* (Tech. Rep. No. Publication 1416). 2101 Constitution Avenue, Washington D.C., 20418 USA: National Academy of Sciences, National Research Council.

Apertium. (2010a). *Icelandic-English language pair.* http://wiki.apertium.org/wiki/Icelandic_and_English.

Apertium. (2010b). *Language and pair maintainer.* http://wiki.apertium.org/wiki/Language_and_pair_maintainer.

Apertium. (2010c). *Translation Quality Statistics.* http://wiki.apertium.org/wiki/Translation_quality_statistics.

Armentano-Oller, C., Carrasco, R. C., Corbí-Bellot, A. M., Forcada, M. L., Ginestí-Rosell, M., Ortiz-Rojas, S., et al. (2006). Open-source Portuguese-Spanish machine translation. In R. Vieira, P. Quaresma, M. Nunes, N. Mamede, D. Oliveira, & M. Dias (Eds.), *Computational Processing of the Portuguese Language, Proc. PROPOR 2006* (Vol. 3960, p. 50-59). Springer-Verlag Berlin Heidelberg.

Armentano-Oller, C., Corbí-Bellot, A. M., Forcada, M. L., Ginestí-Rosell, M., Bonev, B., Ortiz-Rojas, S., et al. (2005, September 16). An open-source shallow-transfer machine translation toolbox: consequences of its release and availability. In *Proceedings of Workshop on Open Source Machine Translation* (p. 23-30). Phuket, Thailand: MT Summit X.

Arnold, D. (2003). Why Translation is Difficult for Computers. In *Computers and Translation: A translator's guide* (First ed., Vol. xvi, p. 119-142). Amsterdam: John Benjamins Publishing Company.

Bjarnadóttir, K. (2004). Beygingarlýsing íslensks nútímamáls (Morphological Description of Modern Icelandic). In *Samspil tungu og tækni* (p. 23-25). Reykjavik, Iceland: Ministry of Education, Science and Culture.

Bjarnadóttir, K. (2005). Modern Icelandic Inflections. In H. Holmboe (Ed.), *Nordisk sprogteknologi 2005 - Nordic Language Technology* (Vol. 5, p. 49-50). Copenhagen,

Denmark: Museum Tusculanum Press.

Brants, T. (2000). TnT: A Statistical Part-of-Speech Tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing* (p. 224 - 231). Seattle, WA, USA: Association for Computational Linguistics.

Briem, S. (1990). Automatisk morfologisk analyse af islandsk tekst (Automatic morphological analysis of Icelandic text). In J. Pind & E. Rögnvaldsson (Eds.), *Papers from the Seventh Scandinavian Conference of Computational Linguistics Reykjavik 1989* (p. 3-13). Reykjavik, Iceland: Institute of Lexocography, Institute of Linguistics.

Briem, S. (2009, May 16th). *Personal communication.* E-mail.

Briem, S. (2010). *Tungutorg.* http://www.tungutorg.is/.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Jr., E. R. H., & Mitchell, T. M. (2010, October). Toward an Architecture for Never-Ending Language Learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010).* Association for the Advancement of Artificial Intelligence (AAAI).

Didriksen, T. (2010). *Visual Interactive Syntax Learning, VISL CG-3.* http://beta.visl.sdu.dk/visl/about/. University of Southern Denmark.

Google. (2010). *19.04.2010.* http://translate.google.com/.

Helgadóttir, S. (2004). Mörkuð íslensk málheild (A Tagged Icelandic Corpus). In *Samspil tungu og tækni* (p. 67-71). Reykjavik, Iceland: Ministry of Education, Science and Culture.

Helgadóttir, S. (2005). Testing Data-Driven Learning Algorithms fro PoS Tagging of Icelandic. In H. Holmboe (Ed.), *Nordisk sprogteknologi 2004 - Nordic Language Technology* (p. 257-265). Copenhagen, Denmark: Museum Tusculanum Press.

Helgadóttir, S. (2007). Mörkun íslensks texta (Tagging Icelandic Text). In *Orð og tunga* (Vol. 9, p. 75-107). Reykjavik, Iceland: Orðabók háskólans.

Hutchins, J. (1993). The first MT patents. In J. Hutchins (Ed.), *MT News International* (p. 14-15).

Hutchins, J. (2005a). *The history of machine translation in a nutshell.* http://www.hutchinsweb.me.uk/Nutshell-2005.pdf.

Hutchins, J. (2005b). Towards a definition of example-based machine translation. In *Proceedings of Workshop on Example-Based Machine Translation* (p. 63-70). Phuket, Thailand.

Hutchins, W. J., & Lovtskii, E. (2000). Petr Petrovich Troyanskii (1854-1950): A forgotten pioneer of mechanical translation. In *Machine translation* (Vols. 15, no.3, p. 187-221).

Hutchins, W. J., & Somers, H. L. (1992). *An Introduction to Machine Translation*. London, UK: Academic Press.

IBM. (1954). *701 translator.* IBM Archives online: Press release January 8th 1954. http://www-03.ibm.com/ibm/history/exhibits/701/701-translator.html.

Ingason, A., Helgadóttir, S., Loftsson, H., & Rögnvaldsson, E. (2008). A Mixed Method Lemmatization Algorithm Using Hierachy of Linguistic Identities (HOLI). In B. Nordström & A. Rante (Eds.), *Advances in Natural Language Processing* (Vol. 5221/2008, p. 205-216). Gothenburg, Sweden: Springer-Verlag Berlin Heidelberg.

Ingason, A. K., Jóhannsson, S. B., Rögnvaldsson, E., Loftsson, H., & Helgadóttir, S. (2009). Context-Sensitive Spelling Correction and Rich Morphology. In K. Jokinen & E. Bick (Eds.), *Proceedings of the 17th Nordic Conference of Computational Linguistics NODALIDA* (p. 231-234). Tartu, Estonia: Northern European Association for Language Technology (NEALT).

Kramarczyk, I. (2009). *Improving the tagging accuracy of Icelandic text*. Unpublished master's thesis, Reykjavík University.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet Physics Doklady* (Vol. 10, p. 707-710).

Loftsson, H. (2006). Tagging Icelandic text: an experiment with integrations and combinations of taggers. In *Language Resources and Evaluation* (Vol. 40, p. 175-181). Springer Science+Business Media B.V.

Loftsson, H. (2008). Tagging Icelandic text: A linguistic rule-based approach. In *Nordic Journal of Linguistics* (Vol. 31(1), p. 47-72).

Loftsson, H., Kramarczyk, I., Helgadóttir, S., & Rögnvaldsson, E. (2009). Improving the PoS tagging accuracy of Icelandic text. In K. Jokinen & E. Bick (Eds.), *Proceedings of the 17th Nordic Conference of Computational Linguistics NODALIDA* (Vol. 4, p. 103-110). Northern European Association for Language Technology (NEALT).

Loftsson, H., & Rögnvaldsson, E. (2007a). IceNLP: A Natural Language Processing Toolkit for Icelandic. In *Proceedings of InterSpeech 2007.* Antwerp, Belgium.

Loftsson, H., & Rögnvaldsson, E. (2007b). IceParser: An Incremental Finite-State Parser for Icelandic. In J. Nivre, H.-J. Kaalep, K. Muischnek, & M. Koit (Eds.), *Proceedings of the 16th Nordic Conference of Computational Linguistics* (p. 128-135). Tartu, Estonia: University of Tartu.

mbl.is. (2010). *Íslenska í þýðingarvél Google (Icelandic in Google Translate).* Online news article, April 17th 2010: http://mbl.is/mm/frettir/taekni/2009/08/29/islenska_i_thydingarvel_google/.

Ólafsson, R., Rögnvaldsson, E., & Sigurðsson Þorgeir. (1999, April). *Tungutækni: Skýrsla starfshóps (Language Technology: a workgroup's report)* (Skýrslur og áitsgerðir 9 No. ISBN 9979-882-22-0). Reykjavik, Iceland: Menntamálaráðuneytið (The Icelandic Ministry of Education).

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002, July). BLEU: a Method for Automatic Evaluation of Machine Translations. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)* (p. 311-318). Philadelphia, PA, USA.

Pind, J., Magnússon, F., & Briem, S. (1991). *Íslensk orðtíðnibók (Icelandic Frequency Dictionary)*. The Institute of Lexicography, University of Iceland.

Rögnvaldsson, E. (2004). The Icelandic Speech Recognition Project Hjal. In H. Holmboe (Ed.), *Nordisk sprogteknologi 2003 - Nordic Language Technology* (p. 239-242). Copenhagen, Denmark: Museum Tusculanum Press.

Rögnvaldsson, E., Kristinsson, B., & Þorsteinsson, S. (2006, January 20th). Nýr íslenskur þulur að koma á markað (A New Icelandic Text-to-Speech System). *Morgunblaðið*.

Rögnvaldsson, E., Loftsson, H., Bjarnadóttir, K., Helgadóttir, S., Whelpton, M., Nikulásdóttir, A. B., et al. (2009). Icelandic Language Resources and Technology: Status and Prospects. In R. Domeij, K. Koskenniemi, S. Krauwer, B. Maegaard, & E. R. nad Koenraad de Smedt (Eds.), *Proceedings of NODALIDA 2009 workshop Nordic Perspectives on the CLARIN Infrastructure of Language Resources* (Vol. 5, p. 27-32). Tartu, Estonia: Northern European Association for Language Technology (NEALT).

Santorini, B. (1995). *Part-of-Speech Tagging Guidelines for the Penn Treebank Project* (3rd Revision, 2nd Printing ed.; Technical Report (CIS) Nos. MS-CIS-90-47, LINC LAB 178). Philadelphia, PA, USA: Department of Computer and Information Science, University of Pennsylvania.

Sigurþórsson, H. (2010). *Daemonizing IceNLP for the purpose of Machine Translation.* Independent study, Reykjavik University.

Skúlason, F. (2004). Endurbætt tillögugerðar- og orðskiptiforrit Púka (Improved Suggestions and Hyphenations in the Púki Spell Checker). In *Samspil tungu og tækni* (p. 29-31). Reykjavik, Iceland: Ministry of Education, Science and Culture.

TranslationExperts. (2010a). *NeuroTran Grammar Translation and Dictionary.* http://www.tranexp.com/win/NeuroTra.htm.

TranslationExperts. (2010b, April 9th). *Personal communication.* E-mail.

Waage, H. (2004). Hjal - gerð íslensks stakorðagreinis (The Making of an Icelandic Isolated Word Recognizer). In *Samspil tungu og tækni* (p. 47-53). Reykjavik, Iceland: Ministry of Education, Science and Culture.

Appendix A

Glossary

ALPAC	Automated Language Processing Advisory Committee
bidix	bilingual dictionary
BLARK	Basic LAnguage Resource Kit
EBMT	Example-Based Machine Translation
DTD	Document Type Definitions
HMM	Hidden Markov Model
HTML	Hyper Text Markup Language
ICLT	Icelandic Centre for Language Technology
LT	Language Technology
monodix	monolingual dictionary
MT	Machine Translation
MWE	Multi-Word Expression
NLP	Natural Language Processing
OS	Operating System
PC	Personal Computer
PER	Position-independent word Error Rate
POS	Part-Of-Speech
RBMT	Rule-Based Machine Translation
RTF	Rich Text Format
SL	Source Language
SMT	Statistical Machine Translation
STMT	Shallow-Transfer Machine Translation
TL	Target Language
WER	Word Error Rate
XML	eXtensible Markup Language

School of Computer Science
Reykjavík University
Menntavegi 1
101 Reykjavík, Iceland
Tel. +354 599 6200
Fax +354 599 6201
www.reykjavikuniversity.is
ISSN 1670-8539