NAIST-IS-DD0561041

Doctoral Dissertation

Applying Deep Grammars to Machine Translation, Paraphrasing, and Ontology Construction

Eric Nichols

March 22, 2010

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Eric Nichols

Thesis Committee:

Professor Yuji Matsumoto (Supervisor)
Professor Hiroyuki Seki (Co-supervisor)
Associate Professor Kentaro Inui (Co-supervisor)
Associate Professor Francis Bond (Co-supervisor)

Applying Deep Grammars to Machine Translation, Paraphrasing, and Ontology Construction*

Eric Nichols

Abstract

Many challenging tasks in the field of Natural Language Processing, such as machine translation, linguistic resource construction, paraphrasing, and natural language understanding, are strongly linked to the problem of semantic representation. These semantically-challenging tasks, as we call them, are not easily solved by the shallow, data-driven approaches that have come to dominate our field.

As researchers recognize the limitations with shallow approaches, they are beginning to apply more sophisticated linguistic information, as evidenced by the shift from word- and phrase-based models to syntactic models in statistical machine translation. However, deep grammars producing rich semantics are often dismissed due to concerns about coverage or complexity.

In this thesis, we show that deep grammars can make meaningful contributions to data-driven NLP by applying Head-driven Phrase Structure Grammars (HPSG) to several semantically-challenging tasks. Our parser produces Minimal Recursion Semantics (MRS), a formalism detailed enough to represent a variety of linguistic phenomena while remaining simple and flexible enough to avoid coverage and complexity issues.

We apply MRS to three tasks: (i) the expansion of a semantic transfer-based Japanese-English MT system, (ii) application of paraphrasing to improve a phrase-based statistical machine translation system, and (iii) ontological construction from machine-readable dictionaries achieving state-of-the-art performance for each one.

Keywords:

machine translation, ontology construction, paraphrasing, HPSG, MRS

*Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD0561041, March 22, 2010.

機械翻訳、言い換え、オントロジー構築への深い文法の適用*

エリック・ニコルズ

内容梗概

自然言語処理の分野においては、機械翻訳をはじめとして、言語資源構築、言い換え、自然言語理解等の、数多くのタスクが意味解析を必要とする。このようなタスクは“semantically-challenging tasks”と呼ばれ、現在自然 言語処理分野で主流となっている、統計処理を中心としたコーパスベースな どの比較的浅い処理で解決することはきわめて難しい。

統計的機械翻訳において、語やフレーズに基づくモデルから統語構造を利用 したモデルへと研究の深化があったように、こうした浅い処理に限界を感じ ている研究者は、更に深い言語に関する情報に目を向けている。しかしなが ら、意味情報を提供する深い文法は、文法の複雑さやカバレージに関しての 懸念により多くは利用されていない。

本稿では、Head-driven Phrase Structure Grammar (HPSG)をsemantically-challenging tasksに適用することでコーパスベー スの自然言語処理に対して深い文法が大きく貢献できる事を示す。ここで用 いるHPSG解析器が出力するMinimal Recursion Semantics (MRS)は、柔軟で扱 いやすいことから先に述べた複雑さやカバレージの問題に影響を受けにくく 多様な言語現象を示すことができる意味表現である。

ここでは、意味変換に基づく機械翻訳システムの拡張、フレーズベース統計 的機械翻訳の改善のための言い換えの適用、電子化された辞書からのオント ロジー構築のタスクで深い文法を用いることにより高い精度を得た。

キーワード

機械翻訳, 意味変換, 資源獲得, HPSG, MRS

*奈良先端科学技術大学院大学 情報科学研究科 情報処理学専攻 博士論文, NAIST-IS-DD0561041, 2010年3月22日.

Acknowledgements

I would like to thank everyone who helped make this thesis possible.

First, to my advisor, Professor Yuji Matsumoto: thank you for your guidance and support. It has been an honor and a privilege being part of your lab.

To my colleague, advisor, and mentor, Associate Professor Francis Bond: words cannot properly express my gratitude for all that you have done for me. Thank you.

I also thank the other members of my advisory committee, Professor Hiroyuki Seki and Associate Professor Kentaro Inui, for reviewing this thesis and giving me very useful feedback on how to improve its clarity.

I am deeply indebted to those who provided financial support for this research. My eternal gratitude goes to the Japanese government for their Monbukagakusho (MEXT) scholarship program that gave me the opportunity to study in Japan. I am also grateful for the additional financial support provided by Matsumoto lab.

Thank you to all of my co-authors and colleagues. Professor Dan Flickinger, Dr Sanae Fujita, Dr Takaaki Tanaka, Dr Hiromi Nakaiwa, Dr Chikara Hashimoto, Dr Shigeko Nariyama, Dr Michael Paul, Takayuki Kuribashi, D. Scott Appling, and many others: it has been a pleasure working with all of you.

I would like to thank the members of the Delphin community for the foundations that made this research possible. My thanks go to the grammar developers and maintainers, Professor Dan Flickinger, Dr Melanie Siegel, and Associate Professor Francis Bond; the developers of the parsers and frameworks, Professor Stephan Oepen, Doctor Ulrich Callmeier, and Dr Bernd Kiefer; and, finally, the developers of the theory, Professor Carl Pollard, Professor Ivan Sag, and Professor Ann Copestake.

I would like to express my gratitude to the developers of the tools and resources used in this research for their hard work. In particular, my thanks go to Dr Taku Kudo for Mecab and Cabocha; to Jim Breen for his excellent Japanese-English dictionary; to the late Professor Yasuhito Takana and his students for compiling the Tanaka Corpus; to the Moses developers; and to the WordNet and GoiTaikei developers.

Many thanks to Alex Shinn and Michael Goodman for risking their sense of English well-formedness to evaluate the output of our machine translation systems. I hope no lasting harm was done.

Finally, this thesis is dedicated to my mother. Mom, thank you for always believing in me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1 Motivation

Many challenging tasks in the field of natural language processing, such as machine translation, linguistic resource construction, paraphrasing, and natural language understanding, are strongly linked to the problem of semantic representation. These semantically-challenging tasks, as we call them, are not easily solved by the shallow, data-driven approaches that have come to dominate our field.

As researchers recognize the problems with shallow approaches, they are beginning to apply more sophisticated linguistic information, as evidenced by the shift from word- and phrase-based models to syntactic models in statistical machine translation and the increasing use of syntactic information in information retrieval tasks. However, deep grammars that produce rich semantic representations are often dismissed due to concerns about coverage or complexity.

In this thesis, we show that deep grammars can make meaningful contributions to data-driven NLP by using Head-driven Phrase Structure Grammars (HPSG) to achieve state-of-the-art performance on several semantically-challenging tasks. Our HPSG parser produces Minimal Recursion Semantics (MRS), a semantic formalism that is detailed enough to represent a variety of linguistic phenomena while remaining simple and flexible enough to avoid coverage and complexity issues.

2 Thesis Organization

We apply MRS to three tasks: the expansion of a prototype semantic transfer based machine translation system, ontological acquisition from machine-readable dictionaries, and applying paraphrasing to improve the performance of a phrase-based statistical machine translation system. This thesis is organized as follows.

In Chapter 2, we expand the semantic transfer based machine translation system by hand-crafting a small number of rules for easily-generalizable linguistic phenomena, and automatically acquiring more rules from bilingual dictionaries, parallel corpora via a bootstrapping alignment technique, and finally from an SMT system's phrase tables.

In Chapter 3, we reconsider the task of machine translation from a statistical perspective and use HPSG to improve the quality of phrase-based SMT by using paraphrasing to increase its training data. We achieve statistically significant improvements over a state-of-the-art baseline on two different corpora.

In Chapter 4, we automatically acquire ontological relations by parsing dictionary definition sentences, identifying the semantically most relevant word, and inferring the ontological relation with the definition word. We successfully construct ontologies for both Japanese and English from several dictionaries and align them with existing ontologies to evaluate their coverage and quality. Our method achieves state-of-the-art performance, outperforming shallow, pattern-matching based approaches.

In Chapter 5, we discuss our contributions to the open-source natural language processing community. These contributions include the foundation of Ubuntu NLP, a repository of NLP software packaged for Ubuntu Linux; development of *Moses Make*, a makefile-based system for automating the development and testing of Moses statistical machine translation systems; and our various contributions to the DELPH-IN deep processing community.

Finally, in Chapter 6 we summarize this thesis and discuss our contributions to the state of natural language processing.

Chapter 2

Semantic Transfer-based Machine Translation

1 Overview

In this chapter, we describe the expansion of a prototype Japanese→English semantic transfer machine translation system (Bond et al., 2005) based on the LOGON framework (Oepen et al., 2004a). We greatly increase the translation coverage and quality on two Japanese-English parallel corpora by using a combination of hand-crafted transfer rules to produce high-quality translations of closed category expressions, and automatically acquired rules to cover many open category expressions.

To automatically acquire transfer rules from a variety of sources, we extend the bilingual dictionary-based transfer rule acquisition method of Nygård et al. (2006) using templates to map arbitrary source-target pairs to valid LOGON rule types. With these templates, we acquire transfer rules from three sources: (i) a Japanese-English dictionary, (ii) the phrase tables of a statistical machine translation system, and (iii) parallel corpora via bootstrapping from partial transfer results.

We construct a phrase-based statistical machine translation system to use as a fallback system and baseline for comparison and conduct quantitative and qualitative evaluation of our system with automatic metrics and a small-scale human evaluation.

2 Motivation

While there have been many advances in the field of machine translation, it is widely acknowledged that current systems do not yet produce satisfactory results. At the same time, many researchers also recognize that no single paradigm solves all of the problems necessary to achieve high coverage while maintaining fluency and accuracy in translation (Way, 1999).

Data-driven approaches, like statistical machine translation and example-based machine translation, have gained a lot of attention for their ability to automatically acquire the resources necessary for machine translation for parallel corpora, however, by-and-large these approaches do not explicitly encode the linguistic knowledge necessary to handle Japanese-English and other divergent language pairs and generate natural, grammatical output.

On the other hand, rule-based and knowledge-based approaches suffer from coverage problems. The resources necessary for these systems are difficult and costly to construct, and it is infeasible to manually encode all necessary linguistic knowledge by hand. In addition, many knowledge-based systems are proprietary in nature, limiting resource sharing between systems. This often leads to researchers often reinventing the wheel when building new systems.

We take the position that translation is fundamentally a problem of meaning preservation, and that detailed linguistic analysis is essential in meeting the goal of high-quality translation. However, faced with the aforementioned problems, we recognize that any new approaches need to focus on striking a balance between automatic acquisition of translation resources and manually encoded linguistic knowledge.

3 Research Goals

The ultimate aim of this research is to have a robust, high quality and easily extensible Japanese→English machine translation system. Current statistical MT systems are robust and of fair quality, but only for those domains and language pairs where there is a large amount of existing parallel text. Changing the type of the text to be translated causes the quality to drop off dramatically (Paul, 2006). Quality is proportional to the log of the amount of training data (Och, 2005), which makes it hard to quickly extend a

system. Rule-based systems can also produce high quality in a limited domain (Oepen et al., 2004a). In addition, it is relatively easy to tweak rule-based systems by the use of user dictionaries (Sukehiro et al., 2001), although these changes are limited in scope.

This leaves the problem of how to build a system that is both high quality and easily extensible. To gain high quality, we accept the brittleness of a rule-based semantic transfer system. In particular, by using a precise grammar in generation we ensure that the output is almost always grammatical. Rule types are hand-made. As far as possible we share types with the Norwegian→English system developed in the LOGON project (Oepen et al., 2004a). To make the system easier to extend, we construct transfer rules instances from a plain bilingual dictionary. As far as possible, we aim to concentrate our rule building efforts on closed-class words, and then fill in the open class transfer rules by automatic conversion of the bilingual lexicon. Finally, we learn extra rules from parallel corpora.

3.1 High Quality Translations

While online translation services such as Babelfish, Altavista, and Google Translate have helped popularize the use of machine translation to get the gist of foreign language documents, there are still many tasks that require high quality translation output that preserves the meaning of the source text.

Statistical approaches that make use of little structural information are at a disadvantage in this case. Often subtleties in word order can result in a drastically different meaning, and pronouns, markers of gender or agreement, and grammatical indicators such as negation can be difficult to learn with n-gram models.

MRS contains the linguistic knowledge required for handling agreement, word order, subcategorization and other phenomena. In addition, MRS represents syntactic categories in a type-hierarchy structure, providing generalizations that simplify transfer rule development, and it enumerates subcategorization information that is useful in cross-lingual alignment.

3.2 Flexible Knowledge Representation

Our goal is to produce a machine translation system that represents its translation knowledge in a format flexible enough that transfer rules can be acquired automati-

cally yet still be understandable to humans. This will allow users to supplement the system's rules with their own linguistic knowledge, making the system customizable to their needs. The Japanese-English MT system we present in this thesis uses a flexible semantic representation produced by high-coverage lexical grammars as its transfer language. This approach gives our system access to the knowledge it needs to generate natural language, while at the same time, the transfer language is sufficiently abstracted away from the syntactic level to eliminate rules with language-dependent features such as word order. Our system makes it easy to represent alignments on a linguistically-meaningful level.

3.3 Automatically Acquired Knowledge

It is impractical to attempt to manually encode all of the linguistic knowledge necessary for high-quality machine translation. We take a pragmatic approach in the construction of our system by hand-crafting rules for only the most essential linguistic phenomena. Rules of a lexical nature are automatically acquired from dictionaries and parallel corpora. By combining engineered knowledge and automatically acquired resources in this manner, our system can achieve robust coverage and high translation quality.

3.4 Open Source Resources

Similar machine translation projects have been worked on in the past as summarized in the following section. However, the majority of these systems are of a proprietary nature; when the project concludes, the resources that were developed are often not made available to other researchers, so it is difficult for the field to directly benefit from what was learned throughout the course of development.

We recognize the problem that closed resources poses to machine translation systems that use detailed linguistic representations, so one of our goals is to make as many of our resources freely available to other researchers. Every component of our machine translation system, from the parser to the grammars, is available as open source. In addition, all of the transfer rules that we have produced are also freely distributable

4 Related Research

In this section, we describe related efforts in machine translation. We limit discussion to recent systems that use rich semantic resources, target the Japanese→English language pair, and/or have been released as open source.

4.1 KBMT-89

KBMT-89 (Goodman and Nirenburg, 1989) is a Japanese↔English machine translation system developed at Carnegie Mellon In the late 1980s. It is developed targeting a small collection of text from IBM PC installation and maintenance manuals.

Although KBMT-89 used *interlingua texts* (ILTs) in place of a transfer mechanism, its creators consider it a knowledge-based approach to machine translation, as is clear from its name. This was likely to emphasize the major role played in the translation process by rich syntactic and semantic resources. KBMT-89 uses Lexical Functional Grammar (LFG: (Kaplan and Bresnan, 1982)) for syntactic analysis and generation, and its ILTs are built with semantic knowledge encoded in ontologies and lexicons using the FRAMEKIT knowledge representation framework.

Translation is carried out in KBMT-89 by parsing input text into LFG f-structures and using syntax→semantics structural mapping rules to produce ILTs. A module called the *interactive augmenter* is used to disambiguate the ILTs, then semantic and syntactic generation processes produce the final translation.

While the KBMT-89 project made many contributions to the state of the art of knowledge representations for Japanese and English, its success as a machine translation system was more limited. It was designed for a very narrow domain of text (only 300 sentence pairs were used in its development), it was unclear how effectively the system could be automatically extended, and there was little concrete evaluation of its translation quality.

4.2 ALT-J/E

ALT-J/E, Automatic Language Translator - Japanese to English (Ikehara et al., 1991), is a machine translation system that was developed at NTT Communication Science Laboratories in the 1990s and early 2000s. Its goal was speech-to-speech translation

over the telephone, requiring high quality translation with little or no pre-editing.

ALT-J/E uses a *Multi-Level Translation* paradigm. The system's Japanese input is analysed and split into a *subjective expression* containing tense, aspect, and modal information, and an *objective expression* containing the *kernel sentence*. The subjective and objective expressions are transferred independently and recombined to generate an English translation. The objective expression undergoes transfer with multiple levels of analysis. First parse-tree based transfer rules are applied, followed by idiomatic expression transfer and transfer based on semantic valency information. Finally, general patterns are used to transfer any remaining expressions.

ALT-J/E has several noteworthy features. It employs semantic dictionaries containing thousands of ontological relations used to identify idioms, multi-word expressions, and semantic relations between verbs and arguments that require special translations. **ALT-J/E** also automatically rewrites Japanese expressions that cannot be handled by the system into easier to translate forms. In order to provide all of the information necessary for English translations, it also conducts discourse analysis at the paragraph level, filling in ellipsed Japanese arguments, and uses fine-grained lexical distinctions to generate English noun phrases with the correct determiners and countability, and to naturally position adverbs. A Japanese-Malay prototype MT system (Ogura et al., 1999) was also developed with the **ALT-J/E** framework.

4.3 Verb*mobil*

The Verb*mobil* project (Wahlster, 2000) was a large-scale international collaboration to build a speech-to-speech translation system for German, English, and Japanese. In was centered at DFKI in Saarbrücken, Germany, included organizations throughout Europe, North America, and Asia, and ran from 1993 to 2000.

Verb*mobil*'s goal was real-time dialog interpretation over the mobile phone to provide users with context-sensitive translations in three business domains. The system used a *multi-blackboard* translation paradigm combining deep and shallow processing to achieve its goal. To achieve robust processing of spontaneous dialog, Verb*mobil* used an underspecified, packed representations that captured non-deterministic nature of language and allowed for the incorporation of linguistic, discourse, domain constraints as they became available throughout analysis.

Verb*mobil* combined five translation engines: statistical translation, case-based

translation, substring-based translation, dialog-act based translation, and semantic transfer using Head-driven Phrase Structure Grammars (Pollard and Sag, 1994; Sag et al., 2003), and it was noteworthy for combining deep and shallow processing, using statistical models to choose the best result for each step in the translation process. Many of the resources from the Verb*mobil*, including the HPSG grammars used in this research, were released to the research community and continue to contribute to NLP research.

4.4 Data Oriented Translation

Data Oriented Translation (DOT) (Poutsma, 2000) is a syntactic tree transfer-based approach to machine translation. Data-Oriented Parsing is used to parse source language text into syntactically-labeled phrase structure trees, and tree fragments are transferred into the target language. Parsing and transfer use packed representations to preserve possible interpretations, with Monte Carlo statistical models used to choose the most likely translation. DOT is similar to syntactic approaches to statistical machine translation such as Chiang (2005) and Watanabe et al. (2006), however, its creators describe it as an example-based approach. Extensions using LFG to enforce grammaticality constraints or improve translation selections have been proposed (Way, 1999), however, efforts have mainly focused on increasing the size of parallel treebanks and decreasing the number of interpretations produced in the translation process.

4.5 Open Source MT

Recently, several large open source machine translation projects have been started. Section 5.1 describes the LOGON system, which provides many of the components for our Japanese→English system, Here, we will discuss two other large systems: OpenTrad and OpenLogos.

OpenTrad is a Spanish open source translation initiative consisting of a general MT framework and two engines (Armentano-Oller et al., 2005). The engines are Apertium, a shallow transfer system used for Spanish↔Catalan, Galician, and Portuguese, with other languages recently added, including English and French. There is also a structural transfer system used for Spanish↔Basque. Both systems share components (tokenizer, deformatter, reformatter, etc.) and are released under the GNU Public License

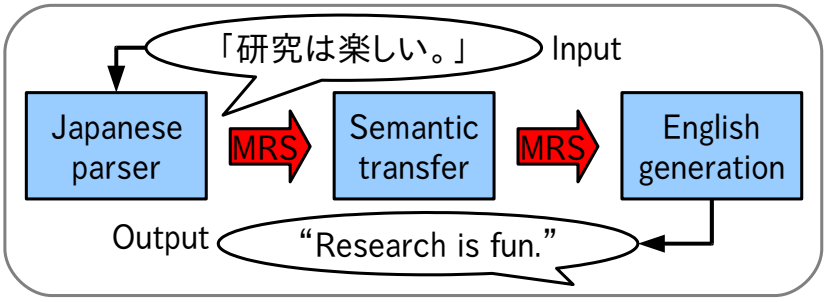


Figure 2.1. The JaEn machine translation architecture

(GPL)[1].

OpenLogos is a commercial transfer-based system (Scott, 2003) that was developed in the late 1970s and released as open source in 2005[2]. It can translate from German or English into a number of languages including French, Italian, Spanish, and Portuguese. The system is released under a dual license (commercial/GPL).

5 The JaEn MT System Architecture

The first version of this system is described in detail in Bond et al. (2005). The architecture of our Japanese→English system (hereafter referred to as JaEn) is semantic transfer via rewrite rules, as shown in Figure 2.1. The source text is parsed using an HPSG grammar for the source language, and a semantic analysis in the form of Minimal Recursion Semantics (MRS) is produced. That semantic structure is rewritten using transfer rules into a target-language MRS structure, which is finally used to generate text from a target-language HPSG grammar.

Statistical models are used at various stages in the process. There are separate models for analyses, transfer and generation, combined as described in Oepen et al. (2007). At each stage we prune the search space, only passing n different results (5 by default) to the next stage.

[1] `http://www.gnu.org/licenses/gpl.html`

[2] According to the versioning at `http://logos-os.dfki.de/release/`

5.1 Semantic Transfer Architecture

The architecture of our Japanese→English system (hereafter referred to as "JaEn") is semantic transfer via rewrite rules, as shown in Figure 2.1. The source text is parsed using a head-driven phrase structure grammar (HPSG) for the source language. This produces semantic representations of the input in the form of Minimal Recursion Semantics (MRS). That semantic representation is rewritten using transfer rules into a target-language MRS structure, which is finally used to generate text from a target-language HPSG grammar.

Statistical models are used at various stages in the process. There are separate models for analyses, transfer and generation, combined as described in Section 5.5. At each stage we prune the search space, only passing n different results (5 by default) to the next stage.

The grammars and processing systems we use are all being developed within the DELPH-IN[3] project (Deep Linguistic Processing with HPSG Initiative) and are available for download. We also use the statistical machine translation system Moses (Koehn et al., 2007), both as a back-off and as a source of alignment data. We learn lexical equivalences from the Japanese-English dictionary JMdict (Breen, 2004), and developed targeting the Tanaka Corpus (Tanaka, 2001).

We illustrate the transfer process using an example. Consider the Japanese sentence (1) and its semantic representation (2). The building blocks are elementary predications (EPs), like **uso_n_2**(x_{10}) "lie" and **tsuku_v**(e_2, x_4, x_{10}), "attach" corresponding to atomic formulas in predicate logic. Quantifiers (e.g. **sono_q**) "that" introduce special relations in an MRS, corresponding to generalized quantifiers, these are introduced for all noun phrases, even if there is no such quantifier in the original text (e.g., the underspecified quantifier **udef_q**). All EPs are labeled with *handles*, e.g. h_9 is the label on the predication **uso_n**(x_{10}).

(1) その うそ は 子供 たち が　　ついた
sono uso-wa kodomo-tachi-ga tsui-ta
that lie-TOP child-PL-NOM attach-PAST
The children told that lie.

(2) $\langle h_1, \{h_3$: **kodomo_n**(x_4)

[3] http://www.delph-in.net

h_3: **tachi_a**(u_1, x_4)
h_5: **udef_q**(x_4, h_7, h_6)
h_9: **uso_n_2**(x_{10})
h_9: **wa_d**(e_3, e_4, x_{10})
h_{11}: **sono_q**(x_{10}, h_{13}, h_{12})
h_{11}: **tsuku_v**($e_2, x_4, x_{10}[tense : past]$) },
$\{h_7 =_q h_3, h_{13} =_q h_9, \}\rangle$

MRS representations abstract away from the surface syntactic structure in several ways, even though it is far from being an interlingua. For example, in (2) the noun phrase *sono uso* "that lie" is recognized as the object of the verb *tsuku* "attach", even though it comes before the subject, and is marked with a topic marker *wa* "topic", rather than the accusative marker. Also, in the noun phrase *kodomo-tachi*, the noun *kodomo* "child" is recognized as the semantic head of the phrase, and the suffix *-tachi* "plural, literally *others*" is treated as a modifier.

Given below are the English MRS (3) and generation result (4) from the semantic transfer of the Japanese MRS in (2).

(3) $\langle h_1, \{h_3$: **child_n_1**($x_4[number : plural]$)
h_5: **def-udef-a_q**(x_4, h_7, h_6)
h_9: **lie_n_1**($x_1 0$)
h_{11}: **that_q**($x_1 0, h_1 3, h_1 2$)
h_{11}: **tell_v_1**($e_2, x_4, x_{10}[tense : past]$) },
$\{h_7 =_q h_3, h_{13} =_q h_9, \}\rangle$

(4) (The) children told that lie.

Although the syntax is quite different, the MRS is similar. There are three main differences. The first, and most obvious, is that the predicate names are different. There are simple translation rules that transform, e.g., **uso_n_2**(x_i) into **lie_n_1**(x_i). A slightly more complicated rule is used to translate **tsuku_v**(e_i, x_j, x_k). The rule is conditioned on its object. If the object of **tsuku_v**(e_i, x_j, x_k) matches **uso_n**(x_j), then rewrite it to **tell_v_1**(e_i, x_j, x_k), otherwise rewrite it to **attach_v_to**(e_i, x_j, x_k).

The simple rules can be semi-automatically compiled from a bilingual dictionary, as described below, but adding context and ordering the rules correctly is currently

done by hand. The order of predicates shown here is also different, but this is unimportant, the EPs are an unordered bag - all the information necessary to order the output text is given in the handles. This flat structure makes it easy to apply rules to MRSs - rules can apply anywhere in the structure, so long as they preserve the relations between handles and arguments.

More interestingly, in the English MRS the Japanese plural marker has disappeared, and instead the noun phrase it used to modify has been made plural. Finally, the underspecified quantifier **udef_q** has been replaced by an underspecified quantifier that is a supertype of the English articles (*a*, *the* or no article): **def-udef-a_q**. This leaves the choice of article to the English generator. The topic marker is ignored here (and deleted in transfer). Ideally it should be used to influence the choice of article and possibly trigger topicalization in English: *That lie the children told*.

The final English translations are shown in (5). Because *child* is constrained to be plural, the only possible determiners are *the* or no article, so these two possibilities are generated.

(5) a. The children told that lie.

b. Children told that lie.

The LOGON semantic transfer architecture is a fairly standard transfer method. The main strength is that it is being applied at the semantic level, not at the syntactic level (such as systems that transfer trees or dependencies). This allows the source and target grammars do much of the work, allowing the transfer to be simpler.

Of course, this approach is far from solving the problems of machine translation. The problems of word sense disambiguation remain, as well as problems due to other differences in meaning representation in languages. For example, Japanese does not distinguish between singular and plural, or countable and uncountable, so most noun phrases are very underspecified. We currently approach these problems by producing multiple candidates and selecting using stochastic models, taking advantage of improvements in empirical methods, as described in Section 5.5.

5.2 Grammars

For parsing and generation we use HPSG-based grammars of Japanese and English from the DELPH-IN project (JACY; (Siegel, 2000) and the English Resource Grammar

(ERG; (Flickinger, 2000)). Both grammars were originally developed within the Verb-*mobil* machine translation effort, but over the past years have been used for a variety of tasks, including automatic email response and extracting ontologies from machine readable dictionaries.

The grammars are being developed by separate groups of researchers, but share a commitment to the same semantic representation: MRS (Copestake et al., 2005). This is a precise, but underspecified, language-specific semantic representation. MRS structures are flat, unordered collections of elementary predications (EPs) with handles (h) indicating scopal relations, events (e), and entities (x). MRS provides several features that make it attractive as a transfer language, such as uniform representation of pronouns, specifiers, temporal expressions, and the like over grammars. More details can be found in Flickinger et al. (2005).

5.3 Processing Engines

For parsing we use PET an efficient parser for unification-based grammars (Callmeier, 2000). For generation, and general grammar development we use the LKB (Copestake, 2002). For transfer and overall system integration we use the LOGON architecture (Oepen et al., 2004a) which is integrated into the LKB.

5.4 Transfer Formalism

As illustrated in (Oepen et al., 2004a), transfer rules take the form of MRS tuples:

```
[CONTEXT:]IN[!FILTER]->OUT
```

where IN(PUT) is rewritten by OUT(PUT), and the optional CONTEXT specifies relations that must be present for the rule to match, and conversely, FILTER specifies relations whose presence blocks a rule from matching. Consider the following transfer rule to translate うそ *uso* into *lie*:

(6) `uso-lie-mtr` = $\langle h_1$: **`uso_n_2`**$(x_i) \rightarrow h_1$: **`lie_n_1`**$(x_i)\rangle$

This rule rewrites any instance of **`uso_n_2`** with **`lie_n_1`**. h_1 and x_i indicate that the LBL and ARG0 arguments of the MRS produced must be preserved. While this may seem like a fairly easy to understand rule, we must repeat the constraint

on LBL and ARG0 every time we write a rule to translate nouns. In order to avoid such redundancy in rule writing, LOGON allows the user to specify rule types that can encapsulate common patterns in rules. The above rule can be generalized to cover nouns:

(7) `noun-mtr` = $\langle h_1\colon(x_i) \rightarrow h_1\colon(x_i)\rangle$

and our example rule can be rewritten as:

(8) `uso-lie-mtr` = $\langle$`noun-mtr &` **`uso_n_2`** $\rightarrow$ **`lie_n_1`**$\rangle$

We were able to share many rule types with the LOGON Norwegian-English system. It contains a rich definition of rule types - many of which were immediately applicable to JaEn. JaEn inherits from LOGON rule types for open category lexical items such as common nouns, adjectives, and intransitive & transitive verbs. In addition, LOGON contains a number of rule types to specify rules for quantifiers, particles, and conjunctions, providing much of the framework needed to develop JaEn. As a practical matter, all the machine translation systems developed with the LOGON architecture now share the upper rule types to the extent that each system can load the same files.

5.5 Ranking Translations in JaEn

JaEn uses a combination of five different stochastic models to rank its translations. The first is used when creating translation rules from dictionaries: rules are ordered according to the phrase table probabilities calculated by Moses (Koehn et al., 2007). Then models are used in each phase of parsing, ranking and generation. A cut-off of five is placed at each step, so only the five top ranked analyses are passed on to transfer, which in turn will pass a maximum of twenty five MRSs to generation (five per branch). Finally the top five realizations in each branch are gathered together and the results are reranked using a global reranking model. The dictionary ranking was described in Section 7.3, the remaining ranking models are described here.

Parse Ranking

Parse ranking is done using a model trained on 7,000 treebanked sentences from the training set of the Tanaka Corpus (Bond et al., 2008). The model is a discriminative

log-linear model, which uses features from the parse derivations such as local subtree configurations, dominance relations and n-grams of lexical types (Fujita et al., 2007). The model is trained using the machinery developed in Redwoods (Oepen et al., 2004b).

Transfer Ranking

To rank the transfer output, we use the model built in the LOGON Norwegian-English machine translation system (Oepen et al., 2007). It ranks the semantic representations using elementary semantic dependencies derived from the MRSes. Because we did not have an English treebank for the Tanaka Corpus, we used the LOGON model, which was trained on the Norwegian hiking corpus.

Realization Ranking

To rank the generator results, we again used the model built by in the LOGON project (Oepen et al., 2007). It uses a combination of word n-grams from the British National Corpus (Burnard, 2000) and a syntactic model trained on an English treebank (Velldal and Oepen, 2006).

We found that the model did not deal very well with choices such as *the dog barks* vs *dogs bark*. It almost always preferred the shorter string, even though *dog* is roughly twice as frequent as *dogs* in the British National Corpus. A solution to this is to train a model where the input is underspecified (**def-udef-a_q**(x) **dog_n_1**($x[number : number]$)) and the target string is *the dog*. This would allow us to learn, for example, that in this context **article_q** should go to *the* (**def_q**) and **dog_n_1**($x[number : number]$) to *dog* (**dog_n_1**($x[number : singular]$)). In a generation corpus based on parsing English sentences, the semantic representation will never be underspecified in this way, so we cannot learn such a model. Instead, we have to take the corpus and rewrite the semantic representation to be more general and then learn from this.

Translation Re-ranking

Once again, we use the generator from the LOGON project (Oepen et al., 2007) to rerank the final translation results. After testing various combinations, JaEn ended up using a combination of the parsing model (weight 0.2), transfer model (weight 0.2),

Frequency	Semantic relation	Translation
25,927	_ni_p	に → in, to, into
25,056	cop_id	だ, です → to be
22,976	_no_p	XのY → X Y, X's Y, Y of X
10,375	_de_p	で → in, on, at, with
9,696	rareru	られる → passive
9,528	neg_v	ない → negation
8,848	_exist_v	ある → to be, to have
7,627	_kono_q	この → this
4,173	tai	たい → to want to
3,588	_hour_n	時 → time, hour

Table 2.1. Highest frequency source language relations and their translations

generation model (weight 0.1) and a simple n-gram based language model based again on the BNC (weight 0.5). Unlike Velldal and Oepen (2006), we got no improvement using either a distortion model or lexical translation probabilities, although we only investigated a limited grid of weight combinations.

6 Hand-crafted Transfer Rules and Rule Types

6.1 Closed Category Transfer Rules

In order to decide which semantic relations to write transfer rules for by hand, we used the automatically acquired translation rules in the above section and attempted to translate sentences from the BTEC corpus. Whenever a relation failed to transfer, the system would be unable to generate a translation, and an error message was produced. We counted the relations and identified the most frequently occurring closed class relations as candidates for handcrafting a transfer rule. There are currently a total of 195 handcrafted rules in our system. Table 2.1 lists the translations we hand-crafted for the ten most frequently occurring semantic relations.

In handcrafting transfer rules for our system, we also encountered several linguistic problems that needed to be solved in order to achieve high-quality translation results, the most interesting of which was pronoun generation in English. Since our Japanese

semantic analyses indicate when arguments of a predicate have been omitted, we came up with a small set of rules that checks what restrictions, if any, are placed on the omitted arguments, and we replace them with underspecified English pronouns, since the nature of the omitted argument is unknown. This leads to over-generation of pronouns, which can cause a combinatorial explosion in the number of translations for sentences with multiple ellipsed pronouns. To avoid this problem, we only allow pronouns to be inserted for the first two argument slots (roughly corresponding to *subject* and *object*).

Other advances made include the treatment of common modal verbs, and natural generation of determiners for negative clauses. A more detailed description of hand-crafted rules and phenomena covered is given in Appendix C.

6.2 Rule Types Unique to JaEn

Here, we briefly describe a few rule types that were developed to handle linguistic phenomena unique to Japanese→English translation.

In JaEn, Japanese verbal nouns are analyzed as events, and they produce messages accordingly. When it is being used as a noun, **kenkyuu_s** is wrapped with the relation noun-relation. We handle these constructions with a special rule that nominalizes the verbal noun by removing its event and the associated message and replacing them with an entity when it appears as a noun:

```
vn-n_jf := monotonic_mtr &
 [ CONTEXT.RELS < [ PRED "ja:udef", ARG0 #x0 ] >,
   INPUT [ RELS < [ PRED "ja:noun-relation",
                    LBL #h6, ARG0 #x0, ARG1 #hp ],
                  [ PRED "ja:proposition_m",
                    LBL #hp, ARG0 #ep, MARG #h5 ],
                  [ PRED #pred, LBL #h0, ARG0 #ep ] >,
           HCONS < qeq & [ HARG #h5, LARG #h0 ] > ],
   OUTPUT [ RELS < [ PRED #pred, LBL #h6, ARG0 #x0 ] >,
            HCONS < > ] ].
```

In short, this rule type removes the noun-relation and all semantic relations resulting in the verbal noun's analysis as an event. This change makes it possible to treat verbal nouns identically to regular nouns in the rest of the transfer rules, eliminating

the need to create multi-word transfer rules that have to distinguish between nouns and verbal nouns. This simplifies rule development significantly. Thus, a rule to translate *kenkyuu* as the noun “research” can now be created using the standard noun template:

```
kenkyuu_s-research_n-omtr := noun_mtr &
 [ IN.RELS <[ PRED "_kenkyuu_s" ] >,
   OUT.RELS <[ PRED "_research_n_1" ] > ].
```

6.3 Handling Translation Phenomena in JaEn

JaEn’s primary source of transfer knowledge are transfer rules that are acquired from JMdict or learned directly from the Tanaka Corpus, however, there are a small number of words and phrases that have a great impact translation quality, but are not found in dictionaries. We think of these items as closed-class in nature and consider it more efficient to hand-craft transfer rules for them rather than attempting to automatically acquire them. In this section, we describe the rules we have written to handle some of the translation phenomena that makes Japanese→English MT such a difficult task.

6.4 Multi-Word Expressions

Multi-word expressions (MWEs) are an area of challenge for all machine translation systems. Many MWEs are lexical or collocational in nature, however some require structural knowledge to translate correctly. Consider the following example:

(9) 買い物 に 行か な けれ ば なら ない 。
kaimono-ni ika-na-kere-ba nara-nai .
shopping-LOC go-NEG-COND become-NEG .

I have to go shopping. (Reference)

If you go shopping. (Moses)

It must go on a shopping. (JaEn)

The Japanese string *ikanakereba naranai* needs to be translated as *must*, but it is composed of two negated verbs connected in a coordinate structure by the particle *kereba*. We can write a transfer rule to accommodate this as so:

(10)

$$\texttt{must-mtr} = \left\langle \begin{array}{lll} h_1 : \textbf{event}(e_1) & \rightarrow & h_1 : \textbf{event}(e_1) \\ h_2 : \textbf{eba\_c}(e_2, h_3, e_1, h_4, e_3) & & h_2 : \textbf{must\_v}(e_2, h_1) \\ h_3 : \textbf{neg}(h_1) & & \\ h_4 : \textbf{neg}(h_5) & & \\ h_5 : \textbf{naru\_v}(e_3) & & \end{array} \right\rangle$$

This rule links the left-hand side argument of the verbal conjunction **eba_c** to **must_v** and discards both of the verbal negations and **naru_v**. Without this rule, JaEn would literally translate this sentence as "It does not become if it does not go on a shopping." Although JaEn gets the pronoun wrong in its translation, it preserves the essential structure of this sentence better than Moses, which also cannot translate the pronoun correctly.

We use rules of a similar formation to handle variations of this pattern such as [*nakereba*/*naito*/*nakute*]*ikenai*. Other structural MWEs handled include *no-tame* → "because of," *koto-ga-dekiru* → "can," and [*koto*/*no*]*ga-suki* → "like to."

6.5 Generalizing Over Structure

MRS-based transfer makes it easy to generalize over structures. This eases translation from Japanese into English in several ways.

Example (11) shows how questions can be difficult for SMT systems to translate due to their unusual word order. This sentence is not a problem for JaEn: both arguments of 作る *tsukuru* "make" are embedded in its arguments structure and transferred to the English grammar intact.

This sentence also gives an example of a politeness marker in Japanese: *mashita*. The Japanese parser converts the polite *tsukurimashita* to its standard form, *tsukuru*, allowing general translation rules to work without change.

(11) (あなた は) 何 を 作り まし た か 。
(anata-ha) nani-wo tsukuri-mashi-ta ka .
you-TPC what-ACC make-POLITE-PAST QUES .
What did you make? (Reference)

What made you? (Moses)

What did you make? (JaEn)

This sentence also demonstrates how JaEn handles zero pronouns. Since our Japanese semantic analyses indicate when arguments of a predicate have been omitted, we came up with a small set of rules that checks what restrictions, if any, are placed on the omitted arguments, and we replace them with underspecified English pronouns, since the nature of the omitted argument is unknown. This leads to over-generation of pronouns, which can cause a combinatorial explosion in the number of translations for sentences with multiple ellipsed pronouns. To avoid this problem, we only allow pronouns to optionally be inserted for the first two argument slots (roughly corresponding to *subject* and *object*).

Another example of generalization is given in (12), where the passivization of the Japanese sentence is represented as a feature on the verb, which is preserved in the semantic transfer stage. This allows English generation to produce both active and passive voice variants, which it considers semantically equivalent.

(12) 犯人 は 警察 に 逮捕 され た 。
hannin ha keisatsu-ni taiho sare-ta .
criminal-TOP police-DAT arrest PASV-PAST .

The criminal was arrested by the police. (Reference)

The criminal was arrested by the police. (Moses)

Police arrested the criminal. (JaEn)

7 Automatic Rule Acquisition

In this section, we present the results of automatic transfer rule acquisition from the Japanese-English bilingual dictionary, JMdict, and directly from Moses' phrase table.

7.1 Parallel Corpora

The BTEC Corpus

As development and testing data, we are currently using the ATR Basic Travel Expression (BTEC) Corpus as made available in the IWSLT 2006 evaluation campaign (Paul, 2006). As is indicated by its name, the BTEC corpus consists of short spoken sentences taken from the travel domain. We selected it because it is a commonly used development set, making our results immediately comparable to a number of different systems, and because the Japanese HPSG parser and grammar we use can successfully analyze approximately 65% of its sentences, providing us with a good base for development. The BTEC data supplied in the IWSLT 2006 evaluation campaign consists of almost 40,000 aligned sentence pairs. Sentences average 10.0 words in length for Japanese and 9.2 words in length for English. There are 11,407 unique Japanese tokens and 7,225 unique English tokens.

The Tanaka Corpus

The Tanaka Corpus is an open corpus of Japanese-English sentence pairs compiled by Professor Yasuhito Tanaka at Hyogo University and his students (Tanaka, 2001) that was released into the public domain.

Professor Tanaka's students were given the task of collecting 300 sentence pairs each. After several years, 212,000 sentence pairs had been collected. The sentences were created by the students, often derived from textbooks, e.g. books used by Japanese students of English. Some are lines of songs, others are from popular books and Biblical passages. The original collection contained large numbers of errors, both in the Japanese and English. These are being corrected by volunteers, as part of ongoing activity to provide example sentences for the Japanese-English dictionary JMdict (Breen, 2003). Recently, translations in other languages, most notably French, have been added by the TATOEBA project.[4] We give a typical example sentence in (13).

(13) あの 木 の 枝 に　　数羽 の 鳥 が　とまっ て いる 。
ano ki no eda-ni　　suuwa no tori-ga tomat-te iru　.
that tree's branch-LOC several bird-ACC are staying　.
Some birds are sitting on the branch of that tree. (en)

[4] http://wwwcyg.utc.fr/tatoeba/

Des oiseaux se reposent sur la branche de cet arbre. (fr)

The version (2008-11) we use has 147,190 sentence pairs. We hold out 4,500 sentence pairs each for development and evaluation data.

7.2 Converting Translation Pairs into Transfer Rules

Nygård et al. (2006) demonstrated that it is possible to learn transfer rules for some open category lexical items using a bilingual Norwegian→English dictionary. They succeeded in acquiring over 6,000 rules for adjectives, nouns, and various combinations thereof. Their method entailed looking up the semantic relations corresponding to words in a translation pair, and matching the results using simple pattern matching to identify compatible rule types.

Our approach generalizes this approach by using rule templates to generate transfer rules from input source and target MRS structures. Template mappings are used to identify translation pairs where there is a compatible rule type that can be used to create a transfer rule. A template mapping is a tuple consisting of:

- a list of HPSG syntactic categories corresponding to the words in the source translation
- a list of HPSG syntactic categories for the target translation words; and
- the name of the rule template that can be used to construct a transfer rule

Consider the following template mapping:

```
<[noun], [adjective, noun], n-adj+n)>
```

This template mapping above identifies a template that creates a rule to translate a Japanese noun into an English adjective-noun sequence, e.g., `mtr` = ⟨悪玉 *aku-dama* → bad character⟩.

Transfer rule generation is carried out in the following manner:

1. Look up each word from source-language translation in HPSG lexicon
    - Retrieve syntactic categories and MRS relations

- Enumerate every possible combination for words with multiple entries
- Refactor results into separate lists of syntactic categories and MRS relations

2. Repeat 1. for all words in target-language translation
3. Map template mappings onto source and target syntactic categories
    - Translations that match indicate the existence of compatible rule template
4. Create a transfer rule by combining the rule template and lists of source and target MRS relations

Using this algorithm we can extract rules from any list of translation pairs. After acquiring transfer rules, we sort them by frequency in the Tanaka Corpus, keeping a maximum of three rules for each input phrase. We create a only one rule for out-of-vocabulary entries, using the first listed dictionary translation to match. Transfer rules are ordered so that transfer words that cover multiple source words are applied first.

7.3 Dictionary-based Rule Acquisition

Our primary source of rules is JMdict. We split compound JMdict entries into individual translation pairs and pass the translation pairs to the algorithm described in Section 7.2. The results of open category transfer rule acquisition from JMdict are summarized in Table 2.2. This is our main source of transfer rules, with 39,120 acquired.

7.4 Acquisition from Moses Phrase Tables

We also acquire translation rules from Moses' phrase table. Moses' phrase table consists of a Japanese chunk, an English chunk and five weights: phrase translation probability $\phi(j|e)$, lexical weighting $lex(f|e)$, phrase translation probability $\phi(e|j)$, lexical weighting $lex(e|f)$ and a phrase penalty (always exp(1) = 2.718). Examples phrase table entries for うそ *uso* "lie" are given in Figure 2.2. We sum all five of the weights and probabilities given for each entry in the phrase table and keep any entries with $total >= 4.0$.

We acquired a total of 4,209 entries from Moses' phrase tables. We found that there were problems for us in that non-constituents were often learned. For example, noun

Rule type	JMdict	Examples
Verb→Adj	272	あり得る → likely
Adj→Verb	276	不安 → to worry
Adj+Noun→Adj+Noun	730	渋いワイン → white wine
Proper Noun	833	シンガポール → Singapore
Adverb	1,543	大変 → very
Intransitive Verb	1,991	現れる → to appear
Noun→Adj	2,470	最低限 → minimum
Adjective	3,040	青い → green
Noun→Adj+Noun	3,684	悪玉 → bad character
Noun→Noun+Noun	4,278	甘党 → sweet tooth
Noun+Noun→Noun	4,378	アイデア商品 → novelty good
Transitive Verb	5,715	選ぶ → to choose
Noun+Noun→Adj+Noun	5,791	暗黒物質 → dark matter
Noun+Noun→Noun+Noun	8,053	原価管理 →cost management
Noun	19,270	字 → character
Total	62,324	

Table 2.2. Semantic transfer rules acquired from JMdict

entries would be surrounded by prepositions or verbs, creating invalid entries such as *okane* → “money out” or *okane*→ “turn money.”

A hand evaluation showed that 274, approximately 7%, of these entries were bad alignments. We removed or edited these entries into correct rules, leaving a total of 3,935 entries. These two extensions make it possible to produce transfer rules only for those entries which are true translations.

7.5 Bootstrapping from Partial Transfers

In order to acquire transfer rules from a parallel corpus, we compare partial transfer results to the parses of target language sentences. Partial transfer results occur when a Japanese sentence cannot be completely translated into English by JaEn. The resulting MRS structure contains both English and Japanese relations. By using the English relations found in both the partial transfer MRS and the target sentence MRS, it is

```
うそ ||| a lie ||| (0) ||| (0) () ||| 0.015873 0.0022841 0.0...
うそ ||| a white ||| (0) ||| (0) () ||| 0.0322581 0.0022841 ...
うそ ||| a ||| (0) ||| (0) ||| 0.000179738 0.0022841 0.1 0.3...
うそ ||| absolute lie ||| (0,1) ||| (0) (0) ||| 0.5 0.086754...
うそ ||| be false ||| (1) ||| () (0) ||| 0.111111 0.0388889 ...
うそ ||| been lying ||| (1) ||| () (0) ||| 0.0357143 0.04240...
うそ ||| false story ||| (0,1) ||| (0) (0) ||| 0.5 0.0203773...
うそ ||| false ||| (0) ||| (0) ||| 0.0487805 0.0388889 0.067...
うそ ||| had been lying ||| (2) ||| () () (0) ||| 0.0357143 ...
うそ ||| lie ||| (0) ||| (0) ||| 0.0229885 0.154989 0.066667...
うそ ||| lied ||| (0) ||| (0) ||| 0.0357143 0.139535 0.03333...
うそ ||| lies ||| (0) ||| (0) ||| 0.04 0.075 0.0666667 0.063...
うそ ||| lying ||| (0) ||| (0) ||| 0.0523256 0.0424028 0.3 0...
うそ ||| tell lies ||| (0,1) ||| (0) (0) ||| 0.2 0.0428106 0...
うそ ||| was lying ||| (1) ||| () (0) ||| 0.0416667 0.042403...
```

Figure 2.2. Moses phrase table entries for *uso*. Entries in **bold** were converted into transfer rules.

possible to acquire additional alignments from a corpus that can be converted into transfer rules. The acquisition algorithm is as follows:

1. Parse source
2. Partial transfer
3. Parse reference
4. Eliminate shared MRS relations
5. Create rules from remaining relations

Consider this pair of sentences from the BTEC Corpus:

(14) エアコン の　　スイッチ は どこ かしら　。
eakon no　　suicchi-ha　doko kashira .
air conditioner-GEN switch-TOP where　　?
Where's the switch for the air conditioner?

Rule type	Rules	Examples
Adj→Verb	3	痛い → to hurt
Verb→Adj	7	ぼやけている → blurry
Intransitive Verb	29	帰国する → to leave
Transitive Verb	36	愛す → to love
Adjective/Adverb	47	可愛い → cute
Noun	48	動物園 → zoo
Total	170	

Table 2.3. Results of transfer rule acquisition from the BTEC Corpus

Sentence (14) has the following simplified parse:

```
{ PLACE, PROPOSITION_M, UDEF, WHQ, cop_id, _eakon_n_1,
  noun-relation, _suicchi_s_1 }
```

After partial transfer, **_eakon_n_1** has been flagged as an unknown Japanese relation, and **_suicchi_s_1** has been replaced with the English relation **_switch_n_1**:

```
{ DEF_Q, PLACE_N, PRPSTN_M, UDEF_Q, UNSPEC_LOC, WHICH_Q,
  ja:_eakon_n_1, _switch_n_1 }
```

Sentence (1b)'s parse has much in common with that of (1a):

```
{ _FOR_P, INT_M, PLACE_N, PRPSTN_M,_THE_Q, UNSPEC_LOC, WHICH_Q,
  _air+conditioner_n_1, _switch_n_1 }
```

When shared relations, or relations of the same type are eliminated, what remains is:

ja:_eakon_n_1 → **_air+conditioner_n_1**

This can be turned into a Noun rule using the templates introduced in the previous section.

Using this algorithm we have acquired over 170 rules from the IWSLT 2006 corpus. A summary of the rules acquired and their types is given in Table 2.3. Many of these rules represent gaps in JMdict knowledge, however, we are also acquiring some rules for alignments that are incompatible with our current rule types. Examples of

Parsing	28,257	/	42,699	66.18%
Transfer	9,903	/	28,257	33.05%
Generation	7,105	/	9,903	71.75%
End-to-end	**7,105**	/	**42,699**	**16.64**%

Table 2.4. Coverage for JaEn on the BTEC Corpus using all rules

Rules used	Translations	/	Sentences	Coverage
Hand-crafted	215	/	16,479	1.30%
+Dictionary	2,006	/	16,479	12.17%
+Bootstrapping	2,727	/	16,479	16.55%

Table 2.5. Coverage for JaEn on a portion of the BTEC Corpus comparing rule sources

this include *hontou* →"truly", *wakatta* →"okay", and rules spanning multiple semantic relations like *tsunagaranai* →"disconnected." Because these rules directly target areas where JaEn's coverage is lacking they are already having a significant impact on translation coverage and quality.

8 JaEn Evaluation

In this section, we evaluate JaEn by analyzing its coverage over the BTEC and Tanaka Corpora and conducting automatic and human evaluation comparing it to a baseline Moses phrasal SMT system.

8.1 BTEC Corpus Evaluation

We tracked JaEn's coverage on the BTEC Corpus data used as the training set in the IWSLT 2006 evaluation campaign using the rules we acquired and handcrafted (§6.1). Coverage results are summarized in Tables 2.4 and 2.5. We split all translation pairs into individual sentences by tokenizing on sentence ending punctuation such as "." and "?" yielding a slightly different number of translation sentences than reported in IWSLT 2006's data.

Data	Parsing	Transfer	Generation	End-to-End
dev	$\frac{3,599}{4,500}$ (79.98%)	$\frac{1,711}{3,599}$ (47.54%)	$\frac{937}{1,711}$ (54.76%)	$\frac{937}{4,500}$ (20.82%)
test	$\frac{3,500}{4,499}$ (77.80%)	$\frac{1,658}{3,500}$ (47.37%)	$\frac{871}{1,658}$ (52.53%)	$\frac{871}{4,499}$ (19.36%)

Table 2.6. Coverage of JaEn on the Tanaka Corpus

Currently, we have increased our system's coverage over ten-fold from a starting point of 1.3% up to over 16.5%. We see that the transfer rules automatically acquired from JMdict make the greatest contribution to coverage, over 11%, but the rules acquired by bootstrapping (§7.5) also make a meaningful contribution. With our current system, we are able to translate a large number of sentences with interesting linguistic phenomena. Our system's bottleneck is still the transfer stage which succeeds approximately one-third of the time in comparison to the over two-thirds success rate of parsing and over 71% of generation.

We report a BLEU score of 0.22 for the sentences we translate over the entire 44,000 sentence pair training set using a 4-gram cumulative BLEU score with one reference translation.

8.2 Tanaka Corpus Evaluation

We have been developing JaEn targeting the Tanaka Corpus for approximately one year, and in that time we have increased our end-to-end coverage on the development data from 8.5% to over 20% while maintaining a consistent level of translation quality. Full statistics for parsing, transfer, and generation coverage are given in Table 2.6.

We can currently parse approximately 80% of the entries in the Tanaka corpus' development and test sets. Our transfer coverage is over 47.5%, and that of generation is greater than 52%.

In comparison to statistical machine translation systems, this level of coverage is low. Nevertheless, we have constructed a firm foundation for future research, with limited resources[5].

[5] JaEnwas developed with approximately three person years of development time.

8.3 Moses: an SMT Baseline

We evaluate JaEn by comparing its translation coverage and quality to an SMT system baseline. For the baseline we use Moses (Koehn et al., 2007), an open source statistical machine translation system that is the result of collaboration at the 2006 John Hopkins University Workshop on Machine Translation. The main component is a beam-search decoder, but it also includes a suite of scripts that, when used together with GIZA++ and SRILM (Stolcke, 2002), make it possible to learn factored phrase-based translation models and carry out end-to-end translation.

We followed the instructions for creating a basic phrase-based factorless system on the Moses homepage[6]. In order to construct a state-of-the-art SMT fallback, we replicate the system used in the ACL 2007 Second Workshop on Statistical Machine Translation: a factorless Moses system with a 5-gram language model. We use external morphological analyzers to tokenize our data instead of using the provided scripts. We use the Tree Tagger (Schmid, 1994) for English and MeCab (Kudo et al., 2004) for Japanese. Part-of-speech information was discarded after tokenization.

In addition to using Moses as a fallback and a point of comparison, we use the phrase table it produces when we build our lexicon (§7.4).

8.4 Automatic Evaluation of JaEn and Moses

We compared JaEn and Moses using several automatic evaluation methods: BLEU, NEVA, NIST, and METEOR. BLEU scores were calculated using the `multi-bleu.perl` implementation distributed with Moses.

NEVA (Forsbom, 2003) is an alternative to BLEU that is designed to provide a more meaningful sentence-level score for short references. It is calculated identically to BLEU, but leaving out the log and exponent calculations. Both the NEVA and NIST scores were produced using LOGON implementations.

METEOR (Banerjee and Lavie, 2005) is an advanced MT evaluation metric that uses stemming and WordNet synonym matching to relax the matching constraints for English n-gram matches to achieve higher levels of correlation to human judgement than possible with simpler metrics like BLEU and NIST. We calculated all METEOR scores with WordNet stemming and synonym matching enabled.

[6] `http://www.statmt.org/wmt07/baseline.html`

Evaluation	System	BLEU	NEVA	NIST	METEOR
uc+detok	Moses	23.85	34.61	6.10	57.99
uc+detok	JaEn	7.35	18.26	3.89	45.43
lc+tok	Moses	30.56	34.82	6.19	58.22
lc+tok	JaEn	11.24	18.36	3.93	45.57
uc+detok: cased, detokenized			**lc+tok**: uncased, tokenized		

Table 2.7. Automatic evaluation of JaEn and Moses

The automatic evaluation results are given in Table 2.7. We conducted evaluation of standard, written English with capitalization and punctuation (**uc+tok**), as well as caseless English with punctuation tokenized (**lc+detok**). As expected, **lc+detok** has higher overall scores.

Currently, JaEn is outperformed by Moses in every metric we measured. We theorize that there are several reasons for this. JaEn draws the majority of its lexical translation knowledge from a bilingual dictionary. This often results in translations, that while semantically accurate, are not found in the corpus used for evaluation. Since the SMT system is trained on data from the same corpus used for evaluation, it is much less likely to produce translations not found in the evaluation corpus. This effect is magnified by the fact that the evaluation corpus only has one reference translation for each sentence, decreasing the likelihood that alternative lexical choices will be highly scored. The comparatively smaller gap in METEOR scores for JaEn and Moses supports this theory.

In addition to the domain problem, JaEn also suffers from poor ranking models: the lexical probabilities and generation models being used were trained on a Norwegian-English corpus. While we have been able to improve ranking somewhat by manually tweaking the weights of the various components as described in Section 5.5, developing models for the Tanaka Corpus and empirically setting their weights remains an important area of future work.

	Moses			JaEn		
Evaluators	Fluency	Adequacy	Best	Fluency	Adequacy	Best
Human 1	2.59	2.37	60.0	1.99	1.94	40.0
Human 2	3.14	3.26	69.0	2.87	2.83	31.0
Average	2.87	2.82	64.5	2.43	2.39	35.5
Variance	±0.28	±0.45	±4.5	±0.44	±0.45	±4.5

Table 2.8. Human evaluation of Moses and JaEn

8.5 Human Evaluation of JaEn and Moses

We conducted a small-scale human evaluation of the quality of JaEnand Moses. 100 sentences were randomly selected from the cross-section of the Tanaka Corpus test data that both JaEn and Moses could translate.

Two evaluators were shown the Japanese source sentence, its accompanying English reference, and the output from the two systems labeled *System A* and *System B*. The labels were randomly selected for each sentence; neither the developers nor the evaluators knew which system produced a given output until the evaluation was concluded.

The human evaluators assigned a score of 0 (bad) - 4 (good) for the fluency and adequacy of each system's translation output. This evaluation procedure is described in more detail in the IWSLT 2006 overview (Paul, 2006). The evaluators also selected a *best* system for each translation.

The results of the human evaluation are summarized in Table 2.8. While JaEn was ranked lower than Moses in fluency and adequacy, the difference was often less than the variance in inter-annotator agreement.

9 Discussion

While our currently level of coverage with JaEn makes a quantitative comparison with Moses uninformative, we give a qualitative comparison of the two systems in (15–19). This small selection of sample translations illustrates the strengths and weaknesses of each of the systems.

(15) その 歌を 聴いて 私は 子供 のころを 思い 出した 。
sono uta-wo kii-te watashi-ha kodomo no koro-wo omoi-dashi-ta .
that song-ACC heard I-TOP child time-ACC remembered.

The song called up my childhood. (Reference)

I heard the song when I was a child. (Moses)

It heard that song, and I remembered the child time. (JaEn)

(16) 私は いやいや その 仕事を した 。
watashi-ha iyaiya sono shigoto-wo shi-ta .
I-TOP unwillingly that work-ACC did .

I did the work against my will. (Reference)

I did the work against his will. (Moses)

I did that work unwillingly. (JaEn)

(17) リストに 彼女 の 名前が なかった 。
risuto-ni kanojo no namae-ga na-kat-ta .
list-DAT her name-ACC is-NEG-PAST .

His name didn't appear on the list. (Reference)

Her name on the list. (Moses)

There was not any name of hers on the list. (JaEn)

Translations (15–17) are typical examples where JaEn does better than Moses. In (15) and (16) Moses produces fluent idiomatic output, but the meaning is not preserved. In (15), the song reminds the listener of the childhood, it was not necessarily heard when they were a child. In (16) the meaning is made quite different - it was the speaker who didn't want to do the work, not some third person. JaEn avoids using pronouns when it can't get the referent correct. An alternative would have been to co-index the pronoun with the subject (and thus produce *X did X against X's will*). (17) given an example of an error in the reference translation. The Japanese source sentence contains the female possessive pronoun instead of the male given in the English reference sentence. Both JaEn's and Moses correctly produce *her* in their translations, however, Moses loses the sentence's negation. JaEn's translation could be made more fluent, but it at least preserves the meaning.

(18) 来年、 私は 外国に 留学したい 。
rainen, watashi-ha gaikoku-ni ryuugaku shi-tai .
next year, I-TOP foreign country-DAT study-abroad want-to .

I want to study abroad next year. (Reference)

Next year, I want to study abroad. (Moses)

I want to study next year in a foreign country. (JaEn)

(19) うまく 言葉が 出てこない 。
umaku kotoba-ga de-te ko-nai .
well words-NOM leave come-NEG .

Words fail me. (Reference)

Words well. (Moses)

Words have been aptly not leaving. (JaEn)

Translations (18) and (19) show typical examples where JaEn does not do so well. In (18) the meaning is clear, but the best translation (`gakoku-ni--abroad` = ⟨外国に *gaikoku ni* "in overseas" → *abroad*⟩) was not acquired from any of the lexicons or corpora we used. In fact, this sentence contains redundant information: 留学 *ryuugaku* "study abroad" contains the meaning *abroad* on its own, so we really need a rule that will take 外国に留学する *gaikoku-ni ryuugaku suru* and produce *study abroad* as Moses does.

Finally (19) shows an example where JaEn's lexical choice fails. 出てくる *dete-kuru* on its own can mean "leave", but combined with 言葉 *kotoba* "word", it means "utter/speak", and is more common in the negative. To acquire patterns like this, we need to learn more complex rules from the corpus.

We feel that the strengths and weaknesses of these two translation systems complement each other; JaEn does a better job at preserving the structure of sentence, where Moses is more capable at picking up idiomatic, non-compositional translations.

JaEn as it currently stands is an interesting research system, but not yet a useful production system. However, we feel it has the potential to improve upon the translation quality of a phrase-based system, at least for those sentences it can translate. We therefore feel future work should focus on improving quality, rather than coverage - we can always fall back to the robust SMT system. This means an emphasis on not just acquiring knowledge, but also making sure that new and existing rules produce only improvements. One way to do this would be to incorporate feedback cleaning (Imamura et al., 2003).

JaEn is being successfully used in teaching machine translation - the system allows all the intermediate steps to be seen, and also gives a platform for students to

experiment with. JaEn builds on a great deal of previous work, using the results of the DELPH-IN group, JMdict, Moses. This allowed an interesting system to be built with only around three person year's of work.

10 Future Work

In addition to continuing to improve the quality of the system by expanding the inventory of rules, and providing feedback to the component grammars, learning rules directly from examples is an area of future work. One potential strategy is to parse both the source and target language sentences, then transfer the source and attempt to align the translation with the parse of the reference translation. Aligned MRS structures would then be learned as rules.

A similar approach has been taken by (Jellinghaus, 2007). The main differences are that they only align very similar sentences; always start the alignment from the root (the handle of the MRS); and directly align the source and target MRSes.

Another area of future work is translation ranking. Our current method relies on JaEn's statistical models to select the best translation, however, our current models often produce unsatisfiable results. We plan to explore methods of directly applying Moses' statistical models to rank system output regardless of its origin, or more general reranking methods.

Chapter 3

Paraphrasing for Statistical Machine Translation

1 Introduction

Large amounts of training data are essential for training statistical machine translation systems. In this chapter we showed how training data can be expanded by paraphrasing one side of a parallel corpus. The new data is made by parsing then generating using an open-source, precise HPSG-based grammar. This gives sentences with the same meaning, but with minor variations in lexical choice and word order. In experiments paraphrasing the English in the Tanaka Corpus, a freely-available Japanese-English parallel corpus, we showed consistent, statistically-significant gains on training data sets ranging from 10,000 to 147,000 sentence pairs in size as evaluated by the BLEU and METEOR automatic evaluation metrics.

Data-driven machine translation systems such as EBMT and SMT learn how to translate by analyzing aligned bilingual corpora. In general, the more data available the higher the quality of the translation. Unfortunately, there are limits to how much bilingual data exists. In this chapter, we propose a method for increasing the amount of parallel text available for training by using a precise, wide-coverage grammar to paraphrase the text in one language.

The novelty in this work is that we are using a hand-crafted grammar to produce the paraphrases, thus adding a completely new source of knowledge to this system. The paraphrases are both meaning-preserving and grammatical, and thus are

quite restricted. Possible changes include: changes in word order (*Kim sometimes goes* ≡ *Kim goes sometimes*), lexical substitution (*everyone* ≡ *everybody*), contractions (*going to* ≡ *gonna*) and a limited number of corrections (*the the* → *the*). We give an example of paraphrasing in (20). The grammar treats all of these sentences as semantically equivalent.

(20) この こと から 、会社 に は 事故 の 責任 が 無い こと に なる 。

a. It follows from this that the company is not responsible for the accident. (= original)
b. It follows that the company isn't responsible for the accident from this.
c. It follows that the company is not responsible for the accident from this.
d. That the company isn't responsible for the accident follows from this.

We next introduce some related work, then the resources we use in this chapter. This is followed by a description of the method and the evaluation. Finally, we discuss the results and our future research plans.

2 Related Work

Approaches to applying paraphrasing in MT can be roughly classified into the following groups: (1) paraphrasing to expand a machine translation system's coverage, (2) paraphrasing to increase the amount of training or development data, and (3) paraphrasing to increase the similarity between the source and target languages.

In this section, we discuss representative works in each group and compare them with our proposed approach. We limit discussion to *applications* of paraphrasing to machine translation, excluding general discussion of methods of acquiring paraphrases.

2.1 Paraphrasing to Expand Translation Coverage

Callison-Burch et al. (2006) use paraphrases to increase the coverage of unknown source language words in an SMT system. They automatically acquire source language paraphrases from a parallel corpus by using target language phrases as pivots.

An example of this approach is given below. By using the German phrase *unter kontrolle* as a pivot, the English phrase *under control* can be paraphrased as *in check*.

(21) what is more, the relevant cost dynamic is completely under control
im übrigen ist die diesbezügliche kostenentwicklung völlig unter kontrolle

(22) wir sind es den steuerzahlern die kosten schuldig unter kontrolle zu haben
we owe it to the taxpayers to keep the costs in check

New translations are constructed by identifying source language unknown words, finding paraphrases of the unknown words in the system's phrase table, and adding new translation pairs consisting of the unknown word and the translation of its paraphrase. New entries to the phrase table are given the original translation probabilities multiplied by the probability of the source language paraphrases used in their construction. Callison-Burch et al. (2006) showed improvements for sparse datasets for Spanish→English and French→English systems constructed on the Europarl Corpus.

Marton et al. (2009) also use paraphrases to expand an SMT system's phrase table, but they use semantic similarity distribution measures to acquire source language paraphrases from monolingual corpora. They evaluate on Chinese→English and Spanish→English translation tasks, also improving systems trained on sparse datasets, but their approach degrades system performance when trained on 80k or more of data.

Guzmán Herrera and Garrido Luna (2007) take a similar approach and learn new translations for Spanish→English from multi-lingual corpora by using French as a pivot. New translations are added to an existing SMT system's phrase table with probabilities estimated by taking a weighted sum of the combination of paraphrases that produced the new translation. They investigate different weights for the composite paraphrases but do not present evaluations against a baseline system.

The work by Callison-Burch et al. (2006), Marton et al. (2009), and Guzmán Herrera and Garrido Luna (2007) focus on integrating paraphrases acquired from multi- and mono- lingual corpora into an existing SMT system's phrase table with the primary goal of reducing the number of unknown words the system encounters. In order to integrate external paraphrases into an existing phrase table, a measure of translation probability is necessary, and so they develop a series of heuristics to score the artificial alignments produced by pairing paraphrases of a source phrase found in a parallel corpus with the original phrase's alignment in the phrase table.

In contrast, our system produces paraphrases of the sentences in a parallel corpus, and does not alter the phrase table creation process. Rather than learning paraphrases directly from corpora, as in the case of the aforementioned works, our paraphrases are produced from external lexico-syntactic knowledge in the form of an HPSG grammar. English sentences are parsed into a semantic representation that normalizes word order, spelling, and small number of lexical items. Paraphrases are produced from the semantic representation using the same grammar and parser, with paraphrases ranked by a maximum entropy generation model trained on an HPSG treebank. Unlike methods that learn paraphrases directly from corpora, our HPSG paraphrases are limited to grammatical English, eliminating the problem of noisy data.

2.2 Paraphrasing to Increase Translation Data

Nakov (2008) used paraphrases to increase training and parameter tuning data for SMT system. He produced sentence-level paraphrases by using a small set of rules to identify and transform noun phrases in the parallel corpus (e.g., NP_1 *of* $NP_2 \equiv NP_2$ *'s* NP_1).

Some examples from Nakov (2008) are:

(23) of members of the Irish parliament

of Irish parliament members

of Irish parliament's members

(24) action at community level

community level action

These transformations were structural, not lexical, in nature and limited in scope. Nakov (2008) found that noun phrase-based paraphrases were most effective when applied to training data, achieving a BLEU score gain of about 1 point for limited corpus sizes.

Paraphrases have also been used to enrich the data used for parameter tuning in SMT systems. Madnani et al. (2007) obtained English language paraphrases by identifying paraphrases using a pivot language as in Callison-Burch et al. (2006) and produced sentence-level English paraphrases by training an English→English hierarchical SMT system (Chiang, 2005). Experiments showed that paraphrasing the tuning data

used for MERT in a Chinese→English hierarchical SMT system performed competitively with increasing human references. Paraphrasing data for parameter tuning is a promising approach, however, evaluating our paraphrasing method in tuning remains future work.

2.3 Paraphrasing to Increase Linguistic Similarity

Another use of paraphrasing is to increase the similarity between source and target languages in order to facilitate translation. The approaches discussed here can be classified into methods that try to simplify the source language vocabulary and those that reorder the source language into a form closer to the word order of the target language.

Simplifying Source Language Vocabulary

One of the earliest applications of paraphrasing to simplify translation input is shown by the rule-based Japanese→English MT system, **ALT-J/E**. Shirai et al. (1993) simplified untranslatable Japanese input into a "pseudo-source language" that, while ungrammatical, was easier for **ALT-J/E** to parse and translate. Yamamoto (2001) adopted a similar approach with his "Sandglass Paradigm" – normalizing input to a rule-based MT system before expanding it again during the translation phase. Watanabe et al. (2002) also used paraphrases to normalize source language text system by detecting paraphrases in a parallel corpus and replacing them with the most commonly occurring variant. Paraphrases were automatically detected with a dynamic programming algorithm, and the normalized data was used to train an SMT system.

Our approach is similar in spirit to these normalization efforts, however, instead of using simple heuristics or identifying paraphrases in a corpus, we apply an external source of knowledge: a rich, lexical grammar. In addition, instead of directly transforming system input, our approach uses paraphrases to enrich the training data, making it more robust by providing instances of lexical and syntactic variants.

Reordering Source Language Text

Overcoming differences in word order is a challenge for translating highly divergent language pairs like Japanese-English or German-English. Recently there has been

much work on improving SMT by reordering the source language to closer resemble the word order of the target language.

Nießen and Ney (2001) identify differences in question order and long-distance verbal prefix scrambling as phenomena that cause difficulties for German↔English statistical machine translation and used shallow patterns to reorder "harmonize word order" between the German and English. Collins et al. (2005) made use of parses of source sentences to develop a reordering heuristic as well. Komachi et al. (2006) proposed a reordering model that took into account predicate-argument structure in Japanese and followed a heuristic for reordering sentences in the training data as a preprocessing step. The reordering produces sentences that are not grammatical Japanese, however, they are closer in word order to English, facilitating the SMT alignment process. Katz-Brown and Collins (2008) found that for Japanese→English phrasal SMT a naïve reversal of Japanese source language word order outperformed a dependency-based reordering model. Xu and Seneff (2008) use a rule-based parser to parse English and then generate *Zhonglish*[1]: English reordered to resemble Chinese, with some Chinese function words added. The result is then translated using an SMT system.

Our approach also produces variants in word order, however, they are not artificial reorderings to reduce word order differences. Rather, these variants are all valid English as defined by the English HPSG grammar. We make the SMT system's training data more robust and representative of English by providing paraphrases that encapsulate the possible positions of adjuncts, such as adverbial and preposition phrases; relative clauses; and other linguistic phenomena in English with variable word order.

3 Resources

In this section we describe the major resources used. For the SMT system we used the open-source Moses system. For paraphrasing we used the open-source English Resource Grammar. We evaluated on the Tanaka Corpus. We chose the Tanaka corpus primarily because of its unencumbered availability (it is in the public domain), making our results easy to reproduce. In the spirit of open science, we have made the paraphrased Tanaka Corpus data as well as the scripts and Moses settings files necessary to

[1] a term coined by the paper's authors

Tool	Description	Version	Web Page
BLEU Kit	BLEU scores and statistical significance testing	1.03	`www.nlp.mibel.cs.tsukuba.ac.jp/bleu_kit/`
ERG	English paraphrasing	`LinGo (Apr-08)`	`www.delph-in.net/erg/`
GIZA++	Word-level alignments via IBM models	1.0.3	`code.google.com/p/giza-pp/`
LKB	HPSG parser/generator	2008/04/13 14:10:44	`www.delph-in.net/lkb/`
METEOR	MT Evaluation using stemming and synonymy	1.0	`www.cs.cmu.edu/~alavie/METEOR/`
MeCab	Japanese tokenization	0.97	`mecab.sourceforge.net/`
Moses	SMT phrasal translation extraction, decoding	20090831svn	`statmt.org/moses`
NAIST Jdic	Japanese part-of-speech dictionary	0.6.1-20090630	`sourceforge.jp/projects/naist-jdic/`
PET	Unification-based chart parser	v0.99.14svn	`www.delph-in.net/pet/`
SRILM	N-gram language models	1.5.9	`www.speech.sri.com/projects/srilm/`
TreeTagger	English tokenization	3.2	`www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/`

Table 3.1. Tools used for paraphrasing and translation.

	10k		**25k**		**50k**		**75k**		**100k**		**125k**		**147k**	
d.0/f.0	35.65		44.47		47.80		50.13		51.55		53.10		53.67	
d.2/f.2	37.58	(+1.93)	45.10	(+0.63)	48.92	(+1.12)	50.72	(+0.59)	52.12	(+0.57)	52.85	(-0.25)	53.95	(+0.28)
d.4	38.00	(+2.35)	44.69	(+0.22)	48.34	(+0.54)	49.98	(-0.15)	52.19	(+0.64)	53.26	(+0.16)	53.76	(+0.09)
f.4	36.91	(+1.26)	43.56	(-0.91)	47.58	(-0.22)	50.04	(-0.09)	52.22	(+0.67)	52.96	(-0.14)	53.22	(-0.45)
d.6	37.56	(+1.91)	44.43	(-0.04)	48.22	(+0.42)	50.00	(-0.13)	51.91	(+0.36)	53.16	(+0.06)	53.40	(-0.27)
f.6	37.28	(+1.63)	44.62	(+0.15)	48.55	(+0.75)	50.87	(+0.74)	51.05	(-0.50)	53.28	(+0.18)	53.55	(-0.12)
d.8	37.32	(+1.67)	44.05	(-0.42)	48.94	(+1.14)	50.32	(+0.19)	51.74	(+0.19)	52.65	(-0.45)	53.86	(+0.19)
f.8	37.17	(+1.52)	44.66	(+0.19)	48.15	(+0.35)	50.69	(+0.56)	51.60	(+0.05)	53.05	(-0.05)	53.24	(-0.43)
d.10	37.59	(+1.94)	44.05	(-0.42)	48.65	(+0.85)	50.06	(-0.07)	51.90	(+0.35)	52.94	(-0.16)	53.82	(+0.15)
f.10	37.73	(+2.08)	44.46	(-0.01)	47.65	(-0.15)	49.76	(-0.37)	51.81	(+0.26)	53.23	(+0.13)	53.77	(+0.10)

Table 3.2. Japanese→English METEOR scores for training data size vs. paraphrases.

reproduce our experiments available online[2]. A summary of all tools used is given in Table 3.1.

3.1 Moses

Moses (Koehn et al., 2007) is in the words of its creators "a factored phrase-based beam-search decoder for machine translation." It is distributed as open-source software with a collection of utilities that make it easy for users to construct their own SMT system when used with tools for constructing word alignments and language models. For word alignments we used the `giza-pp` branch of GIZA++ (Och and Ney, 2003). To construct language models, we used the SRILM Toolkit (Stolcke, 2002).

3.2 The English Resource Grammar

The LinGO English Resource Grammar (ERG; (Flickinger, 2000)) is a broad-coverage, linguistically precise HPSG-based grammar of English that has been under development at the Center for the Study of Language and Information (CSLI) at Stanford University since 1993. The ERG was originally developed within the Verb*mobil* machine translation effort, but over the past few years has been ported to additional domains and significantly extended. The grammar includes a hand-built lexicon of around 43,000 lexemes. We are using the development release `LinGO (Apr-08)`. Parsing was done with the efficient, unification-based chart parser, PET (Callmeier, 2002), and generation with the Linguistic Knowledge Base (Copestake, 2002). The ERG and associated parsers and generators are freely available from the Deep Linguistic Processing with HPSG Initiative.

For the most part, we use the default settings and the language models trained in the LOGON project both for parsing and generation (Velldal and Oepen, 2006). However, we set the root condition, which controls which sentences are treated as grammatical, to be **robust** for parsing and **strict** for generation. This means that robust rules (e.g. a rule that allows verbs to not agree in number with their subject) will apply in parsing but not in generation. The grammar will thus parse *The dog bark* or *The dog barks* but only generate *The dog barks*.

[2] `http://www3.ntu.edu.sg/home/fcbond/data/`

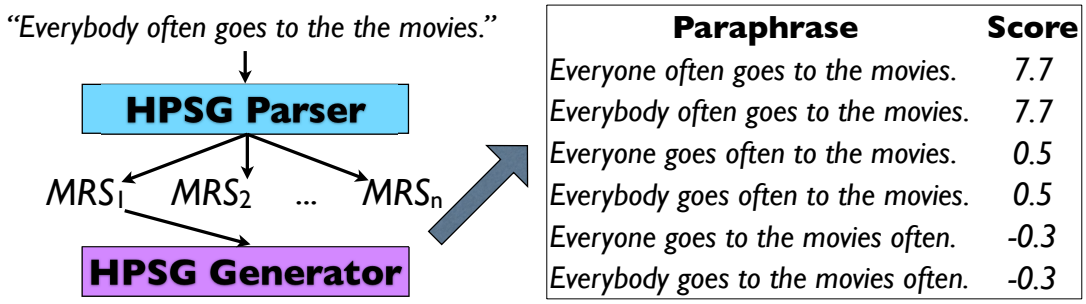


Figure 3.1. Paraphrasing process for "*Everybody often goes to the the movies.*"

3.3 The Tanaka Corpus

The Tanaka corpus is an open corpus of Japanese-English sentence pairs compiled by Professor Yasuhito Tanaka at Hyogo University and his students (Tanaka, 2001) and released into the public domain. For more information on the Tanaka Corpus, see Section 7.1.

The version (2007-04-05) we use has 147,190 sentence pairs in the training split, along with 4,500 sentence pairs reserved for development and 4,500 sentence pairs for testing. After filtering out long sentences (> 40 tokens) as part of the SMT cleaning process, there were 147,007 sentences in the training set. The average number of tokens per sentence is 11.6 for Japanese and 9.1 for English (with the tokenization used in the SMT system).

4 Method

4.1 Paraphrasing

We paraphrase by parsing a sentence to an abstract semantic representation using the English Resource Grammar then generating from the resultant semantic representation using the same grammar. The semantic representation used is Minimal Recursion Semantics (MRS: (Copestake et al., 2005)). We give an example of the paraphrasing process in Figure 3.1 that shows three kinds of paraphrasing. The input sentence is "*Everybody often goes to the the movies.*" It is paraphrased to the MRS shown in Figure 3.2. From that, six sentences are generated. The paraphrased sentences show three

$\langle h_1,$

h_3:person (ARG0 $x_4\{$PERS *3*, NUM *sg*$\}$),
h_5:every_q (ARG0 x_4, RSTR h_6, BODY h_7),
h_8:_often_a_1 (ARG0 $e_9\{$TENSE *untensed*$\}$, ARG1 $e_2\{$TENSE *pres*$\}$),
h_8:_go_v_1 (ARG0 e_2, ARG1 x_4),
h_8:_to_p (ARG0 $e_{10}\{$TENSE *untensed*$\}$, ARG1 e_2, ARG2 x_{11}),
h_{12}:_the_q (ARG0 x_{11}, RSTR h_{14}, BODY h_{13}),
h_{15}:_movie_n_of (ARG0 $x_{11}\{$PERS *3*, NUM *pl*, IND +$\}$, ARG1 $i_{16}\{$SF *prop*$\}$)

$\{ h_6 =_q h_3, h_{14} =_q h_{15} \} \rangle$

Figure 3.2. Semantic representation of "*Everybody often goes to the the movies.*"

changes. Firstly, the erroneous *the the* is corrected to *the*; secondly, *everybody* is optionally paraphrased as *everyone* and finally the adverb *often* appears in three positions (pre-verb, post-verb, post-verb-phrase). We consider the first two to be lexical paraphrases (changes in words) and the latter syntactic paraphrases. Of course, for most sentences there is a combination of lexical and syntactic paraphrases.

"Score" in Figure 3.1 gives a maximum entropy based likelihood estimate to each of the paraphrases. Note that the highest ranked paraphrase is not in this case the original sentence. The paraphrase is quite conservative: sentence initial *often* is not generated, as that is given a different semantics (it is treated as focused). There are no open class paraphrases like *film* $\equiv$ *movie*. Only a handful of closed class words are substituted, typically those that get decomposed semantically, (e.g., *everybody* $\equiv$ *every(x), person(x)*).

We attempted to parse all sentences of the Tanaka Corpus with the ERG and the PET parser. We got one or more well-formed semantic representation for 87.1% of the sentences (the remainder were rejected as ungrammatical). We selected the top ranked representation and attempted to generate from it, this time using the ERG and the LKB generator. We were able to generate one or more realizations for 83.4% of the original sentences. However, many of these gave only one realization and it was identical to the input sentence. Only 53.4% of the sentences had at least one distinct paraphrase; 31.2% had two, 21.2% had three, dropping down to only 1.1% with ten distinct paraphrases.

We show the distribution of paraphrase types over all of the generated paraphrases

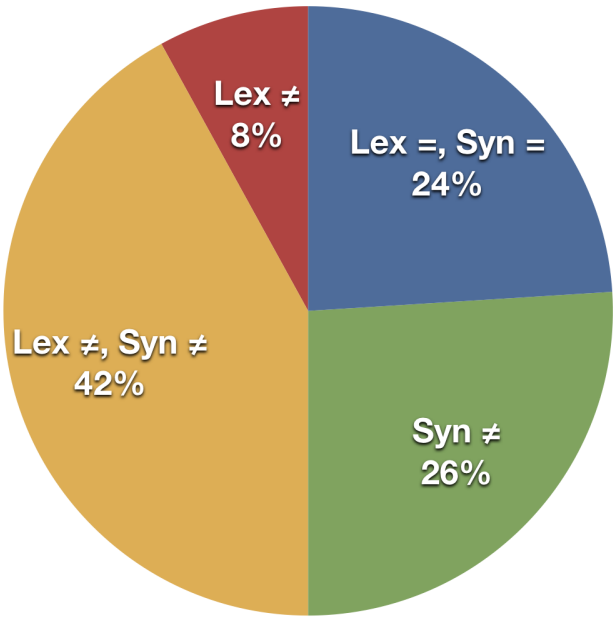


Figure 3.3. Types of paraphrases (Lexical and Syntactic)

in Figure 3.3. Lexical paraphrases are identified by comparing the set of lexical items in the input with those in the output. If they are different, then there is a lexical paraphrase (Lex $\neq$). Syntactic paraphrases are identified by comparing the parse trees. Almost a quarter of the sentences generated are the same as the input (Lex =, Syn =). Most variations include some syntactic paraphrasing (Syn $\neq$: 42%), purely lexical paraphrasing is relatively uncommon (8%).

4.2 Corpus Expansion

Typically when learning translation models, it is assumed that each sentence pair in the parallel corpus is given the same weight. This raises the question of how additional paraphrased data should be weighed. A straight-forward approach would be to simply add each new paraphrase directly to the corpus. However, as sentences can have different numbers of paraphrases, we risk assigning a different weight to each set of

original sentence pair and derived paraphrase. A more sophisticated approach would be to assure that each set maintains the same overall weight by distributing it equally among each paraphrase, or by using the paraphrase generation score from Figure 3.1 to give more likely paraphrases a higher weight. Here, we explore several methods of assigning weights to the paraphrased data by varying the number of times we add each new paraphrase to the parallel corpus.

To make the enhanced training data, we add up to n distinct paraphrases to each unchanged Japanese sentence and original English sentence. We convert all paraphrases to lowercase before checking for uniqueness. If there were m paraphrases, and $n \leq m$ then we just add in the top n ranked paraphrases. If $n > m$ then we produced three test sets:

- (d)istributed: rotate between the original sentence and each paraphrase until the data has been padded out
- (f)irst: after all paraphrases are used, the first (original) sentence is repeated to pad out the data
- (v)arying: add just the paraphrases without padding all entries to the same number of sentences

These variations are shown in the table below. Both (d and f) keep the distribution close to the original corpus. d puts more weight on the paraphrased sentences and f puts more weight on the original sentence. For v the frequency is distorted; some sentences will be repeated many times. For $n \leq 2$, d and f are the same.

d	e_0	e_1	e_2	e_0	e_1
f	e_0	e_1	e_2	e_0	e_0
v	e_0	e_1	e_2		

Paraphrase distributions ($n = 4, m = 2$)

5 Evaluation

In this section, we investigate the effects of supplementing training data with paraphrases on the Tanaka Corpus. We construct phrase-based SMT systems using Moses

for the English→Japanese and Japanese→English language pairs, and evaluate systems on various training corpus sizes.

5.1 Moses Baseline

We replicated the baseline in the ACL 2007 Second Workshop on Statistical Machine Translation. The baseline is a factorless Moses system with a 5-gram language model. We followed the online tutorial[3] as-is, with the exception that we used external morphological analyzers to tokenize our data instead of using the provided scripts. We used the Tree Tagger (Schmid, 1994) for English and MeCab (Kudo et al., 2004) with NAIST Jdic for Japanese. Part-of-speech information was discarded after tokenization.

All data was tokenized, separating punctuation from words and converted to lowercase prior to training and translation. Translations were detokenized and recased prior to evaluation using the helper scripts distributed as part of the baseline system for the ACL 2007 SMT Workshop.

Prior to evaluation we conducted Minimum Error Rate Training on each system using the development data from the target corpus. We used the MERT implementation distributed with Moses. All results reported in this article are post-MERT BLEU scores.

5.2 Data Preparation

In order to measure the effectiveness of our method, we evaluated the Japanese→English and English→Japanese language pairs over the Tanaka Corpus. Because our HPSG parsers perform best on data that is split on the sentence level, wherever possible we split the corpora into the finest possible sentence pairs. We used the following algorithm to split the evaluation data. However, most of the data in the Tanaka Corpus has already been split at the sentence level as part of the JMdict initiative.

- For each sentence pair:
  - split each sentence on sentence-final punctuation (.?!)
  - rejoin split on common English titles (Mr./Ms./Mrs./Dr.)

[3] `http://www.statmt.org/wmt07/baseline.html`

Figure 3.4. Learning curve for English→Japanese *first* paraphrase distribution

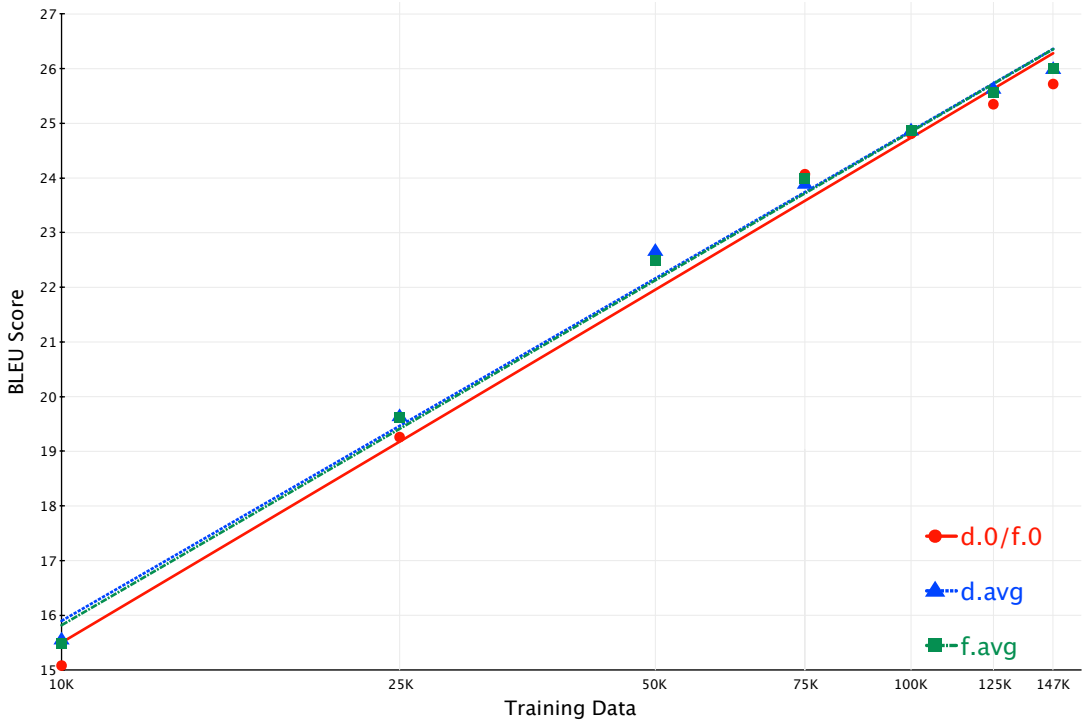
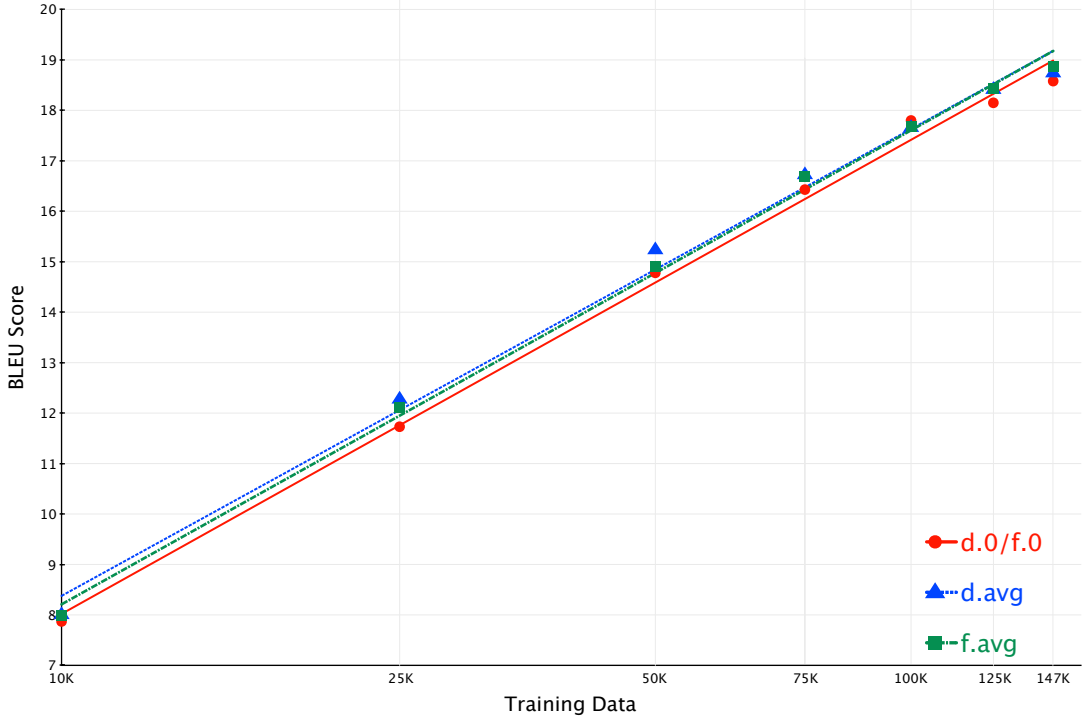


Figure 3.5. Learning curve for Japanese→English *first* paraphrase distribution



– split sentence pairs with same number of source and target sentences into new pairs

– treat sentence pairs with different number of source and target sentences as a single pair

5.3 Results

We evaluated the effects of adding paraphrases to various initial training data sizes using BLEU and METEOR scores. We compared a baseline of no-paraphrases-added ($d.0/f.0$) to systems with progressively larger numbers of new paraphrased sentence pairs added to each training data size. We tested three distributions (d, f and v). v always gave results below the baseline, so we do not report them in more detail. We give several analyses for d and f below.

Learning Curves

We give learning curves in Figures 3.4 and 3.5. The average BLEU scores for *distributed* and *first* paraphrase systems are plotted against training corpus sizes (10k, 25k, 50k, 100k, 125k, and a maximum size of 147k). The training data axis is scaled logarithmically. Best fit lines for the baseline ($d.0/f.0$) and each of the paraphrases show that there is a log-linear relationship in training data size and BLEU score. Paraphrasing almost always outperforms the baseline for small data sets (EJ: 10k-25k, JE: 10k-75k) and large data sets (EJ: 100k-147k, JE: 125k-147k). The region in the middle (EJ: 50k-75k, JE: 100k) is anomalous; the paraphrased averages are below the baseline. We suspect this may be caused by these data sizes containing non-representative samples of data or paraphrases.

BLEU Score

BLEU scores were calculated using the `multi-BLEU.perl` implementation distributed with Moses. We measured the statistical significance of BLEU score differences with the bootstrap methods outlined in Koehn (2004) using Jun-ya Norimatsu's MIT-Licensed BLEU Kit. BLEU scores for d and f are given in Table 3.3, results with an improvement of $p <= 0.05$ over the baseline are shown in **bold**, and the best score

EJ	10k		25k		50k		75k		100k		125k		147k	
d.0/f.0	15.08		19.26		22.58		24.07		24.81		25.35		25.72	
d.2/f.2	**15.42**	**(+0.34)**	**19.75**	**(+0.49)**	22.64	(+0.06)	23.97	(-0.10)	24.97	(+0.16)	25.59	(+0.24)	25.87	(+0.15)
d.4	**15.73**	**(+0.65)**	**19.72**	**(+0.46)**	22.74	(+0.16)	23.87	(-0.20)	24.95	(+0.14)	25.56	(+0.21)	**26.08**	**(+0.36)**
f.4	**15.54**	**(+0.46)**	**19.72**	**(+0.46)**	22.23	(-0.35)	24.22	(+0.15)	24.83	(+0.02)	**25.65**	**(+0.30)**	26.01	(+0.29)
d.6	**15.59**	**(+0.51)**	19.58	(+0.32)	22.72	(+0.14)	23.96	(-0.11)	24.80	(-0.01)	25.54	(+0.19)	25.79	(+0.07)
f.6	**15.66**	**(+0.58)**	19.36	(+0.10)	22.53	(-0.05)	23.97	(-0.10)	24.96	(+0.15)	25.58	(+0.23)	25.91	(+0.19)
d.8	15.38	(+0.30)	**19.57**	**(+0.31)**	22.56	(-0.02)	23.74	(-0.33)	24.70	(-0.11)	**25.83**	**(+0.48)**	**26.13**	**(+0.41)**
f.8	15.31	(+0.23)	**19.65**	**(+0.39)**	22.62	(+0.04)	24.00	(-0.07)	25.13	(+0.32)	25.59	(+0.24)	**26.15**	**(+0.43)**
d.10	**15.64**	**(+0.56)**	19.57	(+0.31)	22.65	(+0.07)	23.91	(-0.16)	24.91	(+0.10)	25.62	(+0.27)	**26.08**	**(+0.36)**
f.10	**15.45**	**(+0.37)**	**19.59**	**(+0.33)**	22.45	(-0.13)	23.81	(-0.26)	24.62	(-0.19)	25.63	(+0.28)	**26.05**	**(+0.33)**
d.avg	15.55	(+0.47)	19.64	(+0.38)	22.66	(+0.08)	23.89	(-0.18)	24.87	(+0.06)	25.63	(+0.28)	25.99	(+0.27)
f.avg	15.48	(+0.40)	19.61	(+0.35)	22.49	(-0.09)	23.99	(-0.08)	24.87	(+0.06)	25.56	(+0.21)	26.00	(+0.28)
JE	**10k**		**25k**		**50k**		**75k**		**100k**		**125k**		**147k**	
d.0/f.0	7.87		11.73		14.78		16.43		17.80		18.15		18.58	
d.2/f.2	8.09	(+0.22)	**12.24**	**(+0.51)**	**15.41**	**(+0.63)**	16.76	(+0.33)	17.46	(-0.34)	18.15	(+0.00)	18.68	(+0.10)
d.4	7.96	(+0.09)	**12.44**	**(+0.71)**	**15.23**	**(+0.45)**	16.76	(+0.33)	17.64	(-0.16)	**18.60**	**(+0.45)**	18.80	(+0.22)
f.4	7.90	(+0.03)	11.98	(+0.25)	15.01	(+0.23)	16.78	(+0.35)	17.64	(-0.16)	**18.52**	**(+0.37)**	18.84	(+0.26)
d.6	7.78	(-0.09)	**12.15**	**(+0.42)**	14.91	(+0.13)	16.70	(+0.27)	17.66	(-0.14)	**18.68**	**(+0.53)**	18.85	(+0.27)
f.6	8.17	(+0.30)	**12.24**	**(+0.51)**	14.83	(+0.05)	**16.95**	**(+0.52)**	17.75	(-0.05)	18.37	(+0.22)	18.90	(+0.32)
d.8	**8.31**	**(+0.44)**	**12.29**	**(+0.56)**	**15.30**	**(+0.52)**	16.70	(+0.27)	17.96	(+0.16)	18.33	(+0.18)	18.82	(+0.24)
f.8	7.66	(-0.21)	**12.22**	**(+0.49)**	14.89	(+0.11)	16.68	(+0.25)	18.06	(+0.26)	**18.57**	**(+0.42)**	**19.04**	**(+0.46)**
d.10	7.91	(+0.04)	**12.29**	**(+0.56)**	**15.36**	**(+0.58)**	16.73	(+0.30)	17.61	(-0.19)	18.34	(+0.19)	18.59	(+0.01)
f.10	8.09	(+0.22)	**12.38**	**(+0.65)**	15.01	(+0.23)	16.60	(+0.17)	17.50	(-0.30)	**18.53**	**(+0.38)**	18.89	(+0.31)
d.avg	8.01	(+0.14)	12.28	(+0.55)	15.24	(+0.46)	16.73	(+0.30)	17.67	(-0.13)	18.42	(+0.27)	18.75	(+0.17)
f.avg	7.98	(+0.11)	12.11	(+0.38)	14.90	(+0.12)	16.69	(+0.26)	17.68	(-0.12)	18.43	(+0.28)	18.87	(+0.29)

Table 3.3. English→Japanese (top) and Japanese→English (bottom) BLEU scores for training data size vs. paraphrases.

for each data size is underlined. We observe a maximum gain of **0.67** BLEU points for English→Japanese at (10k, $d.4$) and a maximum gain of **0.63** for Japanese→English at (50k, $d.2/f.2$). Gains appear to peak at 8 paraphrases; $d.10$ and $f.10$ rarely achieve higher scores that can be achieved with fewer paraphrases. The large number of statistically significant BLEU score improvements reinforce our observations made from the learning curves that paraphrasing is beneficial for small data sets and large data sets. We also notice a trend that small numbers of heavily weighted paraphrases like $d.4$ are more effective for small data sets, while larger numbers of lightly-weighted paraphrases like $f.8$ are more effective for large data sets.

Meteor Score

METEOR (Banerjee and Lavie, 2005) is an advanced MT evaluation metric that uses stemming and WordNet synonym matching to relax constraints for English n-gram matches to achieve higher levels of correlation to human judgement than possible with simpler metrics like BLEU and NIST. We calculated all METEOR scores using version 1.0 with the following options: stemming, WordNet stemming, WordNet synonym matching, and "normalization" – stripping of punctuation and conversion to lower case. METEOR score for Japanese→English for d and f are given in Table 3.2. The METEOR scores do not show as consistent gains as BLEU scores do, but the 10k data set shows great improvements for every paraphrase size. We also note a correlation between statistically significant BLEU score gains and METEOR score improvements; 14/20 paraphrase systems with statistically significant BLEU score gains have increases in METEOR scores.

6 Discussion

Overall, we show consistent, statistically significant improvements on the Tanaka Corpus. Paraphrased SMT systems show statistically significant improvements over the baseline for the majority of the data sizes tested. Furthermore, we observe a log-linear relationship between the size of the system's training data and the BLEU score, with best-fit lines demonstrating the superiority of the paraphrased system over the baseline.

Table 3.4 shows some examples of how translation output changes with the addition of various amounts of paraphrasing data for the Japanese→English language pair.

Data Source	Translation
src	猫 は 台所 から 魚 を 持ち出し た 。
ref	The cat made off with a piece of fish from the kitchen.
d.0/f.0	The cat sprang a kitchen fish.
d.2/f.2/d.4/f.4/d.6/d.8/f.8/d.10/f.10	The cat sprang a fish from the kitchen.
f.6	The cat by the fish from the kitchen.
src	一晩 中 雨 が 降っ たり やん だり し て い た 。
ref	It was raining on and off all night long.
d.0/f.0	During the night it has been raining on and off all day.
d.2/f.2/d.4	All night it has been raining on and off.
f.4/f.8/d.10	During the night it has been raining on and off.
d.6	During the night raining off.
f.6	All night, it has been raining on and off.
d.8	All night it has been raining on.
f.10	All night has been raining off and on.
src	彼 の 話 は 短く て 要点 の つい た もの でし た 。
ref	His speech was brief and to the point.
d.0/f.0/d.2/f.2/f.4/f.6/f.10	His speech was brief and to the.
d.4/d.10	His story of the point.
d.6	His story is made of the point.
d.8	His speech was brief and of.
f.8	His story was short and to the point.
src	ビザ の 延長 を お願い し ます 。
ref	Please extend this visa.
d.0/f.0	Do you accept VISA extension of her hierarchical interpersonal relations, please.
d.2/f.2/d.6/d.8	I would like to extend my stay a visa.
d.4/f.6	I'd like to extend my stay visa.
f.4	I'd like to extend my stay with the visa.
f.8	I'd like to stay a visa.
d.10	I'd like to extend my stay the visa.
f.10	The visa I'd like to extend my stay.

Table 3.4. Example Japanese→English translations from SMT systems trained on 147k of data. The system with the highest BLEU score is underlined.

The example translations contain difficult-to-learn phrasal translations, such as *raining on and off* and *to the point*. As is to be expected from the BLEU scores, the system $f.8$ often gives the best translation. We theorize that the additional data provided by our paraphrases results in better phrasal alignments, which, in turn, improves lexical selection and allows the language model to produce more natural-sounding translations.

Compared to Callison-Burch et al. (2006), Madnani et al. (2007), or Nakov (2008) we are very conservative in our paraphrasing, and this is probably why we get a slightly lower improvement in quality. We could do more extravagant paraphrasing, but would have to retrain our HPSG parser's generation model to effectively rank the new lexical paraphrases. At the moment, it expects fully specified input MRSes. If we were going to allow variation in noun phrase structure or open class lexical variation, then the task could be re-framed as translating between English sentence, and we could build an English→English semantic transfer system to produce richer paraphrases. An example of how to do this (for bilingual transfer of Norwegian→English) is given in Oepen et al. (2007).

Our syntactic reordering is not aimed at matching the target language like Komachi et al. (2006), Xu and Seneff (2008), or Katz-Brown and Collins (2008). We correspondingly get a smaller improvement. On the other hand, because our English paraphrasing method does not depend on a parallel corpus, we expect to get a similar improvement even for different language pairs. Also, our improvement is still there after MERT, whereas the improvement of Komachi et al. (2006) did not make it through the optimization.

7 Further Work

There are three areas in which we think the current use of paraphrasing could be improved: (1) increasing the coverage of the grammar (2) adding new classes of paraphrase rules and (3) improving the integration with the SMT process.

To increase the cover of the paraphrasing, we need to improve the handling of unknown words. Currently, the grammar can parse unknown words (which brings the coverage up to almost 95%), but does not pass enough information to the generator to then generate them. We can overcome this with more powerful hybrid parsing, following Adolphs et al. (2008). A more far-ranging increase would be to paraphrase

the Japanese side as well. We are working on this using Jacy, an HPSG-based Japanese grammar similar to the ERG (Bond et al., 2008) and applying the grammatical error tools of Goodman and Bond (2009) to improve the generation coverage of the Japanese grammar.

Before we increase the types of paraphrases we first need to measure which rules (e.g. lexical or syntactic) have the most effect. We then intend to create English rewriting rules using the MRS transfer machinery from the LOGON project, which is already used in an open source Japanese→English MT system (Bond et al., 2005). For example, we can easily write noun phrase rewriting rules of the type used by Nakov (2008). For lexical substitution we will try using WordNet, after first disambiguating the input.

Finally, we would like to enhance Moses (primarily GIZA++) so that input sentences can be weighted. That way, if we have n paraphrases for one sentence and m for another, each can just be entered with a weight of $1/n$ and $1/m$ respectively. If we could do this, we could then experiment with setting a probability based threshold on the number of paraphrases, for example, to select all paraphrases within β of the probability of the original sentence, according to some language model. In this way we could add only "good" paraphrases, and as many as we deem good for each sentence.

Chapter 4

Ontology Construction

1 Overview

In this chapter, we outline the development of a system that automatically constructs ontologies by extracting knowledge from dictionary definition sentences using Robust Minimal Recursion Semantics (RMRS), a semantic formalism that permits underspecification. Combining deep and shallow parsing resource through the common formalism of RMRS allows us to extract ontological relations in greater quantity and quality than possible with any of the methods independently. Using this method, we construct ontologies from two different Japanese lexicons and one English lexicon. We then link them to existing, handcrafted ontologies, aligning them at the word-sense level. This alignment provides a representative evaluation of the quality of the relations being extracted. We present the results of this ontology construction and discuss how our system was designed to handle multiple lexicons and languages.

2 Background

Automatic methods of ontology acquisition have a long history in the field of natural language processing. The information contained in ontologies is important for a number of tasks, for example word sense disambiguation, question answering and machine translation. In this chapter, we present the results of experiments conducted in automatic ontological acquisition over two languages, English and Japanese, and from

three different machine-readable dictionaries.

Useful semantic relations can be extracted from large corpora using relatively simple patterns (e.g., (Pantel et al., 2004)). While large corpora often contain information not found in lexicons, even a very large corpus may not include all the familiar words of a language, let alone those words occurring in useful patterns (Amano and Kondo, 1999). Therefore it makes sense to also extract data from machine readable dictionaries (MRDs).

There is a great deal of work on the creation of ontologies from machine readable dictionaries (a good summary is Wilkes et al. (1996)), mainly for English. Recently, there has also been interest in Japanese (Tokunaga et al., 2001). Most approaches use either a specialized parser or a set of regular expressions tuned to a particular dictionary, often with hundreds of rules. Agirre et al. (2000) extracted taxonomic relations from a Basque dictionary with high accuracy using Constraint Grammar together with hand-crafted rules. However, such a system is limited to one language, and it has yet to be seen how the rules will scale when deeper semantic relations are extracted. In comparison, as we will demonstrate, our system produces comparable results while the framework is immediately applicable to any language with the resources to produce RMRS. Advances in the state-of-the-art in parsing have made it practical to use deep processing systems that produce rich syntactic and semantic analyses to parse lexicons. This high level of semantic information makes it easy to identify the relations between words that make up an ontology. Such an approach was taken by the MindNet project (Richardson et al., 1998). However, deep parsing systems often suffer from small lexicons and large amounts of parse ambiguity, making it difficult to apply this knowledge broadly.

Our ontology extraction system uses Robust Minimal Recursion Semantics (RMRS), a formalism that provides a high level of detail while, at the same time, allowing for the flexibility of underspecification. RMRS encodes syntactic information in a manner that is general enough to make processing of and extraction from syntactic phenomena including coordination, relative clause analysis and the treatment of argument structure from verbs and verbal nouns. It provides a common format for naming semantic relations, allowing them to be generalized over languages. Because of this, we are able to extend our system to cover new languages that have RMRS resources available with a minimal amount of effort. The underspecification mechanism in RMRS makes

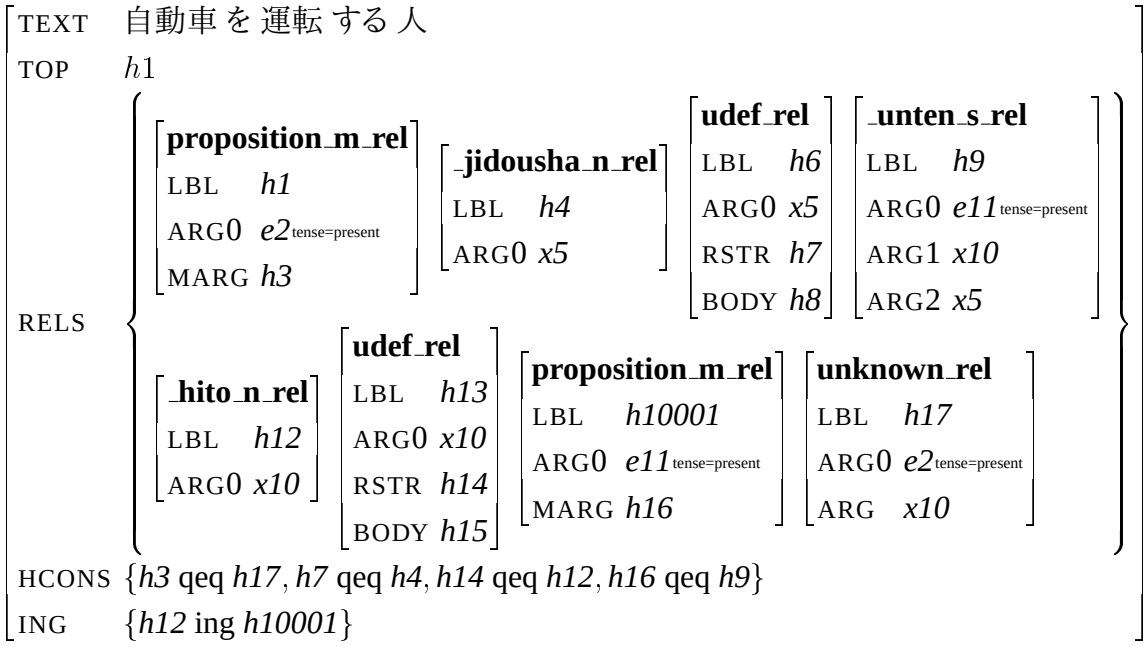


Figure 4.1. RMRS for the Lexeed sense 2 definition of *driver* (Cabocha/JACY)

it possible for us to produce input that is compatible with our system from a variety of different parsers. By selecting parsers of various different levels of robustness and informativeness, we avoid the coverage problem that is classically associated with approaches using deep-processing; using heterogeneous parsing resources maximizes the quality and quantity of ontological relations extracted. Currently, our system uses input from parsers from three levels: with morphological analyzers the shallowest, parsers using Head-driven Phrase Structure Grammars (HPSG) the deepest and dependency parsers providing a middle ground.

Our system was initially developed for one Japanese dictionary (Lexeed). The use of the abstract formalism, RMRS, made it easy to extend to a different Japanese lexicon (Iwanami) and even a lexicon in a different language (GCIDE).

3 Robust Minimal Recursion Semantics

Robust Minimal Recursion Semantics is a form of flat semantics which is designed to allow deep and shallow processing to use a compatible semantic representation,

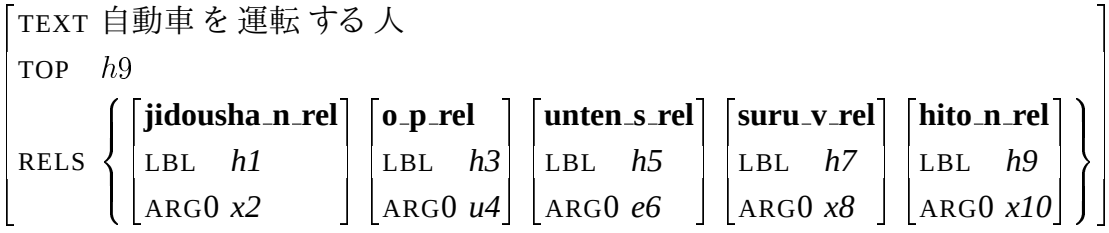


Figure 4.2. RMRS for the Lexeed sense 2 definition of *driver* (ChaSen)

with fine-grained atomic components of semantic content so shallow methods can contribute just what they know, yet with enough expressive power for rich semantic content including generalized quantifiers (Frank, 2004). The architecture of the representation based on Minimal Recursion Semantics (Copestake et al., 2005), including a bag of labeled elementary predicates (EPs) and their arguments, a list of scoping constraints which enable scope underspecification, and a handle that provides a hook into the representation.

The representation can be underspecified in three ways: relationships can be omitted (such as quantifiers, messages, conjunctions and so on); predicate-argument relations can be omitted; and predicate names can be simplified. Predicate names are defined in such a way as to be as compatible (predictable) as possible among different analysis engines, using a lemma_pos_subsense naming convention, where the subsense is optional and the part-of-speech (pos) for coarse-grained sense distinctions is drawn from a small set of general types (**n**oun, **v**erb, **s**ahen (verbal noun), ...). The predicate **`unten_s`**, for example, is less specific than **`unten_s_2`** and thus subsumes it. In order to simplify the combination of different analyses, the EPs are indexed to the corresponding character positions in the original input sentence.

Examples of deep and shallow results for the same sentence 自動車を運転 する人 *jidōsha wo unten suru hito* "a person who drives a car (lit: car-ACC drive do person)" are given in Figures 4.1 and 4.2 (omitting the indexing). Real predicates are prefixed by an under-bar (_). The deep parse gives information about the scope, message types and argument structure, while the shallow parse gives little more than a list of real and grammatical predicates with a hook.

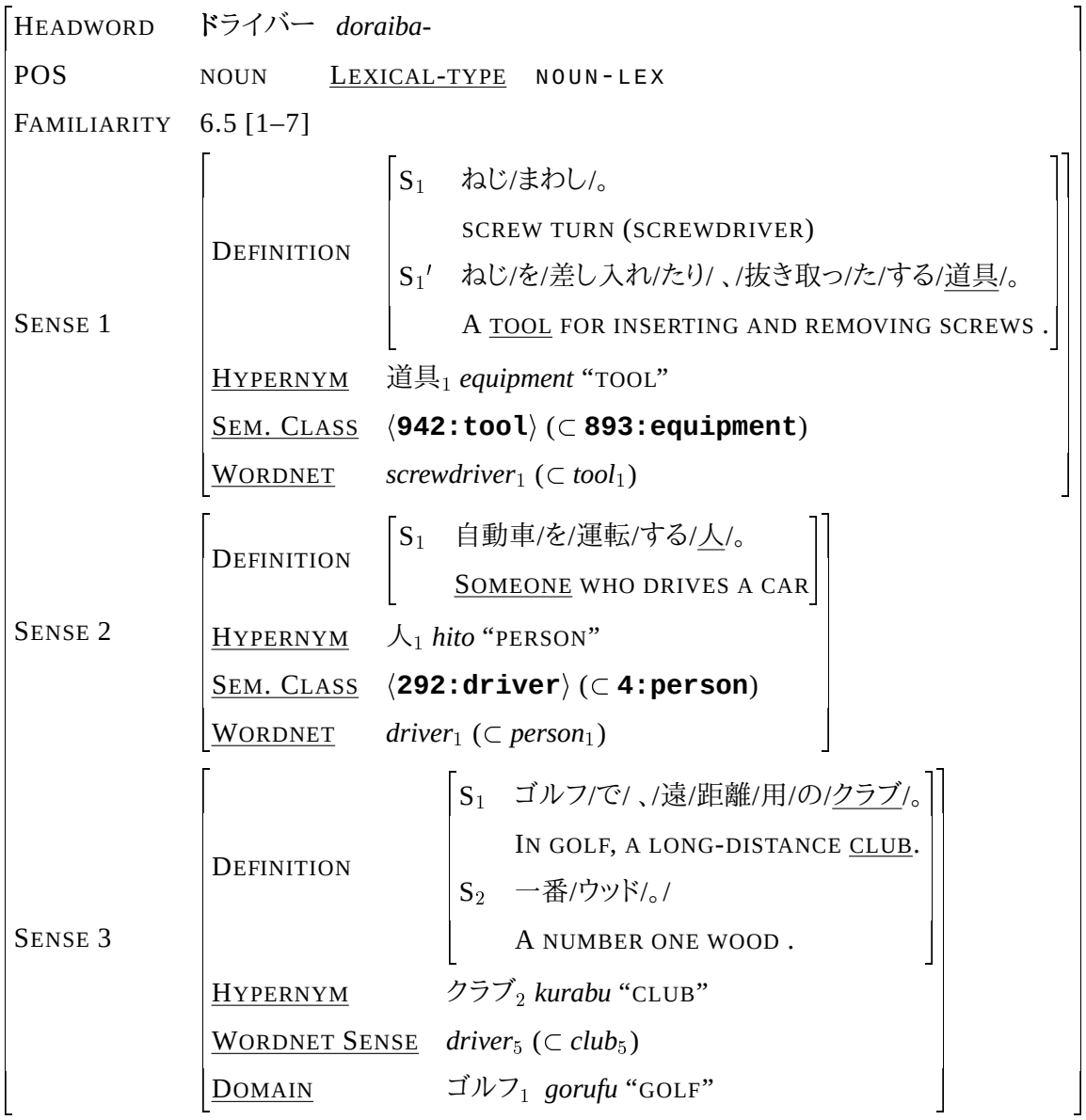


Figure 4.3. Entry for the word *driver* from Lexeed

4 Machine-Readable Dictionaries

4.1 The Lexeed Semantic Database of Japanese

The Lexeed Semantic Database of Japanese is a machine readable dictionary that covers the most familiar open class words in Japanese as measured by a series of psycholinguistic experiments (Kasahara et al., 2004). Lexeed consists of all open class words with a familiarity greater than or equal to five on a scale of one to seven. This gives 28,000 words divided into 46,000 senses and defined with 75,000 definition sentences. All definition sentences and example sentences have been rewritten to use only the 28,000 familiar open class words. The definition and example sentences have been treebanked with the JACY grammar (§ 5.1).

An example entry for the word ドライバー *doraibā* "driver" is given in Figure 4.3, with English glosses added. The underlined material was not in Lexeed originally, we add it in this paper. *doraibā* "driver" has a familiarity of 6.55 and three senses.

4.2 The Iwanami Dictionary of Japanese

The Iwanami Kokugo Jiten (Iwanami) (Nishio et al., 1994) is a concise Japanese dictionary. A machine tractable version was made available by the Real World Computing Project and was used in the SENSEVAL-2 Japanese lexical task (Shirai, 2003). Iwanami has 60,321 headwords and 85,870 word senses. Each sense in the dictionary consists of a sense ID and morphological information (word segmentation, POS tag, base form and reading, all manually post-edited).

4.3 The GNU Contemporary International Dictionary of English

The GNU Collaborative International Dictionary of English (GCIDE) is a freely available dictionary of English based on Webster's Revised Unabridged Dictionary (published in 1913), and supplemented with entries from WordNet and additional submissions from users. It currently contains over 148,000 definitions. The version used in this research is formatted in XML and is available for download[1].

[1] `http://www.ibiblio.org/webster/`

Driver, n. From Drive.

1. One who, or that which, drives; the person or thing that urges or compels anything else to move onward.
2. The person who drives beasts or a carriage; a coachman; a charioteer, etc.; hence, also, one who controls the movements of a any vehicle.
3. An overseer of a gang of slaves or gang of convicts at their work.
4. (Mach.) A part that transmits motion to another part by contact with it, or through an intermediate relatively movable part, as a gear which drives another, or a lever which moves another through a link, etc. Specifically:
   (a) The driving wheel of a locomotive.
   (b) An attachment to a lathe, spindle, or face plate to turn a carrier.
   (c) A crossbar on a grinding mill spindle to drive the upper stone.
5. (Naut.) The after sail in a ship or bark, being a fore-and-aft sail attached to a gaff; a spanker. -Totten.
6. An implement used for driving; as:
   (a) A mallet.
   (b) A tamping iron.
   (c) A cooper's hammer for driving on barrel hoops.
   (d) A wooden-headed golf club with a long shaft, for playing the longest strokes.

Figure 4.4. Example of the word *driver* from the GCIDE

An example of the entry for *driver* is shown in Figure 4.4. There is a typographical error in the definition for sense 2, (*a any*); such minor errors are not uncommon.

We arranged the headwords by frequency and segmented their definition sentences into sub-sentences by tokenizing on semicolons (;). This produced a total of 397,460 pairs of headwords and sub-sentences, for an average of slightly less than four sub-sentences per definition sentence. For corpus data, we selected the first 100,000 definition sub-sentences of the headwords with the highest frequency. This subset of definition sentences contains 12,440 headwords with 36,313 senses, covering approximately 25% of the definition sentences in the GCIDE. The GCIDE has the most polysemy of the lexicons used in this research. It averages over 3 senses per word defined in comparison to Lexeed and Iwanami which both have less than 2.

5 Parsing Resources

We used Robust Minimal Recursion Semantics (RMRS) designed as part of the Deep Thought project (Callmeier et al., 2004) as the formalism for our ontological relation extraction engine. We used deep-processing tools from the Deep Linguistic Processing with HPSG Initiative (DELPH-IN) as well as medium- and shallow-processing tools for Japanese processing (the morphological analyzer ChaSen and the dependency parser Cabocha) from Matsumoto Laboratory.

5.1 Deep Parsers (JACY, ERG and PET)

For both Japanese and English, we used the PET System for the high-efficiency processing of typed feature structures (Callmeier, 2000). For Japanese, we used JACY (Siegel, 2000), for English we used the English Resource Grammar (ERG: (Flickinger, 2000)).[2]

JACY

The JACY grammar is an HPSG-based grammar of Japanese which originates from work done in the Verb*mobil* project (Siegel, 2000) on machine translation of spoken

[2]Both grammars and the PET parser are available at `http://www.delph-in.net/`.

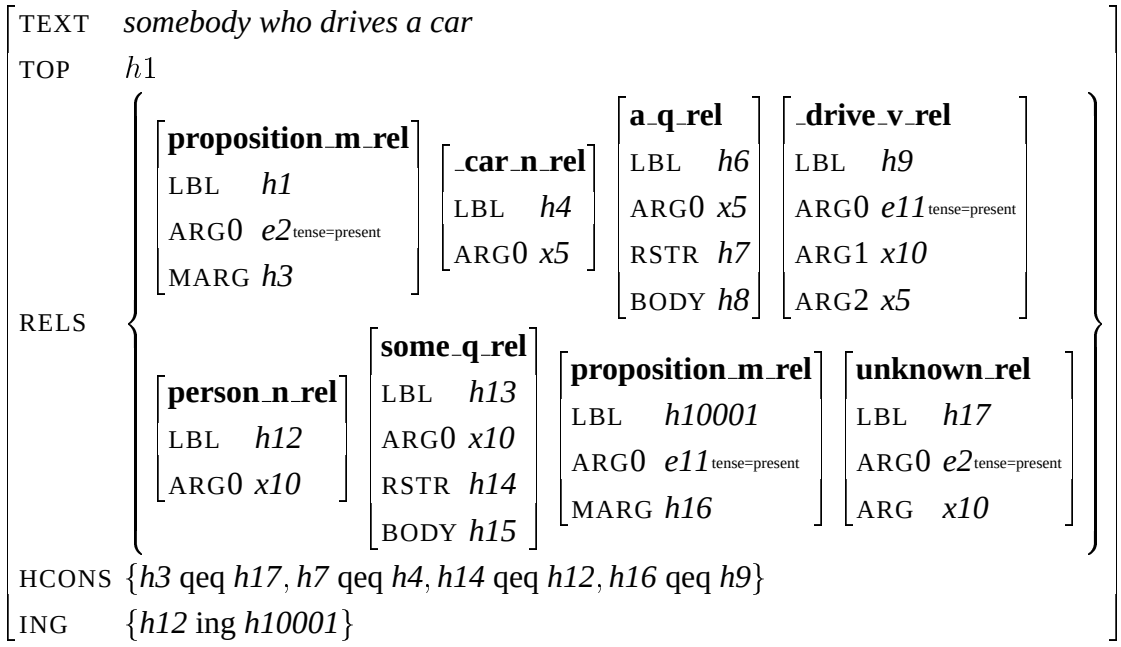


Figure 4.5. RMRS for the GCIDE definition of *driver* (ERG)

dialogues in the domain of travel planning. It has since been extended to accommodate written Japanese and new domains (such as automatic email response and parsing machine readable dictionaries).

The grammar implementation is based on a system of types. There are around 900 lexical types that define the syntactic, semantic and pragmatic properties of the Japanese words, and 188 types that define the properties of phrases and lexical rules. The grammar includes 50 lexical rules for inflectional and derivational morphology and 47 phrase structure rules. The lexicon contains around 36,000 lexemes.

The English Resource Grammar (ERG)

The English Resource Grammar (ERG: (Flickinger, 2000)) is a broad-coverage, linguistically precise grammar of English, developed within the Head-driven Phrase Structure Grammar (HPSG) framework, and designed for both parsing and generation. Originally launched within the Verb*mobil* (Wahlster, 2000) spoken language machine translation project for the particular domains of meeting scheduling and travel plan-

ning, the ERG has since been substantially extended in both grammatical and lexical coverage, reaching 80-90% coverage of sizable corpora in two additional domains: electronic commerce customer email, and the tourism brochures.

The grammar includes a hand-built lexicon of 23,000 lemmas instantiating 850 lexical types, a highly schematic set of 150 grammar rules, and a set of 40 lexical rules, all organized in a rich multiple inheritance hierarchy of some 3000 typed feature structures. Like other DELPH-IN grammars, the ERG can be processed by several parsers and generators, including the LKB (Copestake, 2002) and PET (Callmeier, 2000). Each successful ERG analysis of a sentence or fragment includes a fine-grained semantic representation in MRS.

For the task of parsing the dictionary definitions in GCIDE (the GNU Collaborative International Dictionary of English; see below), the ERG was minimally extended to include two additional fragment rules, for gap-containing VPs and PPs (idiosyncratic to this domain), and additional lexical entries were manually added for all missing words in the alphabetically first 10,000 definition sentences.

These first 10,000 sentences were parsed and then manually tree-banked on the Redwoods (Oepen et al., 2004b) model, to provide the training material for constructing the stochastic model used for best-only parsing of the rest of the definition sentences. Then using POS-based unknown-word guessing for missing lexical entries, PET was used with the ERG to produce a successful parse (and hence an MRS) for about 75% of the first 100,000 definition sentences in GCIDE.

5.2 Medium Parser (Cabocha RMRS)

We produce RMRS from the dependency parser Cabocha (Kudo and Matsumoto, 2002). The method is similar to that of Spreyer and Frank (2005), who produce RMRS from detailed German dependencies. Cabocha provides fairly minimal dependencies: there are three links (dependent, parallel, apposition) and they link base phrases (Japanese *bunsetsu*), marked with the syntactic and semantic head. The Cabocha RMRS parser uses this information, along with heuristics based on the parts-of-speech, to produce underspecified RMRSes. Cabocha RMRS is capable of making use of HPSG resources, including verbal case frames, to further enrich its output. This allows it to produce RMRS that approaches the granularity of the analyses given by HPSG parsers. Indeed, Cabocha RMRS and JACY give identical parses for the example sentence in

Figure 4.1. One of our motivations in including a medium parser in our system is to extract more relations that require special processing; the flexibility of Cabocha RMRS and the RMRS formalism make this possible.

5.3 Shallow Parser (ChaSen RMRS)

The part-of-speech tagger, ChaSen (Matsumoto et al., 2007) was used for shallow processing of Japanese. Predicate names were produced by transliterating the pronunciation field and mapping the part-of-speech codes to the RMRS super types. The part-of-speech codes were also used to judge whether predicates were real or grammatical. Since Japanese is a head-final language, the hook value was set to be the handle of the right-most real predicate. This is easy to do for Japanese, but difficult for English.

6 Ontology Construction

Our approach to ontology construction is to process a definition sentence with shallow and deep parsers and extract ontological relations from the most informative RMRS output. Here, we describe the algorithm used to extract ontological relations from an RMRS structure:

1. let P_i be the number of real predicates in the defining sentence

    - IF $P_i = 1$ (there is a unique real predicate)
      return: $\langle$**synonym**: headword, predicate$\rangle$

2. Initialize a stack of semantic relations to be processed with the semantic relation from the defining sentence's HOOK (the highest scoping handle)

3. Pop a semantic relation from the stack and check it against special relations that require additional processing

    - When a relation indicating coordination or conjunction is found, locate all of its arguments and push them onto the stack for processing
    - IF a special predicate is found, extract its relations and add them to the stack

- ELSE IF the current semantic relation is a real predicate, add it to list of extracted semantic heads

Repeat until stack is empty

4. Filer out all semantic heads whose parts of speech do not match the headword's part of speech

5. Return the ontological relations in the list of extracted semantic heads in the form: ⟨**`relation`**: headword, semantic_head⟩

Step 1 checks for a *synonym* relation, shown by a defining sentence containing a genus term with no differentia. Such a sentence will have a semantic representation with only a single real predicate.

In Step 2, for more complicated defining sentences, we try and find the genus term, looking first at the predicate with the widest scope. This is given by the (R)MRS's HOOK. The default ontological relation for the genus term is *hypernym*.

Step 3 processes each semantic relation in the stack by searching for special relations that require additional processing in order to retrieve the semantic head. Special relations include explicit relation names (such as *ryaku* "abbreviation") and some grammatical predicates. This step identifies and processes special predicates, adding any results to the stack of unprocessed semantic relations. If a relation is not identified as being a special predicate, and it is a non-grammatical predicate, then it is accepted as a semantic head, and it is added to the list of extracted relations. Step 3 is repeated until the stack is empty.

Step 4 filters out headword-semantic head pairs that do not have matching parts of speech. This process is described in Section 6.1.

Step 5 returns the list of all non-grammatical predicates once all semantic heads have been processed for special relations and no new results are produced.

6.1 Special Relations

Occasionally, relations which provide ontological meta-information, such as the specification of domain or temporal expressions, or which help identify the type of ontological relation present are encountered. We refer to these as *special relations*. We follow

Special Relations		Ontological Relation
Japanese	English	
isshu, hitotsu	form, kind, one	**hypernym**
soushou	common name	**hyponym**
ryaku(shou)	abbreviation	**abbreviation**
bubun, ichibu	part, piece	**meronym**
meishou	name	**name**
keishou	polite name	**name:honorific**
zokushou	slang name	**name:slang**

Table 4.1. Special relations and their associated ontological relations

their lead and use a small number of rules to determine where the semantic head is and what ontological relation should be extracted. We give a list of example special relations in Table 4.1. This technique follows in a long tradition of special treatment of certain words that have been shown to be particularly relevant to the task of ontology construction or which are semantically content-free. These words or relations have also be referred to as "empty heads", "function nouns", or "relators" in the literature (Wilkes et al., 1996). Our approach generalizes the treatment of these special relations to rules that are portable for any RMRS (modulo the language specific predicate names) giving it portability that cannot be found in approaches that use regular expressions or specialized parsers.

Special relations also give information about type of ontological relation that has been identified. They can confirm an implicit hypernym such as with *isshu* "one type" in Japanese or identify an entirely different relation, as in the case of the relation *part*, which identifies a meronym relationship in English or *meisho* "honorific name" identifying a name relation in Japanese. Specials predicates can also extract non-ontological relations such as domain.

Augmenting the system to work on English definition sentence simply entailed writing rules to handle special relations that occur in English. Our system currently has 26 rules for Japanese and 50 rules for English. These rules provide processing of relations like those found in Table 4.1, and they also handle processing of coordinate structures, such as noun phrases joined together with conjunctions such as *and*, *or*, and punctuation.

Relation	Lexeed			Iwanami			GCIDE
	Shallow	Medium	Deep	Shallow	Medium	Deep	Deep
hypernym	50,547	45,473	43,319	116,946	117,391	68,590	52,489
synonym	12,780	13,166	9,135	31,838	32,476	18,116	24,421
abbreviation		340	429		1,534	739	
meronym		236	193		398	103	559
name		102	99		271	150	

Table 4.2. Results of ontology extraction

6.2 Filtering by Part-of-Speech

One of the problems we encountered in processing English dictionaries is that many of the definition sentences would have a semantic head with a part-of-speech different than that of the definition word. We found that differing parts-of-speech often indicated an undesirable ontological relation. One reason such relations can be extracted is when a sentence with a non-defining role, for example indicating usage, is encountered. Definition sentence for non-content-bearing words such a *of* or *the* also pose problems for extraction.

We avoid these problems by filtering by parts-of-speech twice in the extraction process. First, we select candidate sentence for extraction by verifying the definition word has a content word POS (i.e. adjective, adverb, noun, or verb). Finally, before we extract any ontological relation, we make sure that the definition word and the semantic head are in compatible POS classes.

While adopting this strategy does reduce the number of total ontological relations that we acquire, it increases their reliability. The addition of a medium parser gives us more RMRS structures to extract from, which helps compensate for any loss in number.

7 Results

The relations acquired in are summarized in Table 4.2. The columns specify source dictionary and parsing method while the rows show the relation type. These counts represent the total number of relations extracted for each source and method combination. The majority of relations extracted are synonyms and hypernyms, however,

some higher-level relations such as meronym and abbreviation are also acquired. It should also be noted that both the medium and deep methods were able to extract a fair number of special relations. In many cases, the medium method even extracted more special relations than the deep method. This is yet another indication of the flexibility of dependency parsing. Altogether, we extracted 105,613 unique relations from Lexeed (for 46,000 senses), 183,927 unique relations from Iwanami (for 85,870 senses), and 65,593 unique relations from GCIDE. As can be expected, a general pattern in our results is that the shallow method extracts the most relations in total followed by the medium method, and finally the deep method.

8 Verification with Hand-crafted Ontologies

Because we are interested in comparing lexical semantics across languages, we compared the extracted ontology with resources in both the same and different languages.

For Japanese we verified our results by comparing the hypernym links to the manually constructed Japanese ontology GoiTaikei. It is a hierarchy of 2,710 semantic classes, defined for over 264,312 nouns (Ikehara et al., 1997). The semantic classes are mostly defined for nouns (and verbal nouns), although there is some information for verbs and adjectives. For English, we compared relations to WordNet 2.0 (Fellbaum, 1998). Comparison for hypernyms done as follows: look up the semantic class or synset C for both the headword (w_i) and genus term(s) (w_g). If at least one of the index word's classes is subsumed by at least one of the genus' classes, then we consider the relationship confirmed (4.1).

$$\exists(c_h, c_g) : \{c_h \subset c_g; c_h \in C(w_h); c_g \in C(w_g)\} \tag{4.1}$$

To test cross-linguistically, we looked up the headwords in a translation lexicon (**ALT-J/E** (Ikehara et al., 1991) and EDICT (Breen, 2004)) and then did the confirmation on the set of translations $c_i \subset C(T(w_i))$. Although looking up the translation adds noise, the additional filter of the relationship triple effectively filters it out again.

The total figures given in Table 4.3 do not match the totals given in Table 4.2. These totals represent the number of relations where both the definition word and semantic head were found in at least one of the ontologies being used in this comparison. By comparing these numbers to the totals given in Section 7, we can get an idea for the

Confirmed relations in Lexeed

Method / Relation	hypernym	synonym	Total
Shallow	58.55 % (16585 / 2328)	61.93 % (5955 / 9615)	59.40 % (22540 / 37943)
Medium	55.97 % (15431 / 27570)	62.61 % (6375 / 10182)	57.76 % (21806 / 37752)
Deep	54.78 % (4954 / 9043)	67.76 % (5098 / 7524)	60.67 % (10052 / 16567)
All	55.22 % (23802 / 43102)	60.46 % (9531 / 15765)	56.62 % (33333 / 58867)

Confirmed relations in Iwanami

Method / Relation	hypernym	synonym	Total
Shallow	61.20 % (35208 / 57533)	63.57 % (11362 / 17872)	61.76 % (46570 / 75405)
Medium	60.69 % (35621 / 58698)	62.86 % (11037 / 17557)	61.19 % (46658 / 76255)
Deep	63.59 % (22936 / 36068)	64.44 % (8395 / 13027)	63.82 % (31331 / 49095)
All	59.36 % (40179 / 67689)	61.66 % (12931 / 20973)	59.90 % (53110 / 88662)

Confirmed relations in GCIDE

POS / Relation	hypernym	synonym	Total
Adjective	2.88 % (37 / 1283)	16.77 % (705 / 4203)	13.53 % (742 / 5486)
Noun	57.60 % (7518 / 13053)	50.71 % (3522 / 6945)	55.21 % (11040 / 19998)
Verb	24.22 % (3006 / 12411)	21.40 % (1695 / 7919)	23.12 % (4701 / 20330)
Total	39.48 % (10561 / 26747)	31.06 % (5922 / 19067)	35.98 % (16483 / 45814)

Table 4.3. Confirmed relations in GoiTaikei and WordNet

coverage of the ontologies being used in comparison. Lexeed has a coverage of approx. 55.74% ($\frac{58,867}{105,613}$), with Iwanami the lowest at 48.20% ($\frac{88,662}{183,927}$), and GCIDE the highest at 69.85% ($\frac{45,814}{65,593}$). It is clear that there are a lot of relations in each lexicon that are not covered by the hand-crafted ontologies. This demonstrates that machine-readable dictionaries are still a valuable resource for constructing ontologies.

8.1 Lexeed

Our results using JACY achieve a confirmation rate of **66.84%** for nouns only and **60.67%** overall (Table 4.3). This is an improvement over Tokunaga et al. (2001), who reported 61.4% for nouns only. We also achieve an impressive 33,333 confirmed relations for a rate of 56.62% overall. It is important to note that our total counts include all unique relations regardless of source. It is interesting to note that shallow processing outperforms medium with 22,540 verified relations (59.40%) compared to 21,806 (57.76%). This would seem to suggest that for the simplest task of retrieving hypernyms and synonyms, information beyond part-of-speech tagging is not necessary. However, since medium and deep parsing obtain relations not covered by shallow parsing and can extract special relations, a task that cannot be performed without syntactic information, it is beneficial to use them as well.

8.2 Human Evaluation

One problem with using existing ontological resources to verify new relations is that only relations which are subsumed by the ontology being used for comparison can be verified. This poses a considerable problem for researchers who wish to extract new relations: be it from domains where such resources are unavailable, or in cases where existing resources are limited in scope, such as for verbs. In this case, it makes more sense to evaluate a selection of the results retrieved by hand than to rely completely on existing ontologies for verification.

In this spirit, we conducted a hand-verification of a selection of the ontological relations acquired from Lexeed using Jacy. 1,471 relations were selected using a stratified method over the entirety of our results (every 35th relationship, ordered by link-type and then headword). In this evaluation we only consider synonyms and any relationships extracted from the first sentence: the second and subsequent definition sentences

tend to contain other non-hypernym information. The results were then evaluated by native speakers of Japanese were given the definition word, the semantic head we retrieved, and the posited relation type and asked to evaluate if the relation was accurate. They had access to the original lexicon.

The human judges found the relations presented to them to be accurate 88.99% of the time. In the 162 relations that were judged unacceptable, it was also determined that a relation did exist in 95 cases, but it was incorrect (i.e. a **`synonym`** in place of a **`hypernym`** and so on). These errors had three sources: the most common was a lack of identified explicit relationships; the next was lack of information from the shallow parse and the last was errors in the argument structure of the deep parse. Tokunaga et al. (2001) report slightly higher results for extracting noun relationships only (91.8%).

8.3 Iwanami

Iwanami's verification results are similar to Lexeed's (Table 4.3). There are on average around 3% more verifications and a total of almost 20,000 more verified relations extracted. It is particularly interesting to note that deep processing performs better here than on Lexeed (63.82% vs 60.67%). Given that a lot of time and effort has gone into optimizing JACY for parsing Lexeed's definition sentence and comparatively little has been spent on Iwanami, it would seem strange to draw the conclusion from this that JACY parses Iwanami's definition sentences better. Rather, we hypothesize that such differences in results may be caused by differences in the information represented in these two lexicons. In particular, less familiar words have fewer senses, and easier to parse definition sentences. These results strongly support our claims that our ontological relation extraction system is easily adaptable to new lexicons.

8.4 GCIDE

At first glance, it would seem that GCIDE has the most disappointing of the verification results with overall verification of not even 36% and only 16,483 relations confirmed. However, on closer inspection one can see that noun hypernyms are a respectable 57.60% with over 55% for all nouns. These figures are comparable with the results we are obtaining with the other lexicons. One should also bear in mind that the definitions found in GCIDE can be archaic; after all this dictionary was first published

in 1913. This could be one cause of parsing errors for ERG. Despite these obstacles, we feel that GCIDE has a lot of potential for ontological acquisition. A dictionary of its size and coverage will most likely contain relations that may not be represented in other sources. One only has to look at the definition of ドライバー "doraibā"/*driver* to confirm this; GoiTaikei has two senses ("screwdriver" and "vehicle operator") Lexeed and Iwanami have 3 senses each (adding "golf club"), and WordNet has 5 (including "software driver"), but GCIDE has 6, not including "software driver" but including *spanker* "a kind of sail". It should be beneficial to propagate these different senses across ontologies.

9 Discussion

We were able to successfully combine deep processing of various levels of depth in order to extract ontological information from lexical resources. We showed that, by using a well defined semantic representation, the extraction can be generalized so much that it can be used on very different dictionaries from different languages. This is an improvement on the common approach to using more and more detailed regular expressions (e.g. Tokunaga et al. (2001)). Although pattern-based methods provide a quick start, the results are not generally reusable. In comparison, the shallower RMRS engines are immediately useful for a variety of other tasks (Callmeier et al., 2004)

However, because the hook is the only syntactic information returned by the shallow parser, ontological relation extraction is essentially performed by this hook-identifying heuristic. While this is sufficient for a large number of sentences, it is not possible to process special relations with the shallow parser since none of the arguments are linked with the predicates to which they belong. Thus, as Table 4.2 shows, our shallow parser is only capable of retrieving hypernyms and synonyms. It is important to extract a variety of semantic relations in order to form a useful ontology. This is one of the reasons why we use a combination of parsers of different analytic levels rather than depending on a single resource.

The other innovation of our approach is the cross-lingual evaluation. As a by-product of the evaluation we enhance the existing resources (such as GoiTaikei or WordNet) by linking them, so that information can be shared between them. In this way we can use the cross-lingual links to fill gaps in the monolingual resources.

GoiTaikei and WordNet both lack complete cover - over half the relations were confirmed with only one resource. This shows that the machine readable dictionary is a useful source of these relations.

Chapter 5

Open Source Natural Language Processing

An important part of our research has been contributing back to the open source projects that made this work possible. We have made every effort to release all of the programs, data, and results so that the NLP community can benefit from them.

1 Ubuntu NLP

Current natural language processing software often has complex dependencies and can be difficult to install and maintain manually. A good example of this is the Moses phrasal SMT system. In order for users to build their own systems, they must download, build, and install not only the Moses decoder and accompanying toolkits, but also the GIZA++ word aligner, and software to create language models. Manual maintenance of locally-installed software also poses problems for NLP research because it is difficult to keep versions of software with complex dependencies in sync across multiple machines.

Ubuntu Linux provides an advanced Debian-based package management system that makes it easy to package and distribute software through internet repositories. Software that requires compilation is compiled once and distributed as a binary saving users the time and effort of doing it on their own. Software dependencies can be specified, making it easy to insure that compatible versions of software are installed together.

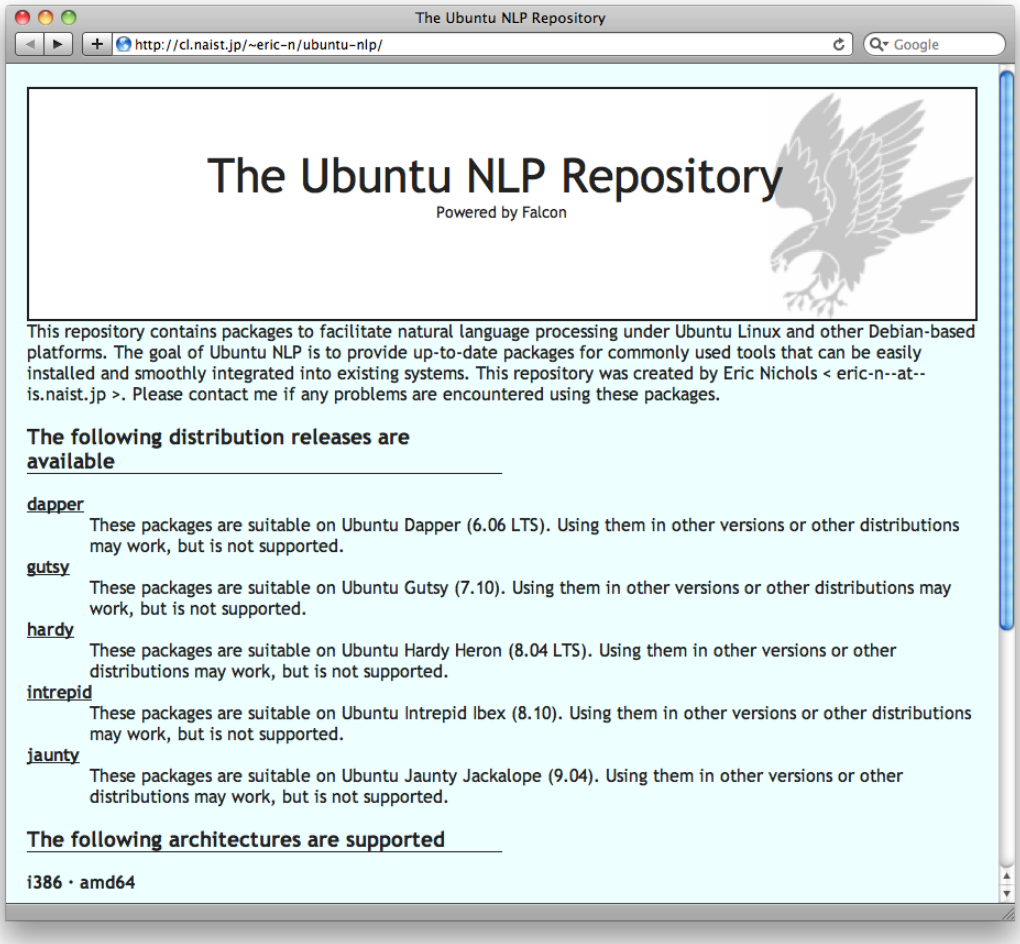


Figure 5.1. The Ubuntu NLP Repository

We started the Ubuntu NLP Repository[1] when we began packaging the software we used in own research, and as it expanded to include many widely-used tools, we made it public. Ubuntu NLP contains packages for popular POS taggers and parsers for English and Japanese, as well as tools for statistical machine translation and tools from the DELPH-IN community. Ubuntu NLP has been well received by the open source software community, and over 5,000 packages were downloaded from Ubuntu NLP per month in 2009. Tables 5.1 and 5.2 give overviews of the packages available

[1] `http://cl.naist.jp/~eric-n/ubuntu-nlp/`

and their popularity.

2 Moses Make

In order to simplify testing and development of Moses SMT systems, we also developed a Makefile-based automation system called *Moses Make*. Users fill out a makefile indicating the location of their training, development, and testing data, and then a simple call of the `make` command will create translation and language models, tune system parameters, and evaluate the system output by producing BLEU, NIST, and METEOR scores.

Moses Make also tokenizes and annotates data with POS, lemma form, and morphology information, providing a flexible framework for the exploration of factored translation models. Currently, Moses Make it supports the English, Italian, Japanese, and Spanish, but it can easily be extended to support any language with a POS tagger and morphological analyzer.

3 DELPH-IN Contributions

Our main contribution to DELPH-IN has been our Japanese→English machine translation system, but we have made several other contributions.

We produced LOGON prototype systems for several other language pairs, including English→Japanese, Korean→Japanese, Spanish→Japanese, Norwegian↔Japanese, and an English→English system for use in paraphrasing.

We made regular contributions to the PET unification parser, providing bug fixes and additional features. We developed a toolset, *Delphin Tools*[2] that simplify the usage of the HPSG framework by automating the parsing, translation, and generation processes. This was distributed along with the paraphrases we made in Chapter 3.

Finally, we have worked to improve the primary resources that we used. We expanded the lexical coverage both JACY and ERG to suit the BTEC and Tanaka corpora, and we released amendments and additions to the Tanaka Corpus and JMdict to the NLP community.

[2] available on Ubuntu NLP

Category	Packages
delph-in	ecl, erg, hog, itsdb, jacy, lkb, logon, pet, utool
english	geniatagger, libwordnet-querydata-perl, morph, python-pywordnet
japanese	cabocha, chasen, crf++, libmecab-perl, libtext-chasen-perl, mecab, mecab-ipadic, mecab-naist-jdic, python-chasen, python-mecab, python-romkan, tinysvm, yamcha
nlp	freeling, giza++, libcfg+, libfries, libomlet, meteor, mgiza++, mkcls, moses, mosesmake, python-nltk, python-nltk-data, python-nltk-doc, srilm, treetagger, treetagger-english, treetagger-italian, treetagger-spanish

Table 5.1. Ubuntu NLP categories and packages

Downloads	Package	Description
10,630	mkcls	Word class clustering for GIZA++
1,370	giza++	GIZA++ word alignment tool
1,204	srilm	SRI Language Model
1,051	moses	Moses SMT decoder
991	geniatagger	The Genia English Part-of-Speech Tagger
953	python-pywordnet	WordNet bindings for Python
893	mecab	The Mecab Japanese Part-of-Speech Tagger
863	python-nltk	The Python Natural Language Toolkit
843	ecl0.9h	Embeddable Common Lisp
801	mgiza++	Multi-threaded GIZA++
755	cabocha	The CaboCha Japanese Dependency Parser
741	treetagger	Tree Tagger: a Multi-lingual Part-of-Speech Tagger
701	cabocha-dic	Dictionary for CaboCha
689	libitsdb	TSDB library
661	pet-cheap	The Cheap Unification Parser
660	mecab-ipadic	IPADIC for Mecab
650	python-nltk-data	Corpora for NLTK
649	libcrf++0	Conditional Random Fields++ library
621	tinysvm	TinySVM: a Support Vector Machines implementation
612	pet-flop	The Flop Unification Grammar Pre-processor

Table 5.2. The top 20 downloaded Ubuntu NLP packages of 2009

Chapter 6

Conclusion

1 Semantic Transfer Based Machine Translation

We greatly expanded the coverage of a Japanese→English semantic transfer-based machine translation system on two corpora using a combination of hand-crafted and automatically acquired transfer rules. Transfer rules we acquired from a bilingual dictionary and directly from parallel corpora via a statistical machine translation system. All of the components in our system are open source: the system itself and all the resources used in it are available for download.

Our system uses a rich semantic representation as a transfer language, allowing the development of powerful transfer rules that produce high-quality translations. By targeting an appropriate corpus for development, automatically acquiring rules from a bilingual dictionary, and hand-crafting transfer rules to handle the most common linguistic phenomenon, we were able to greatly extend our system's coverage.

2 Paraphrasing for SMT

Large amounts of training data are essential for training statistical machine translation systems. We showed how training data can be expanded by paraphrasing one side of a parallel corpus. The new data is made by parsing then generating using an open-source, precise HPSG-based grammar. This gives sentences with the same meaning, but with minor variations in lexical choice and word order. In experiments paraphras-

ing the English in the Tanaka Corpus, a freely-available Japanese-English parallel corpus, we showed consistent, statistically-significant gains on training data sets ranging from 10,000 to 147,000 sentence pairs in size as evaluated by the BLEU and METEOR automatic evaluation metrics.

3 Ontology Construction

We have successfully constructed a large-scale Japanese-English ontology from machine readable dictionaries. Our ontology has been verified both through human evaluation and by comparison to existing ontologies. Research by colleagues has demonstrated that this ontology is useful in improving the stochastic ranking models used in our HPSG parsers (Fujita et al., 2007), and it shows promise for helping to build better translation models for JaEn and to improve alignment, and subsequently, transfer rule quality.

4 Open Source NLP

In an effort to foster open science, throughout our research we have made many contributions to the open source natural language community. We established Ubuntu NLP, a repository of NLP software packaged for the Ubuntu Linux operating system to help improve the access and usability of NLP software. We also made significant contributions to the Moses SMT community and the DELPH-IN deep processing community by providing tools and fixes to their software. Finally, we collaborated with the creators of corpora and dictionaries to enrich these resources for the benefit of the NLP research community.

Appendices

A Transfer Rule Types

A.1 Common Nouns

```
noun_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 #x1 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #x1 ] > ].
```

A.2 Intransitive Verbs

```
arg1_v_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 #e1, ARG1 #x1 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #e1, ARG1 #x1 ] > ].
```

A.3 Transitive Verbs

```
arg12_v_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 #e1, ARG1 #x1, ARG2 #x2 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #e1, ARG1 #x1, ARG2 #x2 ] > ].
```

A.4 Adjectives and Adverbs

```
intersective_attribute_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 #e2, ARG1 #i3 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #e2, ARG1 #i3 ] > ].

adjective_mtr := intersective_attribute_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 #e2, ARG1 #p3 & p ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #e2, ARG1 #p3 ] >,
  FLAGS.SUBSUME < #p3 > ].

relational_adjective_mtr := adjective_mtr &
[ INPUT.RELS  < [ ARG2 #i1 ] >,
  OUTPUT.RELS < [ ARG2 #i1 ] > ].

intersective_adverb_mtr := intersective_attribute_mtr &
```

```
[ INPUT.RELS  < [ LBL #h1, ARG1 #e2 & e ] >,
  OUTPUT.RELS < [ LBL #h1, ARG1 #e2 ] >,
  FLAGS.EQUAL < #e2 > ].

scopal_adverb_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG1 #h2 & h ] >,
  OUTPUT.RELS < [ LBL #h1, ARG1 #h2 ] > ].

intersective_scopal_adverb_mtr := monotonic_mtr &
[ CONTEXT.RELS < [ LBL #h1, ARG0 #e2 & e ] >,
  INPUT  [ RELS < [ LBL #h1, ARG1 #e2 ] >,
           HCONS < qeq & [ HARG #h0, LARG #h1 ] > ],
  OUTPUT [ RELS < [ LBL #h3, ARG1 h & #h4 ] >,
           HCONS < qeq & [ HARG #h0, LARG #h3 ],
                   qeq & [ HARG #h4, LARG #h1 ] > ] ].
```

A.5 Adj+Noun→Adj+Noun

```
adj*n_adj+n_mtr := monotonic_mtr &
[ INPUT.RELS  < [ ARG0 #e2 & e ],
                [ LBL #h3, ARG0 #x4 ],
                [ PRED "unspec_rel",
                  LBL #h3, ARG1 #x4, ARG2 #e2 ] >,
  OUTPUT.RELS < [ LBL #h3, ARG0 #e2, ARG1 #x4 ],
                [ LBL #h3, ARG0 #x4 ] >,
  FLAGS.EQUAL < #e2 > ].
```

A.6 Adj+Noun→Noun

```
adj+n_n_mtr := monotonic_mtr &
[ INPUT.RELS  < [ LBL #h1, ARG0 e_untensed, ARG1 #x2 ],
                [ LBL #h1, ARG0 #x2 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #x2 ] > ].
```

A.7 Noun-Noun Compounds

```
n+n_n+n_mtr := monotonic_mtr &
[ CONTEXT.RELS < [ PRED "unspec_rel",
                   LBL #h3, ARG1 #x4, ARG2 #x2 ] >,
  INPUT.RELS  < [ LBL #h1, ARG0 #x2 ],
                [ LBL #h3, ARG0 #x4 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 #x2 ],
                [ LBL #h3, ARG0 #x4 ] > ].
```

A.8 Noun+Noun→Adj+Noun

```
n+n_adj+n_mtr := monotonic_mtr &
[ INPUT [ RELS < [ LBL #h1, ARG0 #x2 ],
                 [ LBL #h3, ARG0 #x4 ],
                 [ PRED "unspec_rel",
                   LBL #h3, ARG1 #x4, ARG2 #x2 ],
                 [ PRED "udef_q_rel", ARG0 #x2, RSTR #h5 ] >,
          HCONS < qeq & [ HARG #h5, LARG #h1 ] > ],
  OUTPUT.RELS < [ LBL #h3, ARG0 e_untensed, ARG1 #x4 ],
                [ LBL #h3, ARG0 #x4 ] > ].
```

A.9 Noun+Noun→Noun

```
n+n_n_mtr := monotonic_mtr &
[ INPUT [ RELS < [ LBL #h1, ARG0 #x2 ],
                 [ LBL #h3, ARG0 #x4 ],
                 [ PRED "unspec_rel",
                   LBL #h3, ARG1 #x4, ARG2 #x2 ],
                 [ PRED "udef_q_rel", ARG0 #x2, RSTR #h5 ] >,
          HCONS < qeq & [ HARG #h5, LARG #h1 ] > ],
  OUTPUT.RELS < [ LBL #h3, ARG0 #x4 ] > ].
```

A.10 Noun→Adj+Noun

```
n_adj+n_mtr := monotonic_mtr &
```

```
[ INPUT.RELS  < [ LBL #h1, ARG0 #x2 ] >,
  OUTPUT.RELS < [ LBL #h1, ARG0 e_untensed, ARG1 #x2 ],
                [ LBL #h1, ARG0 #x2 ] > ].
```

A.11 Noun→Noun+Noun

```
n_n+n_mtr := monotonic_mtr &
[ INPUT.RELS < [ LBL #h1, ARG0 #x2 ] >,
  OUTPUT [ RELS < [ LBL #h3, ARG0 #x4 ],
                  [ LBL #h1, ARG0 #x2 ],
                  [ PRED compound_rel,
                    LBL #h1, ARG1 #x2, ARG2 #x4 ],
                  [ PRED udef_q_rel,
                    ARG0 #x4 & [ PERS 3, NUM sg, GRIND - ],
                    RSTR #h5 ] >,
           HCONS < qeq & [ HARG #h5, LARG #h3 ] > ] ].
```

B Hand-crafted Transfer Rules

B.1 Requests of Action

```
request_m := monotonic_mtr &
 [ INPUT [ RELS < [ PRED "ja:command_m_rel",
                    LBL #h0, ARG0 #e0, MARG #m4 ],
  [ PRED "ja:_kudasaru_v_2_rel",
                    LBL #h4, ARG0 #e0, ARG3 #h3  ],
  [ PRED "ja:proposition_m_rel",
                    LBL #h3, ARG0 #e1, MARG #m2  ] >,
   HCONS < qeq & [ HARG #m4, LARG #h4 ],
   qeq & [ HARG #m2, LARG #h5 ] > ],
   OUTPUT [ RELS < [ PRED imp_m_rel,
                     LBL #h0, ARG0 #e1 & [ TENSE pres ],
     MARG #m2, PSV #p0, TPC #t0 ],
   [ PRED polite_rel, LBL #h5,
     ARG0 #i & i, ARG1 #e1, CARG "please" ] >,
    HCONS < qeq & [ HARG #m2, LARG #h5 ] > ],
   FLAGS.EQUAL < #i > ].
```

B.2 Requests of Possession

```
kudasai_v_gimme_v := monotonic_mtr &
 [  CONTEXT.HCONS < qeq & [ HARG #hm, LARG #hv ] >,
      INPUT [ RELS < [ PRED "ja:command_m_rel",
                       LBL #h0, ARG0 #e0, MARG #hm ],
  [ PRED "ja:_kudasaru_v_1_rel",
                    LBL #hv, ARG0 #e0,  ARG2 #x2 ] > ],
   OUTPUT [ RELS < [ PRED imp_m_rel,
                     LBL #h0, ARG0 #e0 & [ TENSE pres ],
     MARG #hm, PSV #p0, TPC #t0 ],
   [ PRED polite_rel, LBL #hv,
     ARG0 #i & i, ARG1 #e0, CARG "please" ],
   [ PRED "_give_v_1_rel", LBL #hv, ARG0 #e0,
     ARG1 #x1, ARG2 #x2, ARG3 #x3 ],
```

```
   [ PRED pronoun_q_rel, ARG0 #x1, RSTR #hr1 ],
   [ PRED pron_rel, LBL #hp1, ARG0 #x1 &
    [PERS 2, NUM sg, PRONTYPE zero_pron] ],
   [ PRED pronoun_q_rel, ARG0 #x3, RSTR #hr ],
   [ PRED pron_rel, LBL #hp, ARG0 #x3 &
   [ PERS 1, NUM sg, PRONTYPE std_pron] ]
   >,
    HCONS < qeq & [ HARG #hr1, LARG #hp1 ],
    qeq & [ HARG #hr, LARG #hp ] > ],
   FLAGS.EQUAL < #i > ].
```

B.3 Politeness

```
gozaru_v--exist_v_jf := arg1_v_mtr &
[ INPUT.RELS < [ PRED "ja:_gozaru_v_1_rel" ] >,
 OUTPUT.RELS < [ PRED "ja:_exist_v_rel" ] > ].

iru_v--exist_v_jf := arg1_v_mtr &
[ INPUT.RELS < [ PRED "ja:_iru_v_be_rel" ] >,
 OUTPUT.RELS < [ PRED "ja:_exist_v_rel" ] > ].
```

B.4 Comparatives

```
motto_a_rel-comp_rel-mtr := adjective_mtr &
[ JA.RELS < [ PRED "ja:_motto_a_rel" ] >,
  EN.RELS < [ PRED comp_rel ] > ].

yori_a_rel-comp_rel-mtr := adjective_mtr &
[ JA.RELS < [ PRED "ja:_yori_a_rel" ] >,
  EN.RELS < [ PRED comp_rel ] > ].

mousukoshi_a_rel-a_little+comp_rel-mtr := monotonic_mtr &
 [ JA.RELS < [ PRED "ja:_mousukoshi_a_rel",
               LBL #h, ARG0 #e, ARG1 #e1 ] >,
   EN.RELS < [ PRED comp_rel,  LBL #h, ARG0 #e, ARG1 #e1],
     [ PRED "_a+little_x_deg_rel",
```

```
                LBL #h, ARG0 e, ARG1 #e]> ].
```

B.5 Superlatives

```
ichiban-superlative-mtr := intersective_adverb_mtr &
[ JA.RELS < [ PRED "ja:_ichiban_a_rel", ARG1 #e ] >,
  EN.RELS < [ PRED superl_rel ] > ].

mottomo-superlative-mtr := intersective_adverb_mtr &
[ JA.RELS < [ PRED "ja:_mottomo_a_rel"  ] >,
  EN.RELS < [ PRED superl_rel ] > ].
```

B.6 Verb Modifying Comparatives and Superlatives

```
most_ef := intersective_adverb_mtr &
 [ CONTEXT.RELS <  [ PRED "~_v_", ARG0 #e ] >,
   INPUT.RELS < [ PRED superl_rel, ARG1 #e  ] >,
   OUTPUT.RELS < [ PRED "_most_a_1_rel" ] > ].

more_ef := intersective_adverb_mtr &
 [ CONTEXT.RELS <  [ PRED "~_v_", ARG0 #e ] >,
   INPUT.RELS < [ PRED comp_rel, ARG1 #e  ] >,
   OUTPUT.RELS < [ PRED "_more_a_1_rel" ] > ].

very-a_lot_ef := intersective_adverb_mtr &
 [ CONTEXT.RELS <  [ PRED "~_v_", ARG0 #e ] >,
   INPUT.RELS <    [ PRED "_very_x_deg_rel", ARG1 #e  ] >,
   OUTPUT.RELS <   [ PRED "_a+lot_a_1_rel" ] > ].
```

B.7 Zero Pronoun Insertion

```
zero_arg3_123_ef := optional_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #x1, ARG2 #x2, ARG3 #z3 & u] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1, ARG2 #x2, ARG3 #x3],
```

```
   [ PRED pronoun_q_rel, ARG0 #x3, RSTR #h1 ],
   [ PRED pron_rel, LBL #h2,
     ARG0 #x3 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS.EQUAL < #h, #z3 > ].

zero_arg2_123_ef := optional_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #x1, ARG2 #z2 & u, ARG3 #x3 & i] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1, ARG2 #x2, ARG3 #x3],
   [ PRED pronoun_q_rel, ARG0 #x2, RSTR #h1 ],
   [ PRED pron_rel, LBL #h2,
     ARG0 #x2 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS [EQUAL < #h, #z2 >,
  SUBSUME < #x3 > ] ].

zero_arg1_123_ef := monotonic_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #z1 & u, ARG2 #x2 & i, ARG3 #x3 & i] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1, ARG2 #x2, ARG3 #x3],
   [ PRED pronoun_q_rel, ARG0 #x1, RSTR #h1 ],
   [ PRED pron_rel, LBL #h2,
     ARG0 #x1 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS [EQUAL < #h, #z1 >,
  SUBSUME < #x2, #x3 > ] ].

zero_arg2_12_ef :=  optional_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #x1, ARG2 #z2 & u] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1, ARG2 #x2],
   [ PRED pronoun_q_rel, ARG0 #x2, RSTR #h1 ],
```

```
   [ PRED pron_rel, LBL #h2,
     ARG0 #x2 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS.EQUAL < #h, #z2 >].

zero_arg1_12_ef := monotonic_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #z1 & u, ARG2 #x2 & i] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1, ARG2 #x2],
   [ PRED pronoun_q_rel, ARG0 #x1, RSTR #h1 ],
   [ PRED pron_rel, LBL #h2,
     ARG0 #x1 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS [EQUAL < #h, #z1 >,
  SUBSUME < #x2 > ] ].

zero_arg1_1_ef := monotonic_mtr &
 [ INPUT.RELS < [ PRED #pred, LBL #h, ARG0 #e,
  ARG1 #z1 & u] >,
   OUTPUT [ RELS < [ PRED #pred, LBL #h,
     ARG0 #e, ARG1 #x1],
   [ PRED pronoun_q_rel, ARG0 #x1, RSTR #h1 ],
   [ PRED pron_rel, LBL #h2,
     ARG0 #x1 & [PRONTYPE std_pron] ] >,
    HCONS < qeq & [ HARG #h1, LARG #h2 ] > ],
   FLAGS.EQUAL < #h, #z1 >].
```

C Translation Examples

Given below are the first 25 translations produced by JaEn on the IWSLT 2006 shared task training data and the Tanaka Corpus development data. The system output is completely unedited, and the top 5 ranked translations are listed for each successfully translated sentence. The number to the left of each translation indicates its sentence ID in the data set.

C.1 IWSLT 2006 Corpus

5	**source**	信号 は 赤 でし た 。
	reference	*The light was red.*
	target	Signals were red.
	target	Signal was red.
	target	The signal was red.
	target	The signals were red.
	target	Signals were reds.

10	**source**	重症 です か 。
	reference	*Is it serious?*
	target	Are you serious illness?
	target	Is it serious illness?
	target	Are you serious illnesses?
	target	Is it serious illnesses?
	target	Is it a serious illness?

11	**source**	暗証 番号 を 押し て 下さい 。
	reference	*Please input your pin number.*
	target	Please press code numbers.
	target	Please press the code number.
	target	Press the code number please.
	target	Please press the code numbers.
	target	Press code numbers please.

17	**source**	分かり まし た 。
	reference	*Of course.*
	target	Okay.
	target	Okay?
	target	Okay

18	**source**	九 十 九 ドル です ね 。
	reference	*It was ninety nine dollars, wasn't it?*
	target	It is 99.
	target	I am 99.
	target	He is 99.
	target	She is 99.
	target	They are 99.

23	**source**	抜か ない で ください 。
	reference	*I don't want it extracted.*
	target	Please do not omit it.
	target	Please let's not omit it.
	target	Please don't omit it.
	target	Please do not omit them.
	target	Let's not omit you please.

26	**source**	すり だ 。
	reference	*Pickpocket.*
	target	Thieves.
	target	Thief.
	target	The thief.
	target	A thief.
	target	The thieves.

30	**source**	最寄り の 香水 店 は どこ です か 。
	reference	*Where's the nearest perfumery?*
	target	Where is the nearest perfume store?
	target	Where are the nearest perfume stores?
	target	Where is the most near perfume store?
	target	Where are the most near perfume stores?

40	**source**	今 空港 に い ます 。
	reference	*I'm at the airport right now.*
	target	It is in the airport now.
	target	It is in an airport now.
	target	It is on the airport now.
	target	It is at airports now.
	target	They are in the airport now.

52	**source**	ザリガニ が 欲しい の です が 。
	reference	*I'd like some crayfish.*
	target	I want crayfish.
	target	Crayfish want it.
	target	You want crayfish.
	target	He wants crayfish.
	target	They want crayfish.

55	**source**	これ を フランス フラン に 換え て 下さい 。
	reference	*Into French francs please.*
	target	Substitute it to Furansu franks please.
	target	Please substitute it to Furansu franks.
	target	Please substitute it to Furansu francs.
	target	Substitute this in Furansu franks please.
	target	Substitute it to Furansu francs please.

59	**source**	ライター が 欲しい の です が 。
	reference	*I'd like a lighter.*
	target	I want a writer.
	target	I want writers.
	target	You want a writer.
	target	You want writers.
	target	The writer wants it.

60	**source**	この ホテル に は 会議 施設 が あり ます か 。
	reference	*Does this hotel have conference facilities?*
	target	Is there a convention institution in this hotel?
	target	Is there a convention institution to this hotel?
	target	Are there convention institutions in this hotel?
	target	Are there convention institutions to this hotel?
	target	Are there convention engineers in this hotel?

63	**source**	空港 から 電話 し て い ます 。
	reference	*I'm calling from the airport.*
	target	They are telephoning them from the airport.
	target	They are telephoning him from the airport.
	target	We are telephoning them from the airport.
	target	We are telephoning him from the airport.
	target	They are telephoning her from the airport.

72	**source**	助け て 下さい 。
	reference	*Help me, please.*
	target	Please help me.
	target	Please help us.
	target	Please help her.
	target	Help me please.
	target	Please help you.

73	**source**	救急 車 を 呼ん で 下さい 。
	reference	*Call an ambulance, please.*
	target	Please call emergency cars.
	target	Please call an emergency car.
	target	Please call the emergency car.
	target	Call an emergency car please.
	target	Please call the emergency cars.

83	**source**	住所 を ここ に 書い て 下さい 。
	reference	*Please write down your address here.*
	target	Please write the address here.
	target	Please write residence here.
	target	Please write addresses here.
	target	Write the address here please.
	target	Please write a residence here.

86	**source**	保証 は あり ます か
	reference	*Is there a warranty?*
	target	Is there a security?
	target	Is there security?
	target	Are there securities?

91	**source**	喫茶 室 は どこ です か 。
	reference	*Where is the coffee shop?*
	target	Where is the tea house cellar?
	target	Where is the tea house room?
	target	Where is a tea house cellar?
	target	Where are the tea house rooms?
	target	Where is a tea house room?

92	**source**	子供 用 の セーター が 欲しい の です が 。
	reference	*I'd like a children's sweater.*
	target	I want a child service sweater.
	target	I want a child business sweater.
	target	You want a child service sweater.
	target	I want the child service sweater.
	target	You want a child business sweater.

98	**source**	あれ は 何 です か 。
	reference	*What's that?*
	target	What is it?
	target	What is that?

99	**source**	正装 が 必要 です か 。
	reference	*Do I have to dress up?*
	target	Do you need a uniform?
	target	Do I need a uniform?
	target	Do we need a uniform?
	target	Do you need uniforms?
	target	Do you need the uniform?

103	**source**	着きました。
	reference	*This is it.*
	target	I arrived.
	target	We arrived.
	target	He arrived.
	target	They arrived.
	target	She arrived.

104	**source**	ここで停めて下さい。
	reference	*Please stop here.*
	target	Please stop me here.
	target	Please stop them here.
	target	Please stop him here.
	target	Please stop it here.
	target	Please stop us here.

109	**source**	オートマチックの車が良いです。
	reference	*I prefer an automatic car.*
	target	Ja: O N Tomachikku 1 Rel cars are good.
	target	Ja: O N Tomachikku 1 Rel cars are nice.
	target	The Ja: O N Tomachikku 1 Rel car is good.
	target	The Ja: O N Tomachikku 1 Rel cars are good.
	target	A Ja: O N Tomachikku 1 Rel car is good.

115	**source**	今夜でございますか。
	reference	*Tonight?*
	target	Is it tonight?
	target	Are they tonight?

121	**source**	それ は どんな 話 です か 。
	reference	*What kind of story is it?*
	target	What chat is it?
	target	Which chat is it?
	target	What chat is that?
	target	What chats is it?
	target	Which chat is that?

124	**source**	切手 を 売る 窓口 は どこ です か 。
	reference	*Which window sells stamps?*
	target	Where are the ticket windows, which sell stamps?
	target	Where are the ticket windows, who sell stamps?
	target	Where are the ticket windows, that sell stamps?
	target	Where are ticket windows, who sell stamps?
	target	Where are ticket windows, which sell stamps?

126	**source**	見どころ は あります か 。
	reference	*Are there any special sights?*
	target	Are there highlights?
	target	Is there a highlight?

131	**source**	二 百 ドル です 。
	reference	*Two hundred dollars.*
	target	200.
	target	The 200.

134	**source**	最寄り の 書店 は どこ です か 。
	reference	*Where's the nearest bookshop?*
	target	Where is the nearest bookshop?
	target	Where are the nearest bookshops?
	target	Where is the nearest bookstore?
	target	Where are the nearest bookstores?
	target	Where are the most near bookshops?

141	**source**	赤ワイン を 頂け ます か 。
	reference	*May I have some red wine?*
	target	Do they receive red wine?
	target	Do you receive red wine?
	target	Do you receive a red wine?
	target	Do you receive the red wine?
	target	Do they receive a red wine?

142	**source**	ゲート で 受け取っ て 下さい 。
	reference	*You'll get it at the gate.*
	target	Please let's get it on gates.
	target	Please get it on gates.
	target	Please let's get you on gates.
	target	Please let's get it in gates.
	target	Please let's get them on gates.

156	**source**	彼女 は ビール が 欲しい 。
	reference	*She wants a beer.*
	target	She wants beer.
	target	She wants beers.
	target	She wants a beer.
	target	She wants the beer.
	target	Beer wants her.

157	**source**	手術 が 必要 です 。
	reference	*You need an operation.*
	target	Ja: Shujutsu S 1 Rel is essential.
	target	We need Ja: Shujutsu S 1 Rel.
	target	You need Ja: Shujutsu S 1 Rel.
	target	I need Ja: Shujutsu S 1 Rel.
	target	They need Ja: Shujutsu S 1 Rel.

158	**source**	卵 を もっと 下さい 。
	reference	*More eggs, please.*
	target	Please give me eggs more.
	target	Please give me an egg more.
	target	Please give me the eggs more.
	target	Please give me the egg more.
	target	Please give me roe more.

160	**source**	タクシー です か 。
	reference	*Cabs?*
	target	Are you a taxi?
	target	Is it a taxi?
	target	Is he a taxi?
	target	Is it the taxi?
	target	Is she a taxi?

162	**source**	こちら が 御 部屋 の 鍵 です 。
	reference	*Here's your key, sir.*
	target	We are room keys.
	target	We are a room key.
	target	We are room's keys.
	target	We are room's key.
	target	We are the room key.

168	**source**	ホテル です よ 。
	reference	*This is your hotel.*
	target	Hotel.
	target	Hotels.
	target	The hotel.
	target	A hotel.
	target	The hotels.

179	**source**	遅い です 。
	reference	*It is slow.*
	target	Is slow.
	target	Are slow.
	target	Is slow?
	target	Are slow?
	target	Is slow

185	**source**	ドレッシング は 何 に 致し ましょう か 。
	reference	*What kind of dressing?*
	target	What do you do dressings in?
	target	What do you do dressings to?
	target	What do you do the dressing in?
	target	What do you do the dressing to?
	target	What do you do a dressing in?

188	**source**	メニュー を 見せ て 下さい 。
	reference	*Can we see a menu?*
	target	Please show menus.
	target	Please show the menu.
	target	Please show a menu.
	target	Please show the menus.
	target	Show menus please.

190	**source**	大丈夫 です 。
	reference	*I’m all right.*
	target	Is allright.
	target	Are allright.
	target	Am allright.
	target	Is allright?
	target	Is allright

198	**source**	一番 親しい 人 は 誰 です か 。
	reference	*Who’s your best friend?*
	target	Who is the closest man?
	target	Who is the closest person?
	target	Who are the closest men?
	target	Who is the closest adult?
	target	Who are the closest adults?

210	**source**	マスター カード は 使え ます か 。
	reference	*Can I pay with Master Card?*
	target	Is the master card useful?
	target	Are the master cards useful?
	target	Is a master card useful?
	target	Are master cards useful?
	target	Is the master card serviceable?

213	**source**	この ブランド に し ます 。
	reference	*I’ll have this brand.*
	target	I do it in this brand.
	target	You do it in this brand.
	target	They do it in this brand.
	target	We do it in this brand.
	target	I do it to this brand.

220	**source**	住所 氏名 が 付い て い ます 。
	reference	*My name and address are on it.*
	target	Address identity is starting.
	target	Residence identity is starting.
	target	The address identity is starting.
	target	Address identities are starting.
	target	The residence identity is starting.

222	**source**	これ は どんな 石 です か 。
	reference	*What kind of stone is this?*
	target	What stone is it?
	target	What stones is it?
	target	What gem is it?
	target	What stone is this?
	target	What jewel is it?

229	**source**	果物 を もっと 下さい 。
	reference	*More fruit, please.*
	target	Please give me fruit more.
	target	Please give me a fruit more.
	target	Please give me the fruit more.
	target	Please give me fruits more.
	target	Please give me the fruits more.

237	**source**	この ホテル は 良かっ た です よ 。
	reference	*I enjoyed my stay with you.*
	target	This hotel was good.

C.2 Tanaka Corpus

1	**source**	道 は そこ から 上り坂 に なっ て いる 。
	reference	*The road rises from there.*
	target	Roads are becoming ascent from there.
	target	The roads are becoming ascent from there.
	target	The roads are becoming ascents from there.
	target	The road will be being ascent from there.
	target	The road will be being an ascent from there.

2	**source**	彼 は 泳ぎ を 教え て くれ た 。
	reference	*He taught me how to swim.*
	target	He taught the swimming for him.
	target	He taught the swimming for me.
	target	He taught the swimming for you.
	target	He informed the swimming for him.
	target	He informed the swimming for me.

8	**source**	その 事故 で は 君 が 悪い の だ 。
	reference	*You are to blame for the accident.*
	target	You are bad in that accident.
	target	You are bad at that accident.
	target	You are bad with that accident.
	target	You are bad on that accident.
	target	You are bad by that accident.

12	**source**	バイオリン の 音色 は とても 美しい 。
	reference	*The sound of the violin is very sweet.*
	target	Violin tone quality is very beautiful.
	target	The violin tone quality is very beautiful.
	target	Violins’s tone colorred are very beautiful.
	target	A violin tone quality is very beautiful.
	target	The violins’s tone colorred are very beautiful.

17	**source**	彼 は 真実 を 言っ て い た 。
	reference	*He said truth.*
	target	He was saying truth.
	target	He was saying the truth.
	target	He was calling the truth.
	target	He was calling truth.
	target	He was saying truths.

18	**source**	彼女 の 詩 を どう 思い ます か 。
	reference	*What do you think of her poem?*
	target	How do you feel her poetry?
	target	How do you think her poetry?
	target	How do they think her poetry?
	target	How do I think her poetry?
	target	How do we feel her poetry?

20	**source**	戦争 は その 国 を 貧乏 にした。
	reference	*The war made the country poor.*
	target	War poorly did that country.
	target	The war did that country poorly.
	target	Warring poorly did that country.
	target	The wars poorly did that country.
	target	The warring poorly did that country.

22	**source**	彼 は 病院 で 気が付いた。
	reference	*He regained consciousness in the hospital.*
	target	He noticed at the hospital.
	target	He noticed in the hospital.
	target	He was aware at the hospital.
	target	He was aware in the hospital.
	target	You noticed him in a hospital.

25	**source**	警察 は 彼 の 失踪 を 調査 している。
	reference	*The police are looking into his disappearance.*
	target	Police are surveying his disappearance.
	target	The police are surveying his disappearance.
	target	Police are surveying his disappearances.
	target	The police are surveying his disappearances.

27	**source**	私 は いやいや その 仕事 を し た 。
	reference	*I did the work against my will.*
	target	I did that work unwillingly.
	target	I unwillingly did that work.
	target	They unwillingly did me that work.
	target	I unwillingly did me that work.
	target	He unwillingly did me that work.

32	**source**	彼 は 文句 無し の 巨人 だ 。
	reference	*He is altogether a giant.*
	target	He is phrase pears’s giant.
	target	He is the phrase pears’s giant.
	target	He is complaint pears’s giant.
	target	He is the phrase pear’s giant.
	target	He is phrase pears’s giants.

40	**source**	最善 を 尽くし なさい 。
	reference	*Do your best!*
	target	Serve the best.
	target	Exhaust the best.
	target	Serve as the best.

44	**source**	彼女 は 乱暴 な 運転 を する 人 に対して は 、いつも 批判的 だ 。
	reference	*She is always critical of reckless drivers.*
	target	She is always critical toward the rough men who drive.
	target	She is always critical toward the rough men that drive.
	target	She is always critical toward the rough man that drives.
	target	She is always critical toward the rough men which drive.
	target	She is always critical toward the rough man which drives.

45	**source**	鳥 が 空 を 高く 飛ん で いる 。
	reference	*Some birds are flying high in the sky.*
	target	Birds are highly flying sky.
	target	The birds are highly flying sky.
	target	The birds are highly flying skies.
	target	Birds are highly flying a sky.
	target	The birds are highly flying a sky.

48	**source**	彼女 は 私 の 手紙 を 見 て 腹 を 立て た 。
	reference	*She was displeased at my letter.*
	target	You saw my letter and it made her a stomach.
	target	It saw her, and it made my letter a stomach.
	target	It saw my letter and it made her the stomach.
	target	It saw my letter and it made her a stomach.
	target	It saw her, and it made my letters a stomach.

49	**source**	メイド は テーブル に ナイフ と フォーク を 並べ た 。
	reference	*The maid arranged the knives and forks on the table.*
	target	The maids set up the folk in the tables with a knife.
	target	The maids set up the folk on the tables with a knife.
	target	The maids set up the knife and folk at a table.
	target	The maids set up the knives and folk at a table.
	target	The maids set up the knife and folk in a table.

53	**source**	彼 は 恐怖 で 青ざめ た 。
	reference	*He turned pale with fear.*
	target	Turned pale at the fearing.
	target	Turned pale at the fearing
	target	Turned pale at fearing
	target	Turned pale in the fearing.
	target	Turned pale in fearing

56	**source**	偶然 その レストラン を 見つけ た 。
	reference	*I found that restaurant by accident.*
	target	He unexpectedly found that restaurant.
	target	They unexpectedly found that restaurant.
	target	She unexpectedly found that restaurant.
	target	You unexpectedly found that restaurant.
	target	We unexpectedly found that restaurant.

59	**source**	狭い 部屋 を せいぜい 広く 使った 。
	reference	*I made the best of my small room.*
	target	She at best widely used the small room.
	target	It at best widely used the small room.
	target	You at best widely used the small room.
	target	It widely at best used the small room.
	target	She at best widely used the narrow room.

67	**source**	その 銀行 は ここ から 遠い です か 。
	reference	*Is there bank far from here?*
	target	Is that bank distant from here?
	target	That bank is distant from here?

68	**source**	シェークスピア に 匹敵 する 劇 作家 は い ない 。
	reference	*No dramatist can compare with Shakespeare.*
	target	He, that the play writer matches in Shakespeare, doesn’t live.
	target	He, that the drama writer matches in Shakespeare, doesn’t live.
	target	He, that the play writer matches on Shakespeare, doesn’t live.
	target	He, that the play writer matches in Shakespeare, doesn’t happen.
	target	She, that the play writer matches in Shakespeare, doesn’t live.

71	**source**	彼 は なぜ そんな こと を した の か 。
	reference	*Why did he do that?*
	target	Why did he do that terminology?
	target	Why did you do him that terminology?
	target	Why did they do him that terminology?
	target	Why did he do him that terminology?
	target	Why did she do him that terminology?

72	**source**	医学 で は 日本 は 欧米 に 追いつき まし た 。
	reference	*Japan has caught up with Europe and America in medicine.*
	target	Japan overtaken in the medical science in Oubei.
	target	Japan overtaken in the medical science on Oubei.
	target	It overtaken Japan in the medical science at Oubei.
	target	Japan overtaken on the medical science in Oubei.
	target	It overtaken Japan on the medical science at Oubei.

90	**source**	病気 は 人類 にとって 脅威 で ある 。
	reference	*Disease is a threat to human beings.*
	target	The diseases are threat for a unit of mankind.
	target	The diseases are threats for a unit of mankind.
	target	The diseases are threat for the unit of mankind.
	target	The diseases are menace for a unit of mankind.
	target	There are diseases for a unit of mankind with a threat.

104	**source**	星 が 空 に 光っ て い ます 。
	reference	*The stars are shining in the sky.*
	target	Stars are glitterring in a sky.
	target	Stars are glitterring on a sky.
	target	The stars are glitterring in a sky.
	target	The stars are glitterring on a sky.
	target	The stars are glitterring at a sky.

References

Peter Adolphs, Stephan Oepen, Ulrich Callmeier, Berthold Crysmann, Dan Flickinger, and Bernd Kiefer. Some fine points of hybrid natural language parsing. In European Language Resources Association (ELRA), editor, Proceedings of the Sixth International Language Resources and Evaluation: LREC 2008, Marrakech, Morocco, 2008.

Eneko Agirre, Olatz Ansa, Xabier Arregi, Xabier Artola, Arantza Diaz de Ilarraza, Mikel Lersundi, David Martinez, Kepa Sarasola, and Ruben Urizar. Extraction of semantic relations from a Basque monolingual dictionary using Constraint Grammar. In Proceedings of the Ninth EURALEX International Congress: EURALEX 2000, 2000.

Shigeaki Amano and Tadahisa Kondo. Nihongo-no Goi-Tokusei (Lexical properties of Japanese). Sanseido, 1999.

Carme Armentano-Oller, Antonio M. Corbí-Bellot, Mikel L. Forcada, Mireia Ginestí-Rosell, Boyan Bonev, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Gema Ramírez-Sánchez, and Felipe Sánchez-Martínez. An open-source shallow-transfer machine translation toolbox: consequences of its release and availability. In Proceedings of Open-Source Machine Translation: Workshop at MT Summit X, pages 23–30, Phuket, 2005.

Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

Francis Bond, Stephan Oepen, Melanie Siegel, Ann Copestake, and Dan Flickinger. Open source machine translation with DELPH-IN. In Proceedings of Open-Source Machine Translation: Workshop at MT Summit X, pages 15–22, Phuket, 2005.

Francis Bond, Takayuki Kuribayashi, and Chikara Hashimoto. Construction of a free Japanese treebank based on HPSG. In Proceedings of the 14th Annual Meeting of the Association for Natural Language Processing: NLP 2008, pages 241–244, Tokyo, 2008. (in Japanese).

James W. Breen. Word usage examples in an electronic dictionary. In Proceedings of the Papillon (Multi-lingual Dictionary) Project Workshop, Sapporo, 2003.

James W. Breen. JMDict: a Japanese-multilingual dictionary. In Proceedings of the Coling 2004 Workshop on Multilingual Linguistic Resources, pages 71–78, Geneva, 2004.

Lou Burnard. The British National Corpus Users Reference Guide. Oxford University Computing Services, 2000.

Chris Callison-Burch, Phillip Koehn, and Miles Osborne. Improved statistical machine translation using paraphrases. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pages 17–24, 2006.

Ulrich Callmeier. PET - a platform for experimentation with efficient HPSG processing techniques. Natural Language Engineering, 6(1):99–108, 2000.

Ulrich Callmeier. Preprocessing and encoding techniques in PET. In Stephan Oepen, Dan Flickinger, Jun-ichi Tsujii, and Hans Uszkoreit, editors, Collaborative Language Engineering, chapter 6, pages 127–143. CSLI Publications, Stanford, 2002.

Ulrich Callmeier, Andreas Eisele, Ulrich Schäfer, and Melanie Siegel. The DeepThought core architecture framework. In Proceedings of the Fourth International Conference on Language Resources and Evaluation: LREC 2004, volume IV, Lisbon, 2004.

David Chiang. A hierarchical phrase-based model for statistical machine translation. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics: ACL 2005, pages 263–270, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219873.

Michael Collins, Philipp Koehn, and Ivona Kŭcerová. Clause restructuring for statistical machine translation. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics: ACL 2005, pages 531–540, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

Ann Copestake. Implementing Typed Feature Structure Grammars. CSLI Publications, 2002.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A. Sag. Minimal Recursion Semantics. An introduction. Research on Language and Computation, 3(4):281–332, 2005.

Christine Fellbaum, editor. WordNet: An Electronic Lexical Database. MIT Press, 1998.

Dan Flickinger. On building a more efficient grammar by exploiting types. Natural Language Engineering, 6(1):15–28, 2000. (Special Issue on Efficient Processing with HPSG).

Dan Flickinger, Jan Tore Lønning, Helge Dyvik, Stephan Oepen, and Francis Bond. SEM-I rational MT: Enriching deep grammars with a semantic interface for scalable machine translation. In Proceedings of the Tenth Machine Translation Summit X, pages 165–172, Phuket, 2005.

Eva Forsbom. Training a super model look-alike: Featuring edit distance, n-gram occurrence, and one reference translation. In Proceedings of the Workshop on Machine Translation Evaluation. Towards Systemizing MT Evaluation, 2003.

Anette Frank. Constraint-based RMRS construction from shallow grammars. In Proceedings of the 20th International Conference on Computational Linguistics: COLING 2004, pages 1269–1272, Geneva, 2004.

Sanae Fujita, Francis Bond, Stephan Oepen, and Takaaki Tanaka. Exploiting semantic information for HPSG parse selection. In Proceedings of the ACL 2007 Workshop on Deep Linguistic Processing, pages 25–32, Prague, Czech Republic, 2007.

Kenneth Goodman and Sergei Nirenburg, editors. The KBMT Project: A Case Study in Knowledge-based Machine Translation. Morgan Kaufmann Publishers, San Mateo, California, 1989.

Michael Wayne Goodman and Francis Bond. Using generation for grammar analysis and error detection. In Proceedings of the Joint conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing: ACL/IJCNLP 2009, pages 109–112, Singapore, 2009.

Francisco Guzmán Herrera and Leonardo Garrido Luna. Using translation paraphrases from trilingual corpora to improve phrase-based statistical machine translation: A preliminary report. In Proceedings of the Mexican International Conference on Artificial Intelligence, pages 163–172, Los Alamitos, CA, USA, 2007. IEEE Computer Society. ISBN 978-0-7695-3124-3.

Satoru Ikehara, Satoshi Shirai, Akio Yokoo, and Hiromi Nakaiwa. Toward an MT system without pre-editing — effects of new methods in **ALT-J/E** —. In Proceedings of the Third Machine Translation Summit: MT Summit III, pages 101–106, Washington DC, 1991.

Satoru Ikehara, Masahiro Miyazaki, Satoshi Shirai, Akio Yokoo, Hiromi Nakaiwa, Kentaro Ogura, Yoshifumi Ooyama, and Yoshihiko Hayashi. Goi-Taikei — A Japanese Lexicon. Iwanami Shoten, Tokyo, 1997. 5 volumes/CDROM.

Kenji Imamura, Eiichiro Sumita, and Yuji Matsumoto. Feedback cleaning of machine translation rules using automatic evaluation. In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pages 447–454, Sapporo, Japan, July 2003. Association for Computational Linguistics.

Michael Jellinghaus. Automatic acquisition of semantic transfer rules for machine translation. Master's thesis, Universität des Saarlandes, 2007.

Ronald M. Kaplan and Joan Bresnan. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan, editor, The Mental Representation of Grammatical Relations, pages 173–281. MIT Press, Cambridge, MA, 1982.

Kaname Kasahara, Hiroshi Sato, Francis Bond, Takaaki Tanaka, Sanae Fujita, Tomoko Kanasugi, and Shigeaki Amano. Construction of a Japanese semantic lexicon: Lexeed. In IPSJ SIG Technical Reports: 2004-NLC-159, pages 75–82, Tokyo, 2004. (in Japanese).

Jason Katz-Brown and Michael Collins. Syntactic reordering in preprocessing for Japanese→English translation: MIT system description for NTCIR-7 patent translation task. In Proceedings of the 7th NII Test Collection for IR Systems Workshop Meeting: NTCIR-7, Tokyo, Japan, December 16 – 19 2008.

Philip Koehn. Statistical significance tests for machine translation evaluation. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing: EMNLP 2004, Barcelona, Spain, July 2004.

Philipp Koehn, Wade Shen, Marcello Federico, Nicola Bertoldi, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Ondrej Bojar, Richard Zens, Alexandra Constantin, Evan Herbst, Christine Moran, and Alexandra Birch. Moses: Open source toolkit for statistical machine translation. In Proceedings of the ACL 2007 Demo and Poster Sessions, pages 177–180, Prague, 2007.

Mamoru Komachi, Yuji Matsumoto, and Masaaki Nagata. Phrase reordering for statistical machine translation based on predicate-argument structure. In Proceedings of the 4th International Workshop on Spoken Language Translation: IWSLT 2006, Kyoto, Japan, 2006.

Taku Kudo and Yuji Matsumoto. Japanese dependency analysis using cascaded chunking. In CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops), pages 63–69, Taipei, 2002.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying Conditional Random Fields to Japanese Morphological Analysis. In Proceedings of the 2004 Conference

on Empirical Methods in Natural Language Processing: EMNLP 2004, pages 230–237, Barcelona, Spain, July 2004. Association for Computational Linguistics.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. Using paraphrases for parameter tuning in statistical machine translation. In Proceedings of the Second Workshop on Statistical Machine Translation, pages 120–127, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. Improved statistical machine translation using monolingually-derived paraphrases. In Proceedings of the 2009 Conference in Empirical Methods on Natural Language Processing: EMNLP 2009, pages 381–390, Singapore, August 2009. Association for Computational Linguistics.

Yuji Matsumoto, Kazuma Takaoka, and Masayuki Asahara. ChaSen Morphological Analyzer version 2.4.0 User's Manual, 2007.

Preslav Nakov. Improved statistical machine translation using monolingual paraphrases. In Proceedings of the European Conference on Artificial Intelligence: ECAI 2008, Patras, Greece, 2008.

Sonja Nießen and Hermann Ney. Morpho-syntactic analysis for reordering in statistical machine translation. In Proceedings of the Eighth Machine Translation Summit: MT Summit VIII, pages 247–252, 2001.

Minoru Nishio, Etsutaro Iwabuchi, and Shizuo Mizutani. Iwanami Kokugo Jiten Dai Go Han [Iwanami Japanese Dictionary Edition 5]. Iwanami Shoten, Tokyo, 1994. (in Japanese).

Lars Nygård, Jan Tore Lønning, Torbjørn Nordgård, and Stephan Oepen. Using a bilingual dictionary in lexical transfer. In Proceedings of the 11th Annual conference of the European Association for Machine Translation: EAMT 2006, pages 233–238, Oslo, 2006.

Franz Josef Och. Statistical machine translation: Foundations and recent advances. In Proceedings of the Tenth Machine Translation Summit Tutorial: MT Summit X, Phuket, 2005. MT Summit.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. Computational Linguistics, 29(1):19–51, 2003.

Stephan Oepen, Helge Dyvik, Jan Tore Lønning, Erik Velldal, Dorothee Beermann, John Carroll, Dan Flickinger, Lars Hellan, Janne Bondi Johannessen, Paul Meurer, Torbjørn Nordgård, and Victoria Rosèn. Som å kapp-ete med trollet? Towards MRS-based Norwegian–English Machine Translation. In Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation: TMI-2004, Baltimore, MD, October 2004a.

Stephan Oepen, Dan Flickinger, Kristina Toutanova, and Christoper D. Manning. LinGO redwoods: A rich and dynamic treebank for HPSG. Research on Language and Computation, 2(4):575–596, 2004b.

Stephan Oepen, Erik Velldal, Jan Tore Løning, Paul Meurer, and Victoria Rosen. Towards hybrid quality-oriented machine translation —On linguistics and probabilities in MT—. In Proceedings of the Eleventh International Conference on Theoretical and Methodological Issues in Machine Translation: TMI 2007, Skövde, 2007.

Kentaro Ogura, Francis Bond, and Yoshifumi Ooyama. **ALT-J/M**: A prototype Japanese-to-Malay translation system. In Proceedings of the Seventh Machine Translation Summit: MT Summit VII, pages 444–448, Singapore, 1999.

Patrick Pantel, Deepak Ravichandran, and Eduard Hovy. Towards terascale knowledge acquisition. In Proceedings of the 20th International Conference on Computational Linguistics: COLING 2004, pages 771–777, Geneva, 2004.

Michael Paul. Overview of the IWSLT 2006 Evaluation Campaign. In Proceedings of the 4th International Workshop on Spoken Language Translation: IWSLT 2006, pages 1–15, Kyoto, Japan, 2006.

Carl J. Pollard and Ivan A. Sag. Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago, 1994.

Arjen Poutsma. Data-oriented translation. In Proceedings of the 18th Conference on Computational Linguistics: ACL 2000, pages 635–641, Morristown, NJ, USA, 2000. Association for Computational Linguistics.

Stephen D. Richardson, William B. Dolan, and Lucy Vanderwende. MindNet: acquiring and structuring semantic information from text. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics: COLING/ACL 1998, pages 1098–1102, Montreal, 1998.

Ivan A. Sag, Thomas Wasow, and Emily M. Bender. Syntactic Theory: A Formal Introduction. CSLI Publications, Stanford, 2 edition, 2003.

Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In Proceedings of the International Conference on New Methods in Language Processing, Manchester, UK, 1994.

Bernard (Bud) Scott. The Logos model: An historical perspective. Machine Translation, 18:1–72, 2003.

Kiyoaki Shirai. SENSEVAL-2 Japanese dictionary task. Journal of Natural Language Processing, 10(3):3–24, 2003. (in Japanese).

Satoshi Shirai, Satoru Ikehara, and Tsukasa Kawaoka. Effects of automatic rewriting of source language within a Japanese to English MT system. In Proceedings of the Fifth International Conference on Theoretical and Methodological Issues in Machine Translation, pages 226–239, Kyoto, Japan, July 14–16 1993.

Melanie Siegel. HPSG analysis of Japanese. In Wahlster (2000), pages 265 – 280.

Kathrin Spreyer and Anette Frank. The TIGER RMRS 700 bank: RMRS construction from dependencies. In Proceedings of the 6th International Workshop on Linguistically Interpreted Corpora: LINC 2005, pages 1–10, Jeju Island, Korea, 2005.

Andreas Stolcke. Srilm - an extensible language modeling toolkit. In Proceedings of the 7th International Conference on Spoken Language Processing: ICSLP 2002, volume 2, pages 901–904, Denver, 2002.

Tatsuya Sukehiro, Mihoko Kitamura, and Toshiki Murata. Collaborative translation environment 'Yakushite.Net'. In Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium: NLPRS 2001, pages 769–770, Tokyo, 2001.

Yasuhito Tanaka. Compilation of a multilingual parallel corpus. In Proceedings of PACLING 2001, pages 265–268, Kyushu, 2001.

Takenobu Tokunaga, Yasuhiro Syotu, Hozumi Tanaka, and Kiyoaki Shirai. Integration of heterogeneous language resources: A monolingual dictionary and a thesaurus. In Proceedings of the 6th Natural Language Processing Pacific Rim Symposium: NLPRS 2001, pages 135–142, Tokyo, 2001.

Erik Velldal and Stephan Oepen. Statistical ranking in tactical generation. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing: EMNLP 2006, pages 517–525, Sydney, Australia, July 2006. Association for Computational Linguistics.

Wolfgang Wahlster, editor. Verbmobil: Foundations of Speech-to-Speech Translation. Springer, Berlin, Germany, 2000.

Taro Watanabe, Mitsuo Shimohata, and Eiichiro Sumita. Statistical machine translation on paraphrased corpora. In Proceedings of the Third International Conference on Language Resources and Evaluation: LREC 2002, pages 2074–2081, Las Palmas, Spain, May 27th – June 2nd 2002.

Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. NTT Statistical Machine Translation for IWSLT 2006. In Proceedings of the 4th International Workshop on Spoken Language Translation: IWSLT 2006, pages 95–102, Kyoto, Japan, 2006.

Andy Way. A hybrid architecture for robust MT using LFG-DOP. Journal of Experimental and Theoretical Artificial Intelligence, 11, 1999. Special Issue on Memory-Based Language Processing.

Yorick A. Wilkes, Brian M. Slator, and Louise M. Guthrie. Electric Words. MIT Press, 1996.

Yushi Xu and Stephanie Seneff. Two-stage translation: A combined linguistic and statistical machine translation framework. In Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas: AMTA 2008, Honolulu, Hawaii, 2008.

Kazuhide Yamamoto. Paraphrasing spoken Japanese for untangling bilingual transfer. In Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium: NLPRS 2001, pages 203–210, 2001.

Publication List

Journal Articles

Eric Nichols, Francis Bond, D. Scott Appling, and Yuji Matsumoto. Paraphrasing training data for statistical machine translation. Journal of Natural Language Processing, 17(2), 2010. URL http://cl.naist.jp/~eric-n/papers/jnlp-2010-emalp.pdf. Special Issue on Empirical Methods for Asian Language Processing (to appear).

Refereed International Conferences and Workshops

Francis Bond, Eric Nichols, Sanae Fujita, and Takaaki Tanaka. Acquiring an ontology for a fundamental vocabulary. In Proceedings of the 20th International Conference on Computational Linguistics: COLING 2004, pages 1319–1325, Geneva, 2004. URL http://www.aclweb.org/anthology-new/C/C04/C04-1193.pdf.

Francis Bond, Eric Nichols, Darren Scott Appling, and Michael Paul. Improving statistical machine translation by paraphrasing the training data. In Proceedings of the International Workshop on Spoken Language Translation 2008: IWSLT 2008, pages 150–157, Hawaii, 2008. URL http://www.mt-archive.info/IWSLT-2008-Bond.pdf.

Eric Nichols, Francis Bond, and Daniel Flickinger. Robust ontology acquisition from machine-readable dictionaries. In Proceedings of the International Joint Conference on Artificial Intelligence: IJCAI 2005, pages 1111–1116, Edinburgh, Aug 2005. URL http://ijcai.org/papers/1470.pdf.

Eric Nichols, Francis Bond, Takaaki Tanaka, Sanae Fujita, and Daniel Flickinger. Multilingual ontology acquisition from multiple MRDs. In Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge, pages 10–17, Sydney, 2006. URL http://www.aclweb.org/anthology/W/W06/W06-0502.pdf.

Eric Nichols, Francis Bond, Darren Scott Appling, and Yuji Matsumoto. Combining resources for open source machine translation. In Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation: TMI 2007, Sept 2007. URL http://www.mt-archive.info/TMI-2007-Nichols.pdf.

Other Refereed Publications

Francis Bond, Sanae Fujita, Chikara Hashimoto, Kaname Kasahara, Shigeko Nariyama, Eric Nichols, Akira Ohtani, Takaaki Tanaka, and Shigeaki Amano. The Hinoki treebank: Working toward text understanding. In Proceedings of the 5th International Workshop on Linguistically Interpreted Corpora: LINC 2004, pages 7–10, 2004a. URL http://cl.naist.jp/~eric-n/papers/hinoki-LINC-2004-en.pdf.

Francis Bond, Sanae Fujita, Chikara Hashimoto, Kaname Kasahara, Shigeko Nariyama, Eric Nichols, Akira Ohtani, Takaaki Tanaka, and Shigeaki Amano. The Hinoki treebank: A treebank for text understanding. In Proceedings of the First International Joint Conference on Natural Language Processing: IJCNLP 2004, pages 158–167. Springer Verlag, 2004b. URL http://cl.naist.jp/~eric-n/papers/hinoki-intro-IJCNLP-2004-en.pdf.

Koji Murakami, Shouko Masuda, Suguru Matsuyoshi, Eric Nichols, Kentaro Inui, and Yuji Matsumoto. Annotating semantic relations combining facts and opinions. In ACL-IJCNLP 2009: Proceedings of the Third Linguistic Annotation Workshop: LAW III, pages 150–153, Morristown, NJ, USA, 2009a. Association for Computational Linguistics. ISBN 978-1-932432-52-7. URL http://www.aclweb.org/anthology/W/W09/W09-3027.pdf.

Koji Murakami, Eric Nichols, Suguru Matsuyoshi, Asuka Sumida, Shouko Masuda, Kentaro Inui, and Yuji Matsumoto. Statement Map: Assisting information credibility analysis by visualizing arguments. In Proceedings of the 3rd Workshop on Information Credibility on the Web: WICOW 2009, pages 43–50, 2009b. URL http://www.dl.kuis.kyoto-u.ac.jp/wicow3/papers/p43-murakamiA.pdf.

Shigeko Nariyama, Eric Nichols, Francis Bond, Takaaki Tanaka, and Hiromi Nakaiwa. Extracting representative arguments from dictionaries for resolving zero pronouns. In Proceedings of the Tenth Machine Translation Summit: MT Summit X, pages 3–10, Phuket, 2005a. URL http://www.mt-archive.info/MTS-2005-Nariyama.pdf.

Shigeko Nariyama, Takaaki Tanaka, Eric Nichols, Francis Bond, and Hiromi Nakaiwa. Building a cross-lingual referential knowledge database using dictionaries. In Proceedings of Recent Advances in Natural Language Processing: RANLP 2005, pages 354–360, Sofia, Bulgaria, 2005b. URL http://cl.naist.jp/~eric-n/papers/ranlp-2005-nariyama.pdf.

Eric Nichols and Yuji Matsumoto. Acme as an interactive translation environment. Proceedings of the Second International Workshop on Plan 9: IWP9 2007, pages 35–46, Dec 2007. URL http://cl.naist.jp/~eric-n/papers/acme-trans.pdf.

Eric Nichols, Koji Murakami, Kentaro Inui, and Yuji Matsumoto. Constructing a scientific blog corpus for information credibility analysis. In Proceedings of Pacling 2009, 2009. URL http://cl.aist-nara.ac.jp/~eric-n/papers/bscorpus-pacling2009-paper.pdf.

Local Conferences and Workshops

Ai Azuma, Eric Nichols, Yoshihiro Morimoto, and Yuji Matsumoto. Integration of statistical dependency parsing and constraint based grammar for Japanese sentence analysis. In IPSJ SIG Technical Reports: 2004-NLC-159, volume 1, pages 131–138, Tokyo, 2004. (in Japanese).

Francis Bond, Sanae Fujita, Chikara Hashimoto, Kaname Kasahara, Shigeko Nariyama, Eric Nichols, Akira Ohtani, Takaaki Tanaka, and Shigeaki Amano. The Hinoki treebank: A treebank for text understanding. In IPSJ SIG Technical Reports: 2004-NLC-159, pages 83–90, 2004a. (in Japanese).

Francis Bond, Sanae Fujita, Chikara Hashimoto, Shigeko Nariyama, Eric Nichols, Akira Ohtani, and Takaaki Tanaka. Development of the Hinoki treebank based on a precise grammar. In IPSJ SIG Technical Reports: 2004-NLC-159, pages 91–98, 2004b. (in Japanese).

Francis Bond, 藤田 早苗, 橋本 力, 笠原 要, 成山 重子, Eric Nichols, 大谷 朗, 田中 貴秋, and 天野 成昭. 日本語ツリーバンク「檜」: 言語理解のためのコーパス. In IPSJ SIG Technical Reports: 2004-NLC-159, pages 83–90, 2004c.

Francis Bond, 藤田 早苗, 橋本 力, 成山 重子, Eric Nichols, 大谷 朗, and 田中 貴秋. 精細な文法に基づいたツリーバンク「檜」の構築. In IPSJ SIG Technical Reports: 2004-NLC-159, pages 91–98, 2004d.

Francis Bond, Eric Nichols, and James W. Breen. Enhancing a dictionary for transfer rule acquisition. Seoul, South Korea, December 2007. Center for the Study of Language, Kyung Hee University.

Eric Nichols and Francis Bond. Acquiring ontologies using deep and shallow processing. In Proceedings of the Eleventh Annual Meeting of the Association for Natural Language Processing: NLP 2005, pages 494–498, 2005.

Eric Nichols and Yuji Matsumoto. Using dependency relations as constraints in HPSG parsing. In Proceedings of the Eleventh Annual Meeting of the Association for Natural Language Processing: NLP 2005, pages 899–902, 2005.

Eric Nichols, Francis Bond, and Yuji Matsumoto. Automatic transfer rule acquisition for semantic transfer based MT. In IPSJ SIG Technical Reports: 2007-NLC-178, pages 77–84, 2007.

Eric Nichols, Koji Murakami, Kentaro Inui, and Yuji Matsumoto. Constructing a scientific blog corpus for information credibility analysis. Proceedings of the Fifteenth Annual Meeting of the Association for Natural Language Processing: NLP 2009, Jan 2009.

村上浩司, 増田祥子, 松吉俊, Eric Nichols, 乾健太郎, and 松本裕治. 複数文書から抽出した言明間の意味的関係の整理と関係付与. In IPSJ SIG Technical Reports: 2009-NLC-192, 2009.