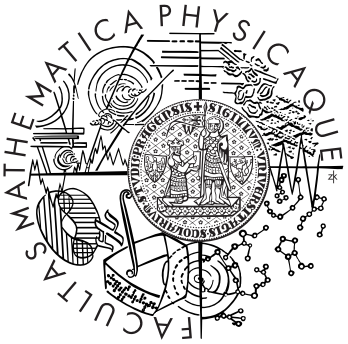
Exploiting Linguistic Data in Machine Translation

Ondřej Bojar

Doctoral Thesis



Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Prague, 2008

Author: RNDr. Ondřej Bojar

Supervisor: RNDr. Vladislav Kuboň, Ph.D.
Institute of Formal and Applied Linguistics (ÚFAL)

Department: Institute of Formal and Applied Linguistics (ÚFAL)
Faculty of Mathematics and Physics
Charles University in Prague
Malostranské náměstí 25, 118 00 Praha 1

Opponents: Ing. Alexandr Rosen, Ph.D.
Institute of Theoretical and Computational Linguistics
Faculty of Philosophy & Arts, Charles University, Prague
Celetná 13, 110 00 Praha 1

RNDr. Jan Cuřín, Ph.D.
IBM Česká republika, spol. s r.o. Voice Technologies and Systems
V Parku 2294/4, 148 00 Praha 4 – Chodov

I certify that this doctoral thesis is all my work, and that I used only the cited literature. The thesis is freely available for all who can use it.

Prague, June 16, 2008.

Ondřej Bojar

Acknowledgment

I would like to express my gratitude to the Institute of Formal and Applied Linguistics (ÚFAL) for excellent support and to my colleagues for all those stimulating discussions. It is impossible to name everyone who influenced my research, so here is an abbreviated list: my supervisor Vláďa Kuboň, the head of our department Jan Hajič, and all the numerous friends and colleagues at ÚFAL; those I met at informatics in Hamburg (Wolfgang Menzel, Michael Daum and others), at the Programming Systems Lab and CoLi in Saarbrücken (Ralph Debusmann, Marco Kuhlmann, Ivana and Geert-Jan Kruijff, Valia Kordoni and many others); the very influential team of Hermann Ney at RWTH Aachen University (Richard Zens, Saša Hasan and many others again); Philipp Koehn's MT team at the Johns Hopkins University summer workshop, MT Marathons and in Edinburgh (Chris Callison-Burch, Chris Dyer, Hieu Hoang, Phil Blunsom and Adam Lopez to name a few) as well as the very warm and inspiring groups in Melbourne: the LT group of Steven Bird, Tim Baldwin and many others; and the Mercury team (Ralph Becket, Julien Fischer and others). Also, I do not wish to forget all the short random friendly encounters with members of our community at summer schools, workshops or conferences.

And last but not least, this thesis would never have come into being without the support of my greater family, my parents, and my wife Pavla.

The work reported in this thesis was also partially supported by the following grants: FP6-IST-5-034291-STP (EuroMatrix), MSM0021620838, MŠMT ČR LC536, GA405/06/0589, GAČR 201/05/H014, GAAV ČR 1ET201120505, GAUK 351/2005, and Collegium Informaticum GAČR 201/05/H014.

Abstract

This thesis explores the mutual relationship between linguistic theories, data and applications. We focus on one particular theory, Functional Generative Description (FGD), one particular type of linguistic data, namely valency dictionaries and one particular application: machine translation (MT) from English to Czech.

First, we examine methods for automatic extraction of verb valency dictionaries based on corpus data. We propose an automatic metric for estimating how much lexicographers' labour was saved and evaluate various frame extraction techniques using this metric.

Second, we design and implement an MT system with transfer at various layers of language description, as defined in the framework of FGD. We primarily focus on the tectogrammatical (deep syntactic) layer.

Third, we leave the framework of FGD and experiment with a rather direct, "phrase-based" MT system. Comparing various setups of the system and specifically treating target-side morphological coherence, we are able to significantly improve MT quality and out-perform a commercial MT system within a pre-defined text domain.

The concluding chapter provides a broader perspective on the utility of lexicons in various applications, highlighting the successful features. Finally, we summarize the contribution of the thesis.

Contents

Chapter 1

Introduction

Computational linguistics and natural language processing (NLP) try to formally capture and model the complexity of how people communicate using a natural language. The field has implications in many aspects of the society: linguistic theories are used as a basis when prescribing what is an appropriate and correct usage of an expression, they predict how a message is perceived by a human recipient and justify which information should be included in language textbooks, dictionaries or lexicons. Applications are built to speed up human processing of text (such as finding relevant documents, answering questions, translating from one language to another) or attempt to turn the computer into a real partner able to share knowledge and obey commands issued in a natural language.

1.1 Relation between Theory, Applications and Data

Both linguistic theories and NLP applications rely heavily on language data, which include raw examples of language expressions (written sentences in books, newspapers, sentences uttered in a dialog, recorded or broadcasted) as well as more or less formalized data *about* the language itself (such as style guides or dictionaries). On the one hand, examples of language usage can validate linguistic theories (by testing predictions on real data) and on the other hand, linguistic theories provide a framework for creating derived language resources like the above mentioned lexicons and dictionaries. Thus, the theory is tested indirectly, by applying and using a derived resource in a practical task. NLP applications are related to data even more tightly simply because the application has some input and output data. Moreover, many NLP applications need to consult varying amounts of language data in order to be able to achieve their goal.

In this thesis, we study the mutual relationship between a linguistic theory, an NLP application and language data. We focus on one particular theory, the theory of Functional Generative Description (FGD), one particular type of derived language data, namely valency dictionaries, and on one

particular NLP application, namely machine translation (MT). Whenever possible, we try to include references to relevant alternatives.

1.2 How Theory Should Help

The general belief is that having an established theory as a background of an NLP application should bring an advantage to the design of the application: the description of the algorithm could be shorter because it builds on top of notions defined in the theory, decisions that have to be made should be more local and thus easier to meet and finally, such an application should produce outputs of a predictable quality. In short, a good theory should constrain the internal structure of applications to their advantage.

There is a similar relation between the theory and language data: a good theory describes which features of unprocessed language data are significant for a particular task. A theory provides a view on unprocessed data. Given a task and following the theory, we can "compress" raw language data by ignoring all but relevant features. Dictionaries are an excellent example of such compression: instead of scanning large texts and looking at many occurrences of a word to understand the meaning and correct ways of using it in context we just read a short (formal) description.

In an NLP application such as MT, there is always someone who has to do the difficult job. In the extreme case, all the intelligence is contained in a "dictionary", i.e. the "dictionary" provides the expected output of the application for every possible input. More realistically, we can expect to know at least *parts* of the output from the top of our head but we have to correctly glue them together to create a complete answer. The more or the better training data we have, the simpler the application can be.

To sum up, a theory provides guidelines on how to build linguistic applications and how to look at language data. If all goes well, such a theoretical background will simplify the design and facilitate better performance at the same time. We study the relationship between the theory and practical applications throughout the thesis, the structure of which is outlined in the following section.

1.3 Structure of the Thesis

This thesis has two major parts: the first one is devoted to lexical acquisition (Chapter 2) and the second one to machine translation (Chapters 3 and 4), linked as follows:

One of the key components in the theory of our choice, FGD (briefly introduced in Section 2.2), is the valency theory which predicts how an element in a grammatically well formed sentence can or must be accompanied by other elements. The prediction primarily depends on the sense of the governing word and it is best captured in a lexicon. The motivation to build such lexicons comes often from applications: some applications simply require a lexicon to e.g. produce an output text, while some only benefit from them by improving accuracy or increasing coverage. Finally, a syntactic lexicon is always a valuable reference for human users of the language. However, the development of lexicons is costly and therefore we focus on the question of automatic suggestion of entries based on available textual data. In short, Chapter 2 explores the theory of FGD and the journey from raw language data in a text to a compressed formalized representation in a lexicon.

In Chapter 3 we pick an NLP application, the task of machine translation (MT) in particular, to study how the theory lends itself to practical employment. After a brief review of various approaches to MT, we follow up on FGD and describe our system of syntax-based machine translation. The full complexity of the system is outlined, but the main focus is given only to our contribution, syntactic transfer. Nevertheless, we implement the whole pipeline of the MT system and we are able to evaluate MT quality using an established automatic metric.

Chapter 4 is devoted to a contrast experiment: we aim at English to Czech MT leaving the framework of FGD aside and using a rather direct method. We briefly summarize the state-of-the-art approach, so-called phrase-based statistical machine translation, including an extension to factored MT where various linguistically motivated aspects can be explicitly captured. Then we demonstrate how to use factors to improve morphological coherence of MT output and compare the performance of the direct approach with the syntax-based system from Chapter 3.

We conclude by Chapter 5, providing a broad survey of documented utility of lexicons in NLP and summarizing our observations and contributions of the thesis.

Chapter 2

Extracting Verb Valency Frames

2.1 Introduction

Verb valency frames[1] formally describe the potential of a verb to combine with other elements in the sentence.[2]

When analyzing an input sentence, the knowledge of the verb frame allows resolving ambiguity at various levels. Consult e.g. Straňáková-Lopatková and Žabokrtský (2002) for simple examples or Hlaváčková *et al.* (2006) for a report on a dramatic reduction in parsing ambiguity.

When generating text from some deep representation, the valency frame of the verb is used to choose the appropriate morphemic form (e.g. the preposition and case) of a modifier and thus to guarantee grammaticality of the output. For some systems, the existence of a valency lexicon is a strict requirement, e.g. RUSLAN (Hajič, 1987; Hajič *et al.*, 1987; Oliva, 1989); for some systems, the valency information is optionally used do refine the output, e.g. (Ptáček and Žabokrtský, 2006).

2.2 FGD and Valency Theory

This section introduces Functional Generative Description (FGD) and its valency theory, including relevant available data.

2.2.1 Layers of Language Description

Let us briefly summarize key components of FGD related to our task. However, since it is not the aim of the thesis to review FGD in detail, please

[1] The term "valency frame" is defined and used in dependency analysis in the framework of FGD theory, see below. A related notion in phrase-structure grammars is traditionally called "subcategorization frames".

[2] Valency frames can be assigned also to nouns, adjectives and possibly other parts of speech. We focus on verbs only.

consult relevant books, reports or tutorials, e.g. PDT Guide[3], Sgall *et al.* (1986), Hajič *et al.* (2006) or Mikulová *et al.* (2006) to get acquainted with the theory and to find definitions of all notions not explained here.

FGD as implemented in the Prague Dependency Treebank (Section 2.2.3 below) defines three layers of language representation called **morphological** (or m-layer), **analytical** (a-layer, corresponds to surface syntax) and **tectogrammatical** (t-layer, corresponds to deep syntax) to annotate an original text (the wordform, w-layer, where even typographical errors are stored verbatim, e.g. no space between *do* and *lesa*), see Figure 2.1:

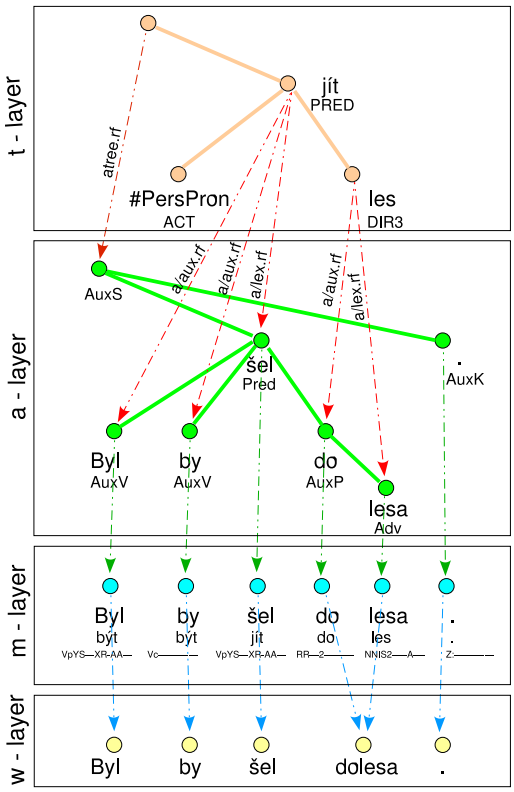


Figure 2.1: Layers of annotation of Czech as implemented in Prague Dependency Treebank. (Picture from the PDT Guide.)

M-layer represents the sentence as a sequence of word forms accompanied by their lemmas (base forms) and morphological tags that include part-of-speech and many other relevant categories such as case, gender, number, or tense.

[3] `http://ufal.mff.cuni.cz/pdt2.0/doc/pdt-guide/en/html/`

A-layer and t-layer use a rooted labelled dependency tree to encode the relations between elements of the sentence. Edge labels, sometimes stored as an attribute of the dependent node, are called **afuns** (e.g. Pred, Sb, Obj) at the a-layer and **functors** (e.g. PRED, ACT, PAT) at the t-layer and they formally describe the relation between the governing and dependent node.

At the a-layer, nodes in the tree correspond one to one to words in the input sentence.

At the t-layer, words bearing meaning have a corresponding node while all auxiliary words only contribute to some attributes of relevant nodes. On the other hand, the t-layer includes nodes for entities that were not explicitly expressed in the sentence but the language syntax and lexicon indicate their presence in the described situation. This is one of several reasons that make the t-layer language dependent and not an Interlingua.

2.2.2 Basics of Valency Theory in FGD

In FGD, (verb) **valency frames** are defined at the t-layer only and describe formal requirements on the immediate dependents of the verbal t-node (Panevová, 1980; Hajič *et al.*, 2006). Here is a brief summary of the key definitions:

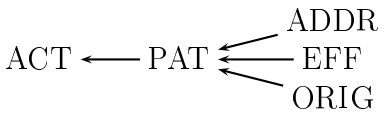
Participants and free modifiers.

FGD defines the distinction between **participants** (actants, inner participants, arguments) and **free modifiers** (adjuncts) of a verb strictly on the tectogrammatical level (and not on the analytic level):

- A participant is characteristic of a verb whereas a free modifier can modify nearly any verb.
- A participant cannot modify a verb twice within a sentence whereas a free modifier can be used repeatedly.

The set of participants is fixed in FGD. The participants are: ACT (actor), PAT (patient), ADDR (addressee), ORIG (origin) and EFF (effect).

Moreover, FGD employs the principle of **shifting**: if a verb has only one participant, it is labelled ACT regardless of its semantic type. Two participants are always ACT and PAT. Starting from three participants, the functors are assigned with respect to the semantics of the modifiers:



Obligatory and optional modifiers.

The distinction between **obligatory** and **optional** modifiers is defined on the t-level only. To summarize the **dialogue test** by Panevová (1980), the modifier is obligatory if its value must be known to the speaker, although the speaker might decide not to express it explicitly on the surface level. This test cannot be performed by a machine so we can only hope for enough indirect evidence in the context or enough examples where none of the obligatory modifications was omitted ("deleted" in some literature).

Valency frame.

A **valency frame** is the set of all **participants** and obligatory **free modifiers** of the verb, i.e. optional free modifiers are not included in the frame. The lexicon of valency frames is needed for all systems aiming at the t-layer annotation in order to re-create t-nodes for obligatory modifiers that were omitted ("deleted") on the surface.

Valency frames, though constructed by observing verb occurrences (and a bit of introspection for the dialogue test), tend to correspond to verb senses (Lopatková and Panevová, 2005)[4]. Performing a word-sense disambiguation task for verbs thus equals to identification of the correct frame of the verb occurrence. In this sense, the **lexical unit** at the t-layer is not just the verb, but also the frame used in the particular instance.

2.2.3 Available Data

This section briefly reviews the properties of available data, i.e. relevant corpora or dictionaries that can be used for automatic extraction of valency frames.

Czech National Corpus (CNC)

The Institute of Czech National Corpus (CNC[5]) provides a collection of balanced and non-balanced corpora of Czech text. In our experiments we used the three versions as listed in Table 2.1.

[4] In the cases where the valency frame is identical for two or more very distinct verb senses, separate frames are introduced for each of the senses, formally differing only in a remark or gloss. Future refinements of the theory, e.g. capturing which lexical classes of modifications are permitted in the slots, might later differentiate such entries.

[5] `http://ucnk.ff.cuni.cz/`

Corpus name	Size (no. words)	Balanced
SYN2006PUB	300 mil.	no
SYN2005	100 mil.	yes
SYN2000	100 mil.	no

Table 2.1: Versions of Czech National Corpus.

VALLEX

VALLEX (Žabokrtský, 2005) is a valency lexicon of Czech verbs. VALLEX uses FGD as its theoretical background and is closely related to the Prague Dependency Treebank (see PDT below). VALLEX is fully manually annotated based on corpus observations and other available Czech lexicons, which poses inevitable limits on the growth rate. On the other hand, manual annotation ensures attaining data of high quality.

The first version of VALLEX 1.0 was publicly released in 2003 and contained over 1,400 verb entries[6]. The set of covered verbs was extended to about 2,500 verb entries in VALLEX 1.5, an internal version released in 2005. For a remark on VALLEX 2.x see Section 2.2.5 below.

	VALLEX 1.0			
	Occ.	[%]	Verb lemmas	[%]
Covered	8.0M	53.7	1,064	3.6
Not covered but frequent	4.1M	27.9	20	0.1
Not covered, infrequent	2.7M	18.3	28,385	96.3
Total	14.8M	100.0	29,469	100.0

	VALLEX 1.5			
	Occ.	[%]	Verb lemmas	[%]
Covered	8.0M	65.6	1,802	6.1
Not covered but frequent	3.5M	23.4	4	0.0
Not covered, infrequent	1.6M	10.9	27,663	93.9
Total	14.8M	100.0	29,469	100.0

Table 2.2: Coverage of VALLEX 1.0 and 1.5 with respect to the Czech National Corpus, SYN2000.

[6]The term **verb entry** refers to a VALLEX entry which distinguishes homographs and reflexive variants of the verb. The term **verb lemma** refers to the infinitive form of the verb, excluding the reflexive particle. See Section 2.2.4 below.

VALLEX 1.5 covers around 66% of verb occurrences; 23% of verb occurrences belong to few frequent auxiliary verbs, esp. *být, bývat (to be)*. (See Table 2.2.) The remaining 10% occurrences belong to verbs with low corpus frequency. The distribution of verbs closely follows Zipf's law and there are about 28k additional verbs needed just to cover our particular corpus. An automated method of lexical extraction would save a lot of labour.

Since the very beginning, VALLEX has been built with computational applications in mind, mostly as a means of ambiguity solving at various levels (lemmatization, tagging, syntactic analysis, sense disambiguation; see (Straňáková-Lopatková and Žabokrtský, 2002) for examples). As a result, VALLEX is sufficiently formalized and the format is very well documented.

VALLEX applications so far, though very significant, are unfortunately still mostly academic:

- In an early stage of the development, VALLEX data was used as a basis for PDT-VALLEX (see below).
- The data format and development technology was reused in the development of VerbaLex (Hlaváčková and Horák, 2006).
- Observations made by VALLEX developers led to refinements in the valency theory (Lopatková and Panevová, 2005).
- VALEVAL data (see below) are used to improve word-sense disambiguation (WSD) methods for Czech verbs (Bojar *et al.*, 2005; Semecký and Podveský, 2006).
- VALLEX was published as a printed lexicon for linguists and Czech speakers in general (Lopatková *et al.*, 2008).
- VALLEX is used when choosing some surface forms in text generation system by Ptáček and Žabokrtský (2006).

VALEVAL

In a lexical sampling task called VALEVAL, the inter-annotator agreement of annotating verb occurrences with VALLEX 1.0 frames was evaluated (Bojar *et al.*, 2005). Despite the fact that VALLEX provides extensive information on distribution contexts (as emphasized by Véronis (2003)), only moderate agreement (in terms of the Cohen's κ statistic (Carletta, 1996)) was achieved. In general, the level 75% of pairwise agreement we achieved is no worse than results for other languages, but a better match is certainly desirable. VALEVAL experiment provided VALLEX developers with a valuable feedback

and a few dozen of serious mistakes were identified in VALLEX entries. A second experiment would have to be carried out to confirm an improvement in inter-annotator agreement.

An independent achievement of VALEVAL are the manual annotations themselves. Cases where our annotators agreed or a final choice was made in a post-processing phase constitute what we call "Golden VALEVAL" corpus. Golden VALEVAL contains 108 verbs in 7804 sentences (72±26 sentences per verb), annotated with a single VALLEX frame that was used in the sentence.

Prague Dependency Treebank (PDT) and PDT-VALLEX

Prague Dependency Treebank (PDT, Hajič *et al.* (2006)) is a corpus of Czech texts extensively manually annotated on the m-, a- and t-layers. Moreover, each occurrence of a verb and some nouns and adjectives are labelled with a pointer to the valency frame used in that particular sentence.

PDT-VALLEX (Hajič *et al.*, 2003) is a valency lexicon of Czech verbs and some nouns and adjectives that accompanies the Prague Dependency Treebank (PDT). While based on the same theoretical background as VALLEX, PDT-VALLEX is tailored to the corpus. In other words, PDT-VALLEX contains only frames that were actually observed in sentences in PDT.

Similarly to VALLEX, PDT-VALLEX suffers from the problem with too specific frame entries. For instance, the verb *zakotvovat* (*to anchor*), is equipped with two distinct frames: ACT(1) PAT(4) DIR3(*) (*to anchor sth to sth*) and ACT(1) PAT(4) LOC(*) (*to anchor sth somewhere*). Each occurrence of *zakotvovat* is annotated with a single frame reference, even in cases where there was no DIR3 and no LOC observed in the sentence (e.g. t-cmpr9410-001-p4s2w11). The annotator's decision between these two frames is then based on his or her detailed understanding of the sentence or simply random, if no clear hints are provided in a wide context. Two annotators are likely to disagree in the frame chosen, although they would agree on a less detailed frame.

As Mikulová *et al.* (2006) mentions (Section 5.2.3.1.1. of the Czech version or 6.2.3.1.1. of the English version), there are cases where the decision is well motivated and allows us to distinguish between concrete, abstract or idiomatic meaning of the verb. At the same time, it is mentioned that the annotation consistence is quite low in this respect (not giving any more specific estimations).

Other Related Resources

There are far too many related projects of computational lexicography. To name a few, we acknowledge:

for Czech VerbaLex (Hlaváčková and Horák, 2006), Czech Syntactic Lexicon (Skoumalová, 2001) and their surface-syntactic predecessor Brief (Pala and Ševeček, 1997),

for English FrameNet (Baker *et al.*, 1998; Fillmore *et al.*, 2001; Fillmore, 2002), PropBank (Kingsbury *et al.*, 2002), Lexical Conceptual Structure (Jackendoff, 1990; Dorr and Mari, 1996), VerbNet (Kipper *et al.*, 2000; Kipper-Schuler, 2005) and EngValLex (Cinková, 2006).

A closely related resource is the lexical database WordNet (Fellbaum, 1998) and its European (Vossen, 1998) and Czech (Pala and Smrž, 2004) versions.

Please consult e.g. Žabokrtský *et al.* (2002) or Lopatková (2003) for a review of some of the projects.

2.2.4 Structure of VALLEX 1.0, 1.5 and PDT-VALLEX

At the topmost level, VALLEX is a list of **verb entries**[7], see Figure 2.2 for an example of two of them. The verb is characterized by its **headword lemma** (including a reflexive particle *se* or *si*, if appropriate) or several spelling variants of the headword lemma equipped with verb aspect (perfective, imperfective, biaspectual). Every verb entry includes one or more **valency frames** of the verb roughly corresponding to its senses. Every valency frame consists of a set of **valency slots** characterizing complementations of the verb. Each slot describes the type of the syntactico-semantic relation between the verb and its complementation (by means of a **tectogrammatical functor**, such as Actor ACT, Patient PAT, Direction $DIR1$; see FGD) as well as all allowed surface realizations (**morphemic forms**) of the verb complementation (e.g. the required preposition and case or the subordinating conjunction for dependent clauses).[8] The slot also indicates obligatoriness of the complementation. Each frame is equipped with a short gloss and an example in order to help human annotators to distinguish among the frames. Aspectual

[7] Due to the lack of space we can only briefly summarize the key terms. Please consult Žabokrtský and Lopatková (2004) for a detailed description, examples and explanation of all the terms not defined here.

[8] In the cases where any morphemic form typical for a functor can be used to realize the slot, the set of morphemic forms is left empty.

odpovídat (imperfective)
1 odpovídat$_1$ ∼ odvětit (answer; respond)
- frame: ACT^{obl}_1 ADDR^{obl}_3 $\text{PAT}^{opt}_{na+4,4}$ $\text{EFF}^{obl}_{4,aby,aǐ,zda,že}$ MANN^{typ}
- example: *odpovídal mu na jeho dotaz pravdu / že ... (he responded to his question truthfully / that ...)*
- asp.counterpart: odpovědět$_1$ pf.
- class: communication

2 odpovídat$_2$ ∼ reagovat (react)
- frame: ACT^{obl}_1 PAT^{obl}_{na+4} MEANS^{typ}_7
- example: *pokožka odpovídala na včelí bodnutí zarudnutím (the skin reacted to a bee sting by turning red)*
- asp.counterpart: odpovědět$_2$ pf.

3 odpovídat$_3$ ∼ mít odpovědnost (be responsible)
- frame: $\text{ACT}^{obl}_1\text{ADDR}^{obl}_3\text{PAT}^{opt}_{za+4}\text{MEANS}^{typ}_7$
- example: *odpovídá za své děti; odpovídá za ztrátu svým majetkem (she is responsible for her kids)*

4 odpovídat$_4$ ∼ být ve shodě (match)
- frame: $\text{ACT}^{obl}_{1,že}\text{PAT}^{obl}_3\text{REG}^{typ}_7$
- example: *řešení odpovídá svými vlastnostmi požadavkům (the solution matches the requirements)*

odpovídat se (imperfective)
1 odpovídat se$_1$ ∼ být zodpovědný (be responsible)
- frame: $\text{ACT}^{obl}_1\text{ADDR}^{obl}_3\text{PAT}^{obl}_{z+2}$
- example: *odpovídá se ze ztrát (he answers for the losses)*

Figure 2.2: Two VALLEX 1.0 entries for the verb lemma *odpovídat (answer, match)*.

counterparts of the verb are not assigned to the verb entry as a whole but to the individual frames: a frame of a verb contains a link to a frame of its aspectual counterpart, if appropriate.

The operational criteria on when to create a new frame entry of a verb are described in Lopatková and Panevová (2005). Roughly speaking, a frame entry corresponds to a "sense" of the verb based primarily on (deep) syntactic observations.

We use the term **verb lemma** to denote the infinitive of the verb, excluding a possible reflexive particle and homograph distinction, e.g. *odpovídat* is the verb lemma for the verbs *odpovídat* and *odpovídat se*. The verb lemma is determined by the morphological analysis of a text.

2.2.5 Frame Alternations and VALLEX 2.x

It should be noted that the slots and sets of allowed morphemic forms listed in VALLEX describe only the "canonical" realizations of the verb. Each of the

frames can undergo one of a small set of pre-defined **frame alternations**.

For instance, if the frame contains an ACT in nominative and a PAT in accusative, we have to alter the frame for occurrences of the verb in passive—the PAT becomes expressed by a nominative and the ACT by an instrumental.

Empirical data for Czech are available in PDT 2.0 where each verb occurrence is labelled with a frame identifier from PDT-VALLEX. By comparing immediate dependents of the verb in the tree with slots of the respective frame, we can see which alternation (if any) was performed in the sentence.

VALLEX versions 2.0 (Lopatková *et al.*, 2006a) and 2.5 (Lopatková *et al.*, 2008) again extend the set of verbs and frames covered. Inspired by the alternation model by Levin (1993), they adopt the idea of alternations as a part of the core design and significantly change the structure of the lexicon (Lopatková *et al.*, 2006b). Until there are some corpus examples annotated with VALLEX 2.x frames, we cannot use this source for most methods of frame extraction, leaving the additional problem of alternation learning aside.

2.2.6 Motivation for Automated Lexical Acquisition

As mentioned in Section 2.2.3, VALLEX 1.5 covers about 66% of verb tokens but only 6% of verb types in CNC. Due to the law of diminishing returns, it is less and less economical to add entries for new verbs manually. Moreover, it is believed that less frequent verbs have a simpler structure of frames. See Stevenson (2003) who discusses the observation by Zipf (1945) and other experiments confirming that the observation is not just an artifact of fewer corpus instances available to lexicographers. In total, we could hope that for most of the remaining verbs, frame information can be derived automatically given enough corpus evidence (and the frames already defined for other verbs) and that a lot of lexicographic labour can be thus saved.

From a different perspective, automatically finding examples of VALLEX entries in a large corpus would allow to:

- add frequencies to VALLEX (to support statistically-aware applications of the lexicon),
- add selectional restrictions (to support more semantically-informed applications or to improve the sense-discriminating power of VALLEX in a way similar to VerbaLex),
- cross-check of VALLEX entries (to test whether all corpus samples of a verb identified automatically to bear the same VALLEX frame are

indeed confirmed to be instances of a single verb sense by a native speaker).

2.3 Simplified Formalization of VALLEX Frames

Section 2.2.4 introduced the formal structure of VALLEX and PDT-VALLEX. Both of the dictionaries reflect the fact that at the t-layer, the t-lemma includes the reflexive particle whenever appropriate. On the other hand, most of our learning methods, as described in Section 2.7 below, do not expect to start with a t-annotation at hand. Anywhere below the t-layer, it is not easy to identify the reflexivity of a verb for several reasons: (1) the reflexive particle does not need to appear next to the verb, (2) it is homomorphic with the vocalized version of a Czech preposition, and (3) it can represent the value of a regular frame slot (e.g. PAT or ADDR), indicate passivization as well as purely syntactically complement the verb lemma (reflexiva tantum). Although (1) and (2) can be recognized at a high precision, (3) has probably not been studied yet. Our preliminary experiments (Figure 2.3) indicate that the verb lemma plays a very significant role in identifying the reflexivity of the verb. If the verb lemma is not known, the decision procedure makes a wrong guess in 9 to 16% of cases. Knowing the verb lemma helps to reduce the error by 5 to 10% absolute.

For the purpose of our learning task, we simplify the structure of VALLEX as follows.

While VALLEX and PDT-VALLEX provide us with the mapping:

$$\begin{pmatrix} \text{verb lemma} \\ \text{index distinguishing homonyms} \\ \text{reflexive particle} \end{pmatrix} \rightarrow \text{frame}$$

we treat the valency lexicon as the mapping:

$$\text{verb lemma} \rightarrow \begin{pmatrix} \text{reflexive particle} \\ \text{frame} \end{pmatrix}$$

Apart from index distinguishing homonyms, it is easy to convert one format into the other one and vice versa.

VALLEX and PDT-VALLEX also differ in formal details of morphemic forms. For instance, PDT-VALLEX uses a nested structure to describe requirements on the presence and attributes of a set of a-nodes (e.g. a preposition and a noun that form a part of a phraseme) while a simple surface string of words is used in VALLEX. Again, we simplify the format and treat all morphemic forms as atomic units.

Features Used	Average Error [%]
Verb lemma, Refl seen close	4.83 ± 0.89
Verb lemma+tag, Refl seen	4.96 ± 0.86
Verb lemma, Refl seen	5.38 ± 1.30
Verb tag, Refl seen close	9.69 ± 1.37
Refl seen close	9.71 ± 1.23
Verb tag, Refl seen	16.06 ± 1.34
Refl seen	16.08 ± 1.69

- Training data: 7000 occurrences of verbs in golden VALEVAL data.
- Learning goal: Decide whether the VALLEX entry assigned to each verb occurrence has the reflexive particle *se*, *si* or is not reflexive at all.
- Procedure: Decision trees (C4.5) using a subset of the following features:
  - Verb lemma – the lemma of the verb in question,
  - Verb tag – individual features for each morphological category of the verb occurrence,
  - Refl seen – features describing the presence and morphological case of the reflexive particle *se*/*si* before or after the verb in question,
  - Refl seen close – like "Refl seen" but only particles between the verb in question and another verb in the sentence are considered.
- Evaluation: Average error over 4- to 10-fold evaluation.

Figure 2.3: Average error of identifying the type of reflexivity (non-reflexive/*se*/*si*) of a verb occurrence.

To sum up, we define **frame** $F = (Refl, Slots)$ as a tuple where:

- $Refl \in \{void, se, si\}$ is a ternary feature describing the reflexivity of a verb in the meaning of frame F.

- $Slots : Functor \mapsto (Oblig, \wp(MorphemicForms))$ is a function assigning an obligatoriness flag $Oblig \in \{obligatory, optional\}$ and a set of allowed $MorphemicForms$ to any $Functor \in \{ACT, PAT, \dots\}$ mentioned by the frame.
  The function $Slots$ is not total, an undefined mapping for a functor indicates there is no slot with such functor in the frame. Note that this formalization does not allow any frame to contain several slots sharing the same value of $Functor$.
  $MorphemicForms$ is a set of atomic values, each describing one of all possible morphemic realizations of a modifier. Unlike VALLEX, we never leave $MorphemicForms$ empty. In the cases where all typical morphemic forms are appropriate, we explicitly fill the set with observed verb modifiers and their functors in PDT 2.0.

2.4 Types of Data Sources

In the following, we use this notation:

$\mathbb{V} = (V, F, L)$ denotes a valency lexicon, where V is the set of verb lemmas of all verbs contained in the lexicon, F is the set of all frames defined in the lexicon and $L\colon V \to \mathcal{P}(F)$ is the actual mapping providing each of the verbs $v \in V$ with a set of frames from F.

In our experiments we can use VALLEX 1.0, VALEVAL or PDT-VALLEX as our $\mathbb{V}$. The difference between VALLEX 1.0 and VALEVAL is both in the set of verbs V covered and the set of known frames F: $\mathbb{V}_{\text{VALEVAL}}$ includes only the frames that were actually observed in golden VALEVAL annotation.

For conciseness, we use dot notation to access individual components of the structure. For instance, we write $\mathbb{V}_{\text{VALLEX 1.0}}.V$ to denote the set of verb lemmas contained in VALLEX 1.0.

$\mathbb{C} = (S, W)$ denotes a corpus of sentences $S = \{s_i \mid s_i \text{ is a Czech sentence}\}$. Although the order of the sentences in $\mathbb{C}$ is not important, we assume an arbitrary fixed order and use $\mathbb{C}.W$ to refer to the sequence of all running words in the corpus. $\mathbb{C}.W_i$ denotes the i^{th} word in the corpus.

We use a superscript on $\mathbb{C}$ to indicate the deepest layer (*m*orphological, *a*nalytical or *t*ectogrammatical) of annotation available for sentences in $\mathbb{C}$. For instance, $\mathbb{C}^t$ refers to a corpus with all layers up to the tectogrammatical analysis.

For $\mathbb{C}^{\geq m}$ (i.e. a corpus with at least morphological annotation) and a verb lemma v, we define the function $\text{find}(v, \mathbb{C}^{\geq m})$ to return all occurrences (indices to $\mathbb{C}.W$) of the verb with the verb lemma v. The corpus manager Manatee (Rychlý and Smrž, 2004) is a very efficient implementation of the function $\text{find}(\cdot, \cdot)$.

In our experiments, we can use PDT 2.0, CNC or VALEVAL as our $\mathbb{C}$, PDT 2.0 being the only corpus with manual annotation on all layers.

$\widehat{\mathbb{C}\mathbb{V}} = (\mathbb{C}, \mathbb{V}, O, A)$ denotes a corpus $\mathbb{C}$ with all occurrences $O \subset \mathbb{C}.W$ of verbs $v \in \mathbb{V}.V$ annotated with the frame used by the speaker in the particular sentence. The function $A\colon O \to \mathbb{V}.F$ formally represents the annotation.

VALEVAL and the combination of PDT 2.0 with PDT-VALLEX are two examples of $\widehat{\mathbb{C}\mathbb{V}}$ we have at hand.

2.5 Learning Task and Evaluation Metrics

Our learning task is to provide a test verb lemma v_t with the set of all valid frames F_{v_t}. For the purpose of evaluation of our learning methods, we always choose v_t from a known dictionary $\mathbb{V}$. This allows us to compare F_{v_t} to the manually assigned set of frames $\mathbb{V}.L(v_t)$. We use the abbreviation "golden frame set" (G) to refer to $\mathbb{V}.L(v_t)$ and "hypothesized frame set" (H) to refer to F_{v_t}.

Given a test verb lemma v_t, how should we evaluate the quality of a hypothesized frame set H given the golden frame set G?

Methods of frame extraction are usually evaluated in terms of precision (P) and recall (R) of either frames as wholes or of individual frame elements (slots). See esp. Korhonen (2002) for a survey and comparison of several approaches using precision and recall.

Note however that depending on the application, different metrics may provide different predictions. As pointed out by Zhang *et al.* (2007), an HPSG parser benefits more from lexical acquisition methods of a high recall, not of a high F-score (harmonic mean of precision and recall).

For the richly structured VALLEX-like verb entries, precision and recall suffer from some limitations:

- frame-based P and R are too rough and penalize the smallest mistake in frame with the same cost as omission of the whole frame,
- slot-based P and R are too fine-grained and cannot account for the complexity of verb entry in terms of various combinations of slots.

To provide a simple means of comparison, we report on the frame-based **precision** and **recall**:

$$\mathrm{P}(H,G) = \frac{|H \cap G|}{|H|} \tag{2.1}$$

$$\mathrm{R}(H,G) = \frac{|H \cap G|}{|G|} \tag{2.2}$$

However, our main focus will lie in a novel metric, **frame edit distance** and **verb entry similarity** as defined below.

2.5.1 Frame Edit Distance and Verb Entry Similarity

In Benešová and Bojar (2006), we define the **frame edit distance** (FED) as the minimum number of edit operations (insert, delete, replace) necessary to

convert a hypothesized frame into the correct frame. The metric described in this section is a refined version that better matches our simplified definition of frames (see Section 2.3).

For the time being, we assign equal costs to all basic editing operations (fixing the reflexive particle of the frame or fixing obligatoriness flag, adding to or removing allowed morphemic forms from a slot). However, the functor of a slot is considered as fixed. In order to change the functor, one pays for a complete destruction of the wrong slot and a complete construction of the correct slot. We consider charging more for slot destruction than for slot construction in future versions of the metric because we prefer methods that undergenerate and produce safer frames to methods that suggest unjustified frames.

In order to evaluate the match between a whole golden frame set G as contained in the lexicon and a frame set H hypothesized by an automatic frame-generation procedure, we need to extend FED to compare whole sets of frames (i.e. verb entries in the lexicon). We call this extension **entry similarity** (ES) and define it as follows:

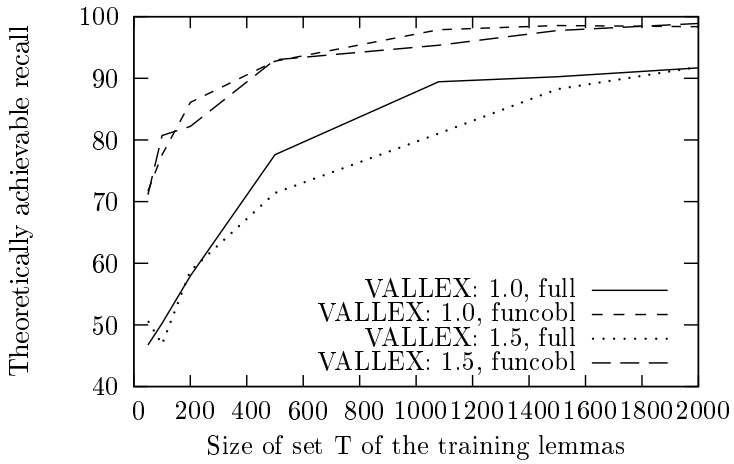
$$ES(H,G) = 1 - \frac{\min FED(G,H)}{FED(G,\emptyset)+FED(H,\emptyset)}$$

G denotes the set golden verb entries of the verb lemma, H denotes the hypothesized entries and $\emptyset$ stands for a blank verb entry (containing no frames). $\min FED(G,H)$ is the minimum edit distance necessary to convert the frames in H into the frames in G, including the possible generation of missing frames or destruction of superfluous frames.

ES attempts to capture how much of lexicographic labour has been saved thanks to the contribution of the automatic frame-generation procedure. If the system did not suggest anything ($H = \emptyset$), the ES is 0%. If the system suggested exactly all the golden frames ($H = G$ and thus $FED(G,H) = 0$), the ES achieves 100%. With this explanation in mind, we will use the term **expected saving** (ES) as a synonym to "entry similarity".

It is important to note that the suggested verb entry or frame can sometimes contain some additional information that should be included in the golden frameset, but it is not. We perform no special treatment for this situation and regard the additional information as a mistake of the learning algorithm, although it is in fact a mistake or omission of the authors of the lexicon.[9]

[9]Thanks to the VALEVAL experiment (Bojar *et al.*, 2005), we know that in a sample of 100 verb lemmas of verbs, annotators observed about 57 missing frames, 6 inappropriately joined or split frames and 12 superfluous frames. Similarly, errors were observed in VALLEX frame entries: in 16 cases a functor was chosen incorrectly or the slot was missing and in 12 cases, the morphemic form was incorrect or missing.



- Input data: A valency lexicon $\mathbb{V}$, the set of verb lemmas $\mathbb{V}.V$ partitioned into a training T and evaluation E sets, $T \cap E = \emptyset$.
- Procedure:
  1. Collect all full frames of training lemmas T into a set K of known frames.
  2. For an unseen verb lemma $l \in E$ with golden frames G, evaluate achievable recall R as the ratio of known frames among golden frames: $R = \frac{|G \cap K|}{|G|}$.

Figure 2.4: Upper bound on full frame recall, i.e. frames are not decomposed into slots.

2.5.2 Achievable Recall without Frame Decomposition

Let us first briefly examine the upper bound on recall of a baseline algorithm. Given VALLEX frames for some known verb lemmas, the most simple approach to learning entries for new verbs is to reuse known frames as wholes.

Figure 2.4 summarizes the baseline algorithm and its upper bound on recall with respect to the number of training verb lemmas. As we see, if frames are treated as full frames (i.e. a set of functors including the obligatoriness flag and the set of allowed morphemic forms), the theoretically achievable recall of any learning algorithm that uses known frames as wholes is about $92 \pm 3\%$. If only the functors and the obligatoriness flags (labelled "funcobl") are taken into account when learning and proposing frames, current VALLEX size proves to suffice: the achievable recall reaches $99 \pm 1\%$. However as the learning curve indicates, this had not been the case until about 1 500 verb lemmas were covered in VALLEX.

It is worth mentioning that the number of frames covered in VALLEX is still growing, and that the growth is observed even in the least detailed

VALLEX version:	1.0	1.5
Everything incl. comments	3871	6506
Functors+Oblig+Forms	1142	1711
Functors+Oblig+Forms, ignoring order of Forms	1141	1705
Functors+Oblig+Forms, ignoring frames with a phraseme	1040	1472
Functors+Oblig	427	574
Functors	330	444

Table 2.3: The number of unique frames defined in VALLEX 1.0 and 1.5 depending on how detailed information is used to distinguish frames. Frames are collected from all verb entries.

definition of frames. Table 2.3 displays the number of frames (collected from all verb entries) in VALLEX 1.0 and 1.5. If two frames are counted as different whenever any attribute differs, VALLEX 1.0 contains about 3 900 and VALLEX 1.5 about 6 500 frames. The other extreme is to consider frames as sets of functors only, ignoring morphemic forms and obligatoriness. There are about 330 of these crude frames in VALLEX 1.0 and about 440 in VALLEX 1.5. This indicates that the set of crude frames is by no means complete yet and that new frames should be expected in more contemporary Czech data.

To sum up, methods that "reuse" known frames as wholes will face a significant limit on achievable recall unless they reduce the notion of frame to the set of functors.

2.6 Lexicographic Process

The aim of this chapter is to automate the creation of VALLEX entries, i.e. to model the work of a lexicographer.

Atkins (1993), Calzolari *et al.* (2001) or Stevenson (2003) delimit two stages in the process of deriving lexical entries:

Analysis: Collecting corpus evidence. The risk connected with this task is that if there is no underlying theory or no direct application targeted, important features might remain neglected. This can effectively block some future applications of the lexicon.

Synthesis: Creating the lexicon entry. The most apparent difficulty is to make entries consistent throughout the whole lexicon. A central question is what to include in the lexicon and what to ignore (which

entries as well as which details within the entries). Here, the only objective criterion is usually the frequency, however for FGD, Panevová (1980) offers a valuable insight by introducing the so-called "dialogue test" to identify obligatory slots (which should thus be included in the dictionary).

A similar delimitation of our task into the two subtasks can be drawn:

- word sense discrimination, i.e. providing verb occurrences with a sense or frame label,
- grouping verb occurrences with the same frame and constructing the formal frame description for the whole group.

Following the delimitation, we now propose three direct methods (Section 2.7) and an indirect one (Section 2.9) for automatic frame suggestion.

2.7 Direct Methods of Learning VALLEX Frames

One could think of many ways of how to automatically generate valency frames for new verbs. This section is devoted to the description and comparison of three rather direct methods we developed. The methods are: WFD (Word-Frame Disambiguation), DSD (Deep Syntactic Distance), and Decomp (Learning frames by decomposition). An additional method, PatternSearch (Searching for patterns indicating a frame), is described in Section 2.9.

One of the key aspect of each learning method is whether it treats verb frames as opaque units and is thus limited by the upper bound described in Section 2.5.2, or whether the method is in principle capable of constructing completely new types of frames if the data seem to suggest it. The methods WFD, DSD and PatternSearch do not consider internal structure of verb frames at all. Decomp is in principle able to construct new types of frames.

Using the notation as defined in Section 2.4, we can formally describe the type of training data necessary to learn frames F for a given test verb lemma v_t:

$\widehat{\mathbb{CV}}$ **and** $\mathbb{C}'$ where $v_t \notin \mathbb{V}.V$ and $\text{find}(v_t, \mathbb{C}') \neq \emptyset$.

When we have a corpus annotated with frames $\widehat{\mathbb{CV}}$ (but no examples for the test verb v_t) and a corpus $\mathbb{C}'$ with no explicit annotation of verbal frames but with some examples of usage of v_t, we can use the methods WFD, DSD and Decomp, as described below.

$\mathbb{V}$ **and** $\mathbb{C}$ where $v_t \notin \mathbb{V}.V$ and $\text{find}(v_t, \mathbb{C}) \neq \emptyset$.

When we have just a seed lexicon $\mathbb{V}$ (not covering the verb v_t) and a corpus $\mathbb{C}$ containing some samples of v_t usage, we can use PatternSearch.

2.7.1 Word-Frame Disambiguation (WFD)

Semecký (2007) describes a system for supervised word-frame disambiguation (WFD). For a training corpus annotated with verb frames $\widehat{\mathbb{CV}}$ and a given verb lemma v (where $\text{find}(v, \mathbb{C}) \neq \emptyset$), the system learns to predict the frame $f \in \mathbb{V}.A(v)$ for a test sentence s_t where no annotation is available. At the minimum, the corpus has to be analysed at the morphological layer ($\mathbb{C}^m$) but significant improvement is gained if analytical trees are available ($\mathbb{C}^a$). The system converts each occurrence of the verb in the training corpus, $o \in \text{find}(v, \mathbb{C})$, into a vector of features describing morphological and surface-syntactic properties of the verb and its neighbourhood. A similar vector of features is extracted for the verb v from the test sentence s_t. Comparing the test vector with the training vectors using one of several machine-learning methods (various vector distance metrics), the system suggests the most likely frame to the verb v occurring in s_t. The system treats verb frames as opaque symbols with no internal structure and achieves accuracy of nearly 80%.

We can reuse the idea to predict the set of frames F for a test verb v_t. We first train a chosen classifier on training examples for all known verbs ignoring their lemmas (i.e. pretending that all annotated verb occurrences in $\widehat{\mathbb{CV}}$ belong to the same verb, namely v_t). Given a set of real examples of v_t, i.e. $\mathbb{C}'$ annotated at the same layer as $\mathbb{C}$, the classifier will suggest the most likely frame from all occurrences $o \in \text{find}(v_t, \mathbb{C}')$. In our experiments, we used MaxEnt classifier by Zhang (2004) but any other classifier such as decision trees or support vector machines could be used. A very promising approach would be to use some of discriminative learning methods (e.g. averaged perceptron, Collins and Roark (2004)) that learn to predict the most likely frame by contrasting it to other candidates whereas traditional methods consider each candidate independently estimating its chance to win.

Simply collecting all frames suggested for various examples of the given verb will give us an estimate which frames should we assign to the verb. Formally:

$$F_{v_t} := \{f \mid \exists o \in \text{find}(v_t, \mathbb{C}') \text{ s.t. WFD system assigned } f \text{ to } o\} \qquad (2.3)$$

Summary of WFD:

- frames opaque

- input: v_t; output: F_{v_t}
- required data:
  - $\widehat{\mathbb{C}\mathbb{V}}$ where $v_t \notin \mathbb{V}.V$, and
  - $\mathbb{C}'^{m \text{ or } a}$ where $\text{find}(v_t, \mathbb{C}') \neq \emptyset$

2.7.2 Deep Syntactic Distance (DSD)

One of the drawbacks of WSD described in the previous section is the lack of a direct link between the theory of valency and the model predicting one of the frames for a given verb occurrence. In order to address this issue, we propose a novel metric called Deep Syntactic Distance (DSD). DSD is directly motivated by valency theory as expressed in guidelines for VALLEX authors: for each verb occurrence, the underlying deep syntactic analysis of the sentence is considered.

Given two occurrences o_1 and o_2 of a verb (or two distinct verbs) in a corpus $\mathbb{C}^a$ annotated at the a-layer, $DSD(o_1, o_2)$ estimates how difficult it is to believe that the underlying verb frame used in o_1 is the same as the frame used in o_2. DSD considers the surface realization of each analytical dependent son_j of o_i and the likelihood $p(F|son_j)$ of that particular form to express a tectogrammatical functor F. The dependents of o_1 and o_2 are paired assuming a common functor F for both of them. DSD is the minimum cost (highest likelihood) over all possible pairings π, optionally with a penalty for unpaired dependents in case the verb occurences have a different number of sons.

$$DSD(o_1, o_2) \quad := \quad \min_{p \in \pi(o_1, o_2)} \sum_{(son_1, son_2, F) \in p} 1 \; - \; p(F|son_1) \; \cdot \; p(F|son_2) \quad (2.4)$$

The application of DSD to our task (i.e. providing a test verb v_t with a hypothesized frameset F_{v_t}) is in essence identical to the well-known nearest neighbours (NN) machine-learning method: given a training corpus annotated with verb frames $\widehat{\mathbb{C}^a\mathbb{V}}$ and a sample unlabelled observation o_t in a sentence containing v_t, we evaluate $DSD(o, o_t)$ for all labelled observations $o \in (\widehat{\mathbb{C}^a\mathbb{V}}).O$. The test observation o_t is assigned the same frame as the winning o in the labelled data has. Similarly to the nearest neighbours method, various modifications of the voting scheme (e.g. k-NN or k-NN weighted by the distance) might be considered.

Given a corpus $\mathbb{C}'$ of example sentences of v_t, each sentence in $\mathbb{C}'$ will contribute with a single suggested frame f_{best}. We collect all suggested frames

and return them as the hypothesized frameset F_{v_t}. Formally:

$$F_{v_t} = \left\{ f_{best} \;\middle|\; \exists o_t \in \text{find}(v_t, \mathbb{C}') \text{ s.t. } \begin{array}{ll} o_{best} & = \text{argmin}_{o \in (\widehat{\mathbb{C}^a\mathbb{V}}).O} \, DSD(o, o_t) \\ f_{best} & = (\widehat{\mathbb{C}^a\mathbb{V}}).A(o_{best}) \end{array} \right\} \tag{2.5}$$

Another possible application of DSD is to help in consistence checking of manual annotation in a $\widehat{\mathbb{C}^a\mathbb{V}}$. Given a verb $v \in \mathbb{V}.V$ and all its occurrences $O = \text{find}(v, \mathbb{C})$, we can evaluate $DSD(o_1, o_2)$ for each pair $(o_1, o_2) \in O \times O$. All cases where $DSD(o_1, o_2)$ is low but o_1 and o_2 have a different frame assigned in the annotation $(\widehat{\mathbb{C}^a\mathbb{V}}).A$ as well as all cases with $DSD(o_1, o_2)$ high but identical frames assigned, i.e. $A(o_1) = A(o_2)$, should be manually checked. Assuming DSD estimates are correct, the discrepancy between DSD and manual annotation can suggest an error in the annotation or at least demonstrate that the differences between frames $f_1 = A(o_1)$ and $f_2 = A(o_2)$ are maybe too subtle to be noticed based on purely syntactic information in the context of the verb.

Summary of DSD:

- frames opaque
- input: v_t; output: F_{v_t}
- required data:
  - $\widehat{\mathbb{C}^a\mathbb{V}}$ where $v_t \notin \mathbb{V}.V$, and
  - $\mathbb{C}'^a$ where $\text{find}(v_t, \mathbb{C}') \neq \emptyset$

2.7.3 Learning Frames by Decomposition (Decomp)

Both WFD and DSD assumed frames are opaque units and relied on a similarity between verb occurrences. We now propose a method called Decomp that decomposes frames into basic building blocks ("frame components") and suggests frames for unseen occurrences by combining some of the frame components.

Given a labelled training corpus $\widehat{\mathbb{C}\mathbb{V}}$ and a test verb v_t not present in $\widehat{\mathbb{C}\mathbb{V}}$ but present in a separate unlabelled corpus $\mathbb{C}'$, we formulate the goal of providing v_t with a set of frames F_{v_t} as a multi-class classification task using a suitable set E of "frame components", each describing a particular aspect of the frame. For instance, the frame components "refl-is-*se*", "ACT-obligatory", "ACT-can-be-nominative", ... could be used to describe the frame "se ACT.obl.nom ...".

Two additional functions are needed: decomp: $\mathbb{V}.F \to \mathcal{P}(E)$ to decompose frame into atomic pieces and recomb: $\mathcal{P}(E) \to$ frame to recombine them again.

Algorithm 1 Suggesting frames by decomposition (Decomp).

1. Prepare training data for the multi-class classifier:
2. For each occurrence o of each training verb v in $\widehat{\mathbb{CV}}$
3. Extract "surface features" from the neighbourhood of o, as in WSD.
4. Construct "deep features" from the frame assigned to o:
   decomp$((\widehat{\mathbb{C}^a\mathbb{V}}).A(o))$.
5. Enter the pair (surface features, deep features) as
   a training instance to the classifier.
6. Suggest frame F for an occurrence o_t of a test verb v_t:
7. Use the multi-class classifier to predict the set of deep features
   $D \in \mathcal{P}(E)$ for o_t based on its observed surface features.
8. Assign the recombined frame to o_t: $F = \text{recomb}(D)$.

The multi-class classification is employed in the process as described in Alg. 1. In our particular case, we use independent binary classifiers instead of a single multi-class classifier. For each deep feature (i.e. frame component) independently, we train to predict "present" or "not-present" based on the full observed context. It is up to the machine learner to identify if any surface features predict that particular frame component more reliably. In our experiments, we used MaxEnt classifier by Zhang (2004) but any other classifier such as decision trees (Quinlan, 1986, 2002) or support vector machines (Cortes and Vapnik, 1995) could be used.

We cannot assume that the learner would be able to suggest frame components of morphemic forms not realised in a particular sentence. Instead of simply collecting all suggested frames, we merge$(\cdot)$ them based on the "skeleton" of obligatory slots. For instance, if the frame *ACT.obl.nom PAT.obl.acc* was proposed for one verb occurrence and *ACT.obl PAT.obl.na+acc* for another one, we include a single merged frame in the final suggested frame set: *ACT.obl.nom PAT.obl.{acc,na+acc}*.

Formally:

$$F_{v_t} := \text{merge}\left(\left\{ f \;\middle|\; \begin{array}{l} \exists o \in \text{find}(v_t, \mathbb{C}') \\ \quad \text{s.t. Decomp system assigned } f \text{ to } o \end{array} \right\}\right) \tag{2.6}$$

Summary of Decomp:

- frames decomposed and recombined
- input: v_t; output: F_{v_t}
- required data:
  - $\widehat{\mathbb{CV}}$ where $v_t \notin \mathbb{V}.V$, and
  - $\mathbb{C}'^{m \text{ or } a}$ where $\text{find}(v_t, \mathbb{C}') \neq \emptyset$

2.7.4 Post-processing of Suggested Framesets

As a consequence of the definition, one of the key properties judged by ES is the number of frames suggested. For every missing or superfluous frame, ES charges a significant penalty based primarily on the number of slots of all unmatched frames.

Certainly, one could try to automatically predict the number of frames needed for each verb on the basis of the frequency of the verb, some measure of diversity of syntactic properties or the number of translation equivalents in a translation dictionary or a parallel corpus. (Frequency alone is a reasonable but not sufficient predictor, there are frequent verbs with relatively few frames.)

We leave this for further investigation and instead use two methods that modify a suggested frame set to *match the expected* number of frames for each verb, thus allowing the methods to peek at the test data partly:

SIMPLE If the number of expected frames is higher than the number of suggested frames, additional baseline frames (*ACT.obl.nom PAT.obl.acc*) are added to reach the expected number of frames. If the number of expected frames is lower than the number of suggested frames, only the frames with high support are added. (The definition of support is straightforward: for WFD and DSD it is the number of verb occurrences that were assigned that particular frame. For Decomp, the latter case never happens, as Decomp always suggests fewer frames than expected, see the discussion below)

CLUST If the number of expected frames is higher than the number of suggested frames, we use the same approach as SIMPLE: add baseline frames up to the expected frame count. If the number of expected frames is lower, we use automatic clustering and centroid selection to choose a set of the expected size containing the most representative frames. The objects that enter our clustering algorithm are frames suggested by individual verb occurrences. We compute the frame edit distance (FED, Section 2.5.1) between every pair of frame occurrences and use the clustering toolkit by Karypis (2003) to cluster the occurrences to the expected number of frame groups. Groups are chosen to maximize distances between the groups and minimize distances within the groups. For each of the groups we then choose a representative (a "centroid"): the frame with the lowest distance to all other members in the group.

Method	Options	Fit Frame Count	Avg ES	Avg Prec	Avg Rec
WFD		no	21.4±4.7	4.1±1.4	26.9±11.1
DSD	noPenalize	no	25.6±3.1	20.5±14.0	3.8±2.8
Baseline	1×ACT-PAT	no	27.7±4.9	45.7±21.9	9.7±6.8
DSD	noPenalize, ReqObl	no	33.9±5.6	1.5±3.1	3.4±6.9
DSD	Penalize	no	38.5±8.5	6.0±5.2	13.7±11.0
Baseline	2×ACT-PAT	no	38.8±4.9	22.8±11.0	9.7±6.8
Decomp		no	43.0±1.5	4.2±2.1	4.3±2.0
DSD	Penalize, ReqObl	no	43.1±8.1	7.9±6.5	14.2±11.3
Baseline	3×ACT-PAT	no	43.7±3.6	15.2±7.3	9.7±6.8
Baseline	avg×ACT-PAT	no	45.3±4.6	5.9±2.7	9.7±6.8
Baseline	4×ACT-PAT	no	46.8±3.2	11.4±5.5	9.7±6.8
DSD	Penalize	CLUST	61.7±6.9	10.1±6.8	10.1±6.8
DSD	Penalize, ReqObl	CLUST	62.2±9.3	11.7±8.0	11.7±8.0
Decomp		SIMPLE/CLUST	64.5±3.6	4.5±2.0	4.5±2.0
Baseline	expected×ACT-PAT	SIMPLE	65.3±3.8	9.7±6.8	9.7±6.8
WFD		CLUST	66.0±3.1	13.4±8.6	13.4±8.6
WFD		SIMPLE	67.8±1.1	12.7±3.3	12.6±3.3

Table 2.4: Evaluation of direct frame suggestion methods.

2.8 Empirical Evaluation of Direct Methods

Table 2.4 summarizes the results of the various methods in terms of expected saving (ES), frame precision (Prec) and frame recall (Rec), averaged over individual verb lemmas. The ± bounds represent standard deviations based on four iterations of a 10-fold evaluation.

The methods were evaluated on VALEVAL verbs and framesets from VALLEX 1.0. In every fold we pick one tenth of verb lemmas as the test verbs. The remaining 9/10s of verbs and their VALEVAL occurrences are available to the methods for training. Every method has to produce a frameset for every test verb based on unlabelled occurrences in the VALEVAL corpus.

The column "Fit Frame Count" specifies whether the method had access to the expected (correct) number of frames and how did it use it (SIMPLE or CLUST). Our "Baseline" method is to suggest a frame with two obligatory slots: *ACT.obl.nom PAT.obl.acc*. The baseline method varies in the number of times we repeat this frame in the suggested frameset, e.g. 2× indicates that every verb receives the frame twice while avg× uses the training verbs to find out the average number of frames per verb.

We observe that baseline methods generally perform better than our frame-suggestion techniques both in case when the methods do not access the expected number of frames as well as when they do. It is only WFD (CLUST and SIMPLE) that insignificantly outperforms the baseline.

An inspection of detailed logs revealed that the methods differ in reasons of failure. Both WFD and DSD tend to suggest too many different frames (which is confirmed by a relatively higher recall). The reason for this overgeneration lies simply in abundance of training frames leading to a big variety in frames suggested. By fitting the output frame count to the expected number of frames, we significantly raise the ES. The very extreme improvement can be seen for WFD, jumping from the worst rank (ES 21.4%) to the best one (ES 67.8%).

For DSD, we evaluated two minor modifications of the method. First, as we see, penalizing superfluous slots helps to find more relevant training observations (compare Penalize vs. noPenalize). Second, we consider only such training observations where all obligatory slots are most likely realised on the surface (ReqObl). The set of training observations thus better represents the possible frames and DSD gains a small improvement in ES. Alternatively, we could group training verb occurrences by semantic class and use only a restricted set of most typical instance of a frame from each group, partially approaching the method described in Section 2.9 below.

Decomp on the other hand fails because it produces too few (and too short) frames. Only very few frame components such as *ACT.obl.nom* or *PAT.obl.acc* are proposed. For other frame components, the learners have seen too many negative training examples (instances of other frames without that particular component) so they tend to undergenerate.

In conclusion, the key aspect of frame suggestion as evaluated by ES, is to guess correctly the number of frames. Beyond that, more complicated methods as Decomp or DSD do not bring any improvement. A more promising approach is to carefully filter training examples and to add additional features to the relatively straightforward method of WFD. We further discuss the problems of frame extraction methods in Section 2.10 below.

2.9 PatternSearch: Guessing Verb Semantic Class

As seen in Section 2.8, direct methods of frame suggestion averaged over all verbs do not bring much improvement over the baseline. In this section, we tackle frame suggestion indirectly, via the semantic class of a verb (sense). In this preliminary experiment published in Benešová and Bojar (2006), we focus on one class, namely the verbs of communication (see Section 2.9.2 below).

As noted by Véronis (2003), syntax provides extremely powerful tool for sense discrimination and likewise, verbs with a similar sense tend to have similar frames (Levin, 1993). With these observations in mind, we formulate

the syntactic pattern typical for verbs expressing communication and search a given corpus $\mathbb{C}$ for verbs appearing in the pattern (thus the name PatternSearch). If a substantial portion of the verb's occurrences matches the pattern, we assume the verb belongs to the communication class. As such, the VALLEX entry of the verb should include at least one frame conveying the communication meaning.

In the following we provide details on semantic verb classes as available in VALLEX (Section 2.9.1) and verbs expressing communication in particular. In Section 2.9.3, we evaluate automatic identification of verbs belonging to this semantic class. Finally Section 2.9.5 utilizes class identification to prescribe valency frames to unseen verbs.

2.9.1 Verb Classes in VALLEX

Verb classes were introduced to VALLEX primarily to improve data consistence because observing whole groups of semantically similar verbs together simplifies data checking.

Classification of verbs into semantic classes is a topical issue in linguistic research (see e.g. Levin's verb classes Levin (1993), PropBank Palmer *et al.* (2005) , LCS Jackendoff (1990); Dorr and Mari (1996), FrameNet Baker *et al.* (1998)). Verb classes as defined in VALLEX 1.0 and 1.5, though influenced by the various streams of research, are built independently and using a custom classification, mainly due to differences in the theoretical background and in the methods of description. VALLEX classes are built thoroughly in a bottom-up approach: frame entries already listed in VALLEX are assigned to a common class mostly on the basis of syntactic criteria: the number of complements (actants and free modifications), their type (mainly obligatory or optional), functors and their morphemic realizations. It should be noted that verb classes and their descriptions in VALLEX 1.5 are still tentative and the classification is not based on a defined ontology but it is to a certain extent intuitive.

VALLEX 1.5 defines about 20 verb classes (communication, mental action, perception, psych verbs, exchange, change, phase verbs, phase of action, modal verbs, motion, transport, location, expansion, combining, social interaction, providing, appoint verb, contact, emission, extent) that contain on average 6.1 distinct frame types (disregarding morphemic realizations and complement types).

2.9.2 Verbs of Communication

The communication class is specified as the set of verbs that render the situation when "a speaker conveys information to a recipient". For the sake of simplicity, we use the term **verbs of communication** to refer to verbs with at least one sense (frame) belonging to the communication class.

Besides the slots ACT for the "speaker" and ADDR for the "recipient", verbs of communication are characterized by the entity "information" that is usually expressed as a dependent clause introduced by a subordinating conjunction or as a nominal structure.

There are some other classes (mental action, perception and psych verbs) that also include the "information" element in the frame but they usually do not require any slot for a "recipient". However, in a small number of cases when the addressee which represents the "recipient" does not appear explicitly in the valency frame of a verb of communication (e.g. *speak* or *declare*), this distinctive criterion fails.

Verbs of communication can be further divided into subclasses according to the semantic character of "information" as follows: simple information (verbs of announcement: *říci (say), informovat (inform)*, etc.), questions (interrogative verbs: *ptát se (ask)*, etc.) and commands, bans, warnings, permissions and suggestions (imperative verbs: *poručit (order), zakázat (prohibit)*, etc.). The dependent clause after verbs of announcement is primarily introduced by the subordinating conjunction *že (that)*, interrogative by *zda (whether)* or *jestli (if)* and imperative verbs by *aby (in order to)* or *ať (let)*.

2.9.3 Automatic Identification of Verbs of Communication

In the present section, we investigate how much the information about the valency frame combined with the information about morphemic realizations of valency complements can contribute to an automatic recognition of verbs of communication.

The experiment is primarily based on the idea that verbs of communication can be detected by the presence of a dependent clause representing the "information" and an addressee representing the "recipient".

This idea can be formalized as a set of queries to search the corpus for occurrences of verbs accompanied by: (1) a noun in one of the following cases: genitive, dative and accusative (to approximate the ADDR slot) and (2) a dependent clause introduced by one of the set of characteristic subordinating conjunctions (*že*, *aby*, *ať*, *zda* or *jestli*) (to approximate the slot of "information").

We disregard the freedom of Czech word order which, roughly speak-

ing, allows for any permutation of a verb and its complements. In reality, the distribution of the various reorderings is again Zipfian with the most typical ordering (verb+N234+subord) being the most frequent. In a sense, we approximate the sum of occurrences in all possible reorderings with the first, maximal, element only. On the other hand we allow some intervening adjuncts between the noun and the subordinating clause.

We use the Manatee corpus manager (Rychlý and Smrž, 2004) to perform the searches in Czech National Corpus.

2.9.4 Evaluation against VALLEX and FrameNet

We sort all verbs by the descending number of occurrences of the tested pattern. This gives us a ranking of verbs according to their "communicative character", typical verbs of communication such as *říci* (say) appear on top. Given a threshold[10], one can estimate the class identification quality in terms of a confusion matrix: verbs above the threshold that actually belong to the class of verbs of communication (according to a golden standard) constitute **true positives** (TP), verbs below the threshold and not in the communication class constitute **true negatives** (TN), etc.

A well-established technique of the so-called ROC curves allows to compare the quality of rankings for all possible thresholds at once. We plot the **true positive rate** ($TPR = TP/P$ where P is the total number of verbs of communication) against the **true negative rate** ($TNR = TN/N$, N stands for the number of verbs with no sense of communication) for all thresholds.

We evaluate the quality of class identification against golden standards from two sources. First, we consider all verbs with at least one frame in the communication class from VALLEX 1.0 and 1.5 and second, we use all possible word-to-word translations of English verbs listed in FrameNet 1.2[11] Communication frame and all inherited and used frames (For an explanation, see Fillmore *et al.* (2001); Fillmore (2002); the English-to-Czech translations were obtained automatically using available on-line dictionaries). As the universum (i.e. $P + N$), we use all verbs defined in the respective version of VALLEX and all verbs defined in VALLEX 1.5 for the FrameNet-based evaluation.

Figure 2.5 displays the TPR/TNR curve for verbs suggested by the pattern V+N234+subord. The left chart compares the performance against various golden standards, the right chart gives a closer detail on the contribution of different subordinating conjunctions.

[10] See Kilgarriff (2005) for a justification of this simple thresholding technique as opposed to more elaborated methods of statistical significance testing.

[11] `http://framenet.icsi.berkeley.edu/`

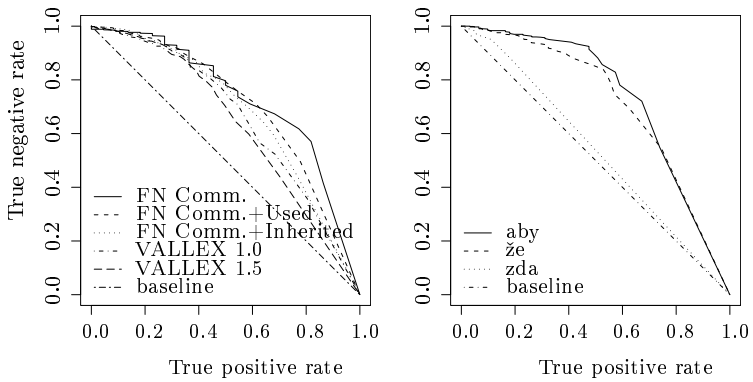


Figure 2.5: Verbs of communication as suggested by the pattern V+N234+subord, evaluated against VALLEX and FrameNet (left) and evaluated against VALLEX 1.0 for the three main contributing subordinating conjunctions (*aby*, *že*, *zda*) independently (right).

The closer the curve lies to the upper right corner, the better the performance is compared to the golden standard. With an appropriate threshold, about 40% to 50% of verbs of communication are identified correctly while 20% of non-communication verbs are falsely marked, too. We get about the same performance level for both VALLEX and FrameNet-based evaluation. This confirms that our method is not too tightly tailored to the classification introduced in VALLEX.

The right chart in Figure 2.5 demonstrates that the contribution of different subordinating conjunctions is highly varied. While *aby* and *že* contribute significantly to the required specification, the verbs suggested by the pattern with *zda* are just above the baseline. (The conjunctions *ať* and *jestli* had too few occurrences in the pattern.)

Weak Points of Patterns

On the one hand, our queries are not able to find all verbs of communication for the following reasons: (1) We search only for cases where the "information" element is expressed as a subordinate clause. While nominal structures can be used here, too, allowing them in the queries would cause confusion with verbs of exchange (e.g. *give* or *take*). (2) Verb occurrences with some of the core frame elements not expressed on the surface are not identified by the queries.

On the other hand, the fact that conjunctions *aby* and *že* are homonymous lowers the precision of the queries and introduces false positives. We

Suggested frames	*ES* [%]
Specific frame for verbs of communication, default for others	38.00 ± 0.19
Baseline 1: ACT(1)	26.69 ± 0.14
Baseline 2: ACT(1) PAT(4)	37.55 ± 0.18
Baseline 3: ACT(1) PAT(4) ADDR(3,4)	35.70 ± 0.17
Baseline 4: Two identical frames: ACT(1) PAT(4)	39.11 ± 0.12

Table 2.5: Expected saving when suggesting frame entries automatically.

tried to eliminate some of incorrectly chosen verbs by a refinement of the queries. (For instance, we omitted certain combinations of demonstratives plus conjunctions: *tak*, *aby* (*so that*), *tak*, *že* (*so that*), etc.) A further problem is caused by cases when the identified dependent clause is not a member of the valency frame of the given verb but depends on the preceding noun. PatternSearch does not make use of the syntactic analysis of the sentence and thus cannot reject such examples.

2.9.5 Application to Frame Suggestion

The method of searching corpus for typical patterns described in the previous section can contribute to frame extraction task in the following manner: for all verbs occurring frequently enough in the typical pattern, we propose the most typical "communication frame" consisting of ACT, ADDR and PAT (all obligatory). For each verb independently, we assign only conjunctions discovered by the queries to the PAT. Every verb of communication can have some additional senses not noticed by our method but at least the communication frame should be suggested correctly.

Table 2.5 displays the *ES* (expected saving, Section 2.5.1) as reported in Benešová and Bojar (2006) of four various baselines and the result obtained by our method. When we assume that every verb has a single entry and this entry consists of a single frame with the ACT slot only, *ES* estimates that about 27% of editing operations was saved. Suggesting an ACT and a PAT helps even better (Baseline 2, 38%), but suggesting a third obligatory slot for an addressee (realized either as a dative (3) or an accusative (4)) is already harmful, because not all the verb entries require an ADDR.

We can slightly improve over Baseline 2 if we first identify verbs of communication automatically and assign ACT PAT ADDR with appropriate subordinating conjunctions to them, leaving other verbs with ACT PAT only. This confirms our assumption that verbs of communication have a typical three-slot frame and also that our method managed to identify some of the verbs correctly.

Our *ES* scores are relatively low in general and Baseline 4 suggests a reason for that: most verbs listed in VALLEX have several senses and thus several frames. In this experiment, we focus on the communication frame only, so it still remains quite expensive (in terms of *ES*) to add all other frames. In Baseline 4, we suggest a single verb entry with two core frames (ACT PAT) and this gives us a greater saving because most verbs indeed ask for more frames.

2.10 Discussion

All our direct methods (WFD, DSD and Decomp) perform relatively poorly compared to the baselines. It is only the very specific experiment with verbs of communication (Section 2.9) that provides somewhat promising results.

Before suggesting general conclusions, let us briefly mention similar projects. Of the many lexicographic enterprises we name just a few that closely relate to our observations.

2.10.1 Related Research

Rosen *et al.* (1992) describe formal representation of valency frames for the machine translation system MATRACE (Hajič *et al.*, 1992) and design a procedure to convert subcategorization frames from Oxford Advanced Learners' Dictionary (Hornby, 1974).

Skoumalová (2001) implements rules to convert surface frames collected from a compilation of manual dictionaries (BRIEF, (Pala and Ševeček, 1997)) to tectogrammatical valency frames, including explicit encoding of allowed passivization alternations. The resulting lexicon is utilized in a toy LFG grammar.

Bond and Fujita (2003) describe a successful semi-automatic method for extending a Japanese valency dictionary by copying frames from translation equivalents: a verb not covered in the target valency dictionary is translated (using a simple translation dictionary) to English and back to arrive at a known verb. Frames of the known verb are copied to the newly added verb, subject to various forms of manual filtering. The experiment confirms that verb valency is strongly related to verb meaning (and exploits the fact that translation preserves meaning). A surprising observation is that manual checking whether the new frame belongs to a verb performed either by untrained annotators validating correctness of a paraphrase or by trained lexicographers validating the frame assignment as such is equally time-consuming. In practice, Bond and Fujita (2003) suggest to prefer the lexicographers because the whole entry is checked and also because untrained

annotators often judge the grammaticality of the paraphrase unreliably. An automatic learner (C5.0, Quinlan (2002)) failed to improve over the baseline and Bond and Fujita (2003) thus mention that frame entry construction inevitably requires manual effort.

Kipper-Schuler (2005) follows up on experiments by Kingsbury (2004) to automatically cluster verbs appearing in Penn Treebank for the purpose of VerbNet extension. A manual evaluation of the clusters revealed that only about 5% of verbs were assigned to a reasonably accurate cluster and could have been added to the VerbNet. Reasons for the little precision include (1) highly skewed domain of the Penn Treebank (mostly financial texts), (2) lack of syntactic context in the sentences that would enable to disambiguate between verb usages and finally (3) no semantic classification of verbs' arguments. Apart from the domain dependence, the same problems apply to our automatic extraction of VALLEX frames. A more fruitful approach was to exploit clustering of verbs already present in WordNet from where 36–40% of suggested verbs could have been used.

Dorr and Jones (1996) successfully use WordNet and syntactic descriptions of verbs in LDOCE (Procter, 1978) to semantically classify verbs not covered in Levin's verb classes (Levin, 1993): for each new verb, synonyms are found in WordNet. All Levin classes the synonyms belong to are considered as candidate classes, but only the single class is chosen that best matches the syntactic description of the verb in LDOCE. The procedure can also hypothesize a new class in case none of the verb's synonyms is covered in Levin's classification or the syntactic descriptions of the class and the verb differ too much. Manual evaluation on a small sample suggested 82% accuracy: the class chosen was one of plausible classes for the verb in 82% of verbs. The syntactic descriptions from LDOCE serve as a filter to restrict the set of classes suggested by the synonyms. We believe that corpus evidence could be used as an alternative filtering technique if LDOCE syntactic description were not available. The key component though remains WordNet as the source of synonyms.

Schulte im Walde (2003) carries out extensive research on automatic clustering of German verbs into semantic classes based on syntactic criteria and also selectional restrictions. After fine-tuning the set of features she is able to automatically derive semantic clustering of verbs that ignores sense ambiguity of verbs (hard clustering method, each verb is assumed to belong to one class only). However, her classes are described by a set of frames, so one could use this method to assign sets of frames to verbs. The main difference between her and our goal is thus the surface vs. deep syntactic layer of representation.

2.10.2 Lack of Semantic Information

The failure of our direct methods suggests that purely surface syntactic observations are not sufficient to derive deep syntactic (or semantic) generalizations.

Successful projects mentioned above always include some ready-made component capable of semantic generalization employed either for the verb itself or for the modifiers. For instance, synonyms of the verb from WordNet or synonyms derived via translation to another language are used as source verbs to copy the syntactic information from.

Though not clearly confirmed by Schulte im Walde (2003), selectional restrictions on verb modifiers are a significant predictor of verb sense distinctions. We thus believe that both further refinement of VALLEX verb classes as well as the addition of selectional restrictions could improve the accuracy of our application.

2.10.3 Deletability of Modifiers

One of the main problems of our direct methods is that they do not explicitly handle "deleted" modifiers, i.e. frame slots that are not realized on the surface. It is only the method PatternSearch that inherently solves the problem by ignoring all occurrences of the verb in question where some of the modifiers required by the pattern are missing, though lowering the recall of the method.

An approach similar to Sarkar and Zeman (2000) where frame subsets are considered or the hierarchical browsing of verb occurrences suggested by Bojar (2003) would have to be incorporated into the methods.

2.10.4 Need to Fine-Tune Features and Training Data

The features we use in our direct methods WFD and Decomp are rather straightforward observations from the close (syntactic) neighbourhood of the verb. We also train our models on all available instances of all training verbs.

Possibly, the noise in the training data could be reduced to a great extent by carefully restricting the set of training verbs to a few representatives (e.g. one frame per semantic class or a limited number of centroids selected automatically from all known frames). We could also use some selection of training sentences, such as the promising method of selecting syntactically simple sentences as implemented in Bojar (2003) but aiming at sentences with most modifiers realized on the surface.

Similarly, it is well known that feature selection is vital for performance of classification methods. In out preliminary experiments with WFD features,

every feature type contributed to the performance and we could not restrict the set of features in any way without a loss. This suggests that additional features (or feature combinations) are still to be sought for.

2.10.5 Lack of Manual Intervention

One of the reasons of the failure of our direct methods is undoubtedly the the aim at an end-to-end automatic approach.

Our PatternSearch experiment as well as related approaches include a manual filtering step of either the suggestions the system has made or the patterns the system searches for.

We envisage a lexicographers' tool that automatically "summarizes" corpus evidence to clusters based on e.g. DSD or the surface-syntactic features used in WFD. The lexicographer would then mark occurrences not fitting well to the suggested cluster, thus creating some WFD-annotated training data for the verb. In the next iteration, the system would try to follow the suggested classification and summarize further corpus data, possibly employing some semi-supervised clustering techniques (Basu, 2005). A similar approach, though limited to independent pairs of verb and one of its modifiers and without the proposed annotation loop, is successfully employed in Word Sketches (Rychlý and Smrž, 2004).

2.11 Conclusion and Further Research

Chapter 2 was devoted to methods of automatic extraction of valency frames based on corpus evidence. We motivated the creation of valency dictionaries by expected contribution to various NLP applications. Then we reviewed basic formal aspects of valency frames in FGD and simplified the definition for our purpose.

A novel metric (ES) was proposed to evaluate directly how much of a lexicographer's work is saved using a method of automatic suggestion of verb frames. We proposed three rather direct methods of frame suggestion (WFD, DSD and Decomp) and one indirect method that exploits semantic classification of the verbs (PatternSearch, Section 2.9).

We have to conclude that the task of automatic creation of lexicon entries is a very complex process. None of our direct methods was able to significantly improve over the baseline. As confirmed by related research for other languages, manual intervention in the process seems inevitable.

More or less successful methods such as (Bond and Fujita, 2003) or our PatternSearch exploit the fact that verbs with a similar meaning have similar valency frames. In general, an acceptable performance of the methods of

extraction is achieved only in setups aimed at high precision (and thus low recall) that heavily filter available data but this may negatively affect the utility of the lexicons in applications (Zhang *et al.*, 2007).

Ideally, the lexicons we have just described would improve NLP applications, e.g. the quality of machine translation (MT). To achieve this, the methods would have to be extended to acquire bilingual valency dictionaries. As other research suggests (Ikehara *et al.*, 1991; Boguslavsky *et al.*, 2004; Fujita and Bond, 2004; Liu *et al.*, 2005), such dictionaries might indeed help, though we are not aware of any conclusive improvement over the state-of-the-art translation quality, see Section 5.1.3. For Czech-English pair, we carried out some preliminary experiments with extracting parallel verb frames (Bojar and Hajič, 2005).

In the following, we do not take any side steps and move towards the goal of machine translation, describing a syntax-based (Chapter 3) and a phrase-based (Chapter 4) MT system. Later, we will come back to a more general discussion on the utility of lexicons in NLP applications in Chapter 5.

Chapter 3

Machine Translation via Deep Syntax

In the previous chapter we studied methods of automated lexical acquisition. Resulting syntactic lexicons can serve as a resource for various NLP applications. In order to better empirically understand the applicability of lexicons, we now focus on a single practical task, namely machine translation (MT). After a brief review of approaches to MT (Section 3.1), we describe a syntax-based MT system. In theory, this is the approach where deep syntactic lexicons could be later used.

3.1 The Challenge of Machine Translation

Machine translation (MT) is an intriguing task. Researchers have hoped in automated text translation since the era of John von Neumann and Alan Turing (see Hutchins (2005) or the IBM press release in 1954[1]), and the field has seen both spectacular failures[2] as well as surge of activity and success. For a review including a summary of issues that an MT system has to overcome see e.g. Dorr *et al.* (1998).

While fully automatic high-quality MT is still far beyond our reach, restricted settings often allowed to create highly successful applications such as computer tools aiding human translation (e.g. translation memories, see Lagoudaki (2006)), closed-domain fully automatic systems (Chevalier *et al.*, 1978), or tentative machine translation to enable at least a partial access to information in a foreign text (e.g. web services Babelfish[3] or Google Translation[4]).

[1] `http://www-03.ibm.com/ibm/history/exhibits/701/701_translator.html`

[2] Failure to meet expectations causing a decline in funding for a decade (ALPAC, 1966; Hutchins, 2003) or failure to produce any working system in the EUROTRA project (Oakley, 1995; Hutchins, 1996). Note however, that there are quite conflicting objectives in MT research and even a failing project can bring a very significant progress in theoretical understanding or language modelling, see Rosen (1996) for a discussion.

[3] `http://babelfish.altavista.com/`

[4] `http://translate.google.com/`

In essence, the task of MT is to efficiently store and correctly reuse pieces of texts previously translated by humans to translate sentences never seen so far.[5] Some methods follow the line very tightly, not being able to produce any word or expression not seen in some training text, while some methods (most notably all rule-based or dictionary-based ones) operate with a very distilled representation of words and their translations. In the latter setup, training texts as well as a broad world knowledge were processed by human experts, so there is no well defined set of training data and no direct link between the data and the system. Further serious empirical questions arise as we start to investigate what the best "piece" of a sentence to reuse might be, as discussed below.

3.1.1 Approaches to Machine Translation

One of the key distinctions between various MT systems is the level of linguistic analysis employed in the system, see the MT triangle by Vauquois (1975) in Figure 3.1. Roughly speaking, an MT system is "direct" or "shallow" if it operates directly with words in source and target languages and it is "deep" if is uses some formal representation (partially) describing the meaning of the sentence. We examine both of the approaches further below.

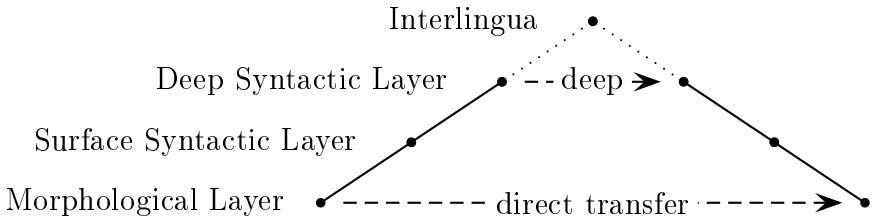


Figure 3.1: Vauquois' triangle of approaches to machine translation.

Another distinction is made between "rule-based" and "statistical" (or "stochastic" or "data-driven") systems. In rule-based systems, all the implementation work is done by human experts, in statistical systems, humans design a probabilistic model describing the process of translation and use large amounts of data to train the model.

To an extent, we do not consider the difference between "rule-based" and "statistical" approaches being too big. In both cases, there has to be someone

[5]Human translators proceed well beyond this boundary, trying to understand the described situation based on other information sources and e.g. to enrich the translation with all explanation necessary for the reader.

who does some data abstraction at some point. In hand-crafted rule-based systems, the abstraction happens as human translators learn the two languages and formally describe the rules of translation. In data-driven systems, the abstraction according to the specification of the model happens either at a pre-processing phase (collecting statistics) or on the fly when searching for sentences similar to the one that is to be translated (example-based methods). Moreover, many rule-based systems rely on large linguistic resources such as translation dictionaries anyway and in such cases, automated creation of such resources is highly desirable (see Chapter 2).

Direct (Shallow) MT

Introduced by King (1956) and applied by Brown *et al.* (1988), shallow MT systems treat words in a input sentence as more or less atomic units and attempt a direct conversion of the input sequence of atomic units into the output sequence of atomic units.

For instance, the Czech sentence *Dobré ráno* can be translated to English *Good morning* using a simple word-to-word translation dictionary. The linguistic inadequacy of the direct approach becomes apparent if we consider a similar sentence *Dobrý večer (Good evening)*. A completely uninformed system wastefully needs two new entries to the dictionary (*Dobrý* for *Good* and *večer* for *evening*) because it has no idea that both *Dobré* and *Dobrý* are just two morphological variants of the same word. In order to reverse the translation direction, some additional information has to be provided to make the system correctly choose between *Dobrý* and *Dobré* for *Good*.

In short, direct approaches start with little or no linguistic theory and introduce further extensions to the process of translation only when necessary. As we will see in Chapter 4, such systems can still deliver surprisingly good results, and more so once some (limited) linguistic knowledge is implemented into the design of the system.

Deep Syntactic MT

First machine translation systems as well as prevailing commercial MT systems to date (e.g. SYSTRAN) incorporate principles from various linguistic theories from the very beginning.

For an input sentence represented as a string of words, some symbolic representation is constructed, possibly in several steps. This symbolic representation, with the exception of a hypothetical Interlingua, remains language dependent, so a transfer step is necessary to adapt the structure to the target language. The translation is concluded by generating target-language string

of words from the corresponding symbolic representation.

In the following, we focus on one particular instance of this symbolic representation, namely the framework of FGD (see Section 2.2). We experiment primarily English-to-Czech translation via the t-layer (deep) and compare it to transfer at the a-layer (surface syntax). Previous research within the same framework but limited to rather surface syntax includes the system APAČ (Kirschner and Rosen, 1989).

Other examples of a deep syntactic representation, in essence very similar to FGD, include Mel'čuk (1988), Microsoft logical form (Richardson *et al.*, 2001) or the ideas spread across the projects PropBank (Kingsbury and Palmer, 2002), NomBank (Meyers *et al.*, 2004) and Penn Discourse Treebank (Miltsakaki *et al.*, 2004). MT systems are also being implemented in less dependency-oriented formalisms such as the DELPH-IN initiative (Bond *et al.*, 2005) for HPSG (Pollard and Sag, 1994). See e.g. Oepen *et al.* (2007) and the cited papers for a recent overview of the LOGON project that combines various formalisms of deep syntactic representation.

3.1.2 Advantages of Deep Syntactic Transfer

The rationale to introduce additional layers of formal language description such as the tectogrammatical (t-) layer in FGD is to bring the source and target languages closer to each other. If the layers are designed appropriately, the transfer step will be easier to implement because (among others):

- t-structures of various languages exhibit less divergences, fewer structural changes will be needed in the transfer step.
- t-nodes correspond to auto-semantic words only, all auxiliary words are identified in the source language and generated in the target language using language-dependent grammatical rules between t- and a- layers.
- t-nodes contain word lemmas, the whole morphological complexity of either of the languages is handled between m- and a- layers.
- the t-layer abstracts away word-order issues. The order of nodes in a t-tree is meant to represent information structure of the sentence (topic-focus articulation). Language-specific means of expressing this information on the surface are again handled between t- and a- layers.

Overall, the design of the t-layer aims at reducing data sparseness so less parallel training data should be sufficient to achieve same coverage.

Moreover, the full definition of the t-layer includes explicit annotation of phenomena like co-reference to resolve difficult but inevitable issues of

e.g. pronoun gender selection. As tools for automatic tectogrammatical annotation improve, fine nuances could be tackled.

3.1.3 Motivation for English→Czech

This thesis focuses on translation from English to Czech. Apart from personal reasons, our choice has two advantages: both languages are well studied and there are available language data for both of the languages.

Table 3.1 summarizes some of the well known properties of Czech language[6]. Czech is an inflective language with rich morphology and relatively free word order. However, there are important word order phenomena restricting the freedom. One of the most prominent examples are clitics, i.e. pronouns and particles that occupy a very specific position within the whole clause. The position of clitics is rather rigid and global within the sentence. Examples of locally rigid structure include (non-recursive) prepositional phrases or coordination. Other elements, such as the predicate, subject, objects or other modifiers of the verb may be nearly arbitrarily permuted. Such permutations correspond to the topic-focus articulation of the sentence. Formally, the topic-focus articulation is expressed as the order of nodes at the t-layer.

Moreover, like other languages with relatively free word order, Czech allows non-projective constructions (crossing dependencies). Only about 2% of edges in PDT are non-projective, but this is enough to make nearly a quarter (23.3%) of all the sentences non-projective. While in theory there is no upper bound on the number of gaps (Holan *et al.*, 2000; Kuhlmann and Möhl, 2007) in a Czech sentence (see Figure 3.2), Debusmann and Kuhlmann (2007) observe that 99% of sentences in PDT contain no more than one gap and are well-nested, which makes them parsable by Tree-Adjoining Grammars (TAG, Joshi *et al.* (1975), see also the review by Joshi *et al.* (1990)). Note that other types of texts may exhibit more complex sentence structure.

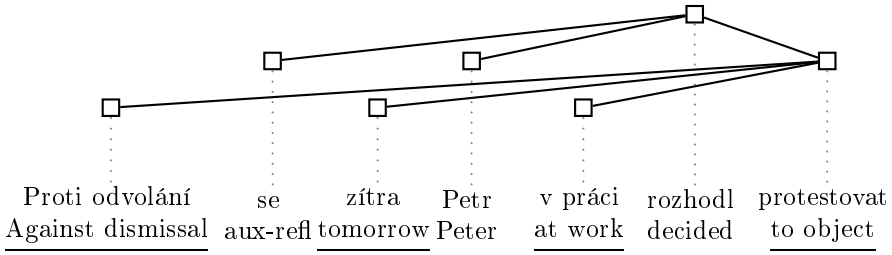
3.1.4 Brief Summary of Czech-English Data and Tools

Table 3.2 summarizes available Czech monolingual and Czech-English parallel corpora, including the available annotation. We use the tools listed in Table 3.3 to automatically add any further layers of annotation and to generate plaintext from the deep representation.

[6]Data by Nivre *et al.* (2007), Zeman (http://ufal.mff.cuni.cz/~zeman/projekty/neproj), Holan (2003), and Bojar (2003). Consult Kruijff (2003) for empirical measurements of word order freeness.

	Czech	English
Morphology	rich	limited
	$\geq$ 4,000 tags	50 used
	$\geq$ 1,400 actually seen	
Word order	free with rigid	rigid
	global phenomena	
Known dependency parsing results		
Labelled edge accuracy	80.19%	89.61%
Unlabelled edge accuracy	86.28%	90.63%

Table 3.1: Properties of Czech compared to English.



Peter decided to object against the dismissal at work tomorrow.

The construction, taken from Holan *et al.* (2000), is based on two verbs and intermixed modifiers where the dependency relations are disambiguated on the basis of syntactic criteria (e.g. obligatory reflexive particle *se* or subcategorization for a particular preposition or case) and semantic criteria (e.g. verb in past tense cannot accept time modifier referring to future): The non-projective dependencies are *se* and *Peter* depending on the main verb *decided* but appearing within the span of dependents of *to object*: *against dismissal*, *tomorrow*, *at work*. With the main verb itself, there are 3 gaps within the yield of *to object*.

Figure 3.2: Number of gaps in a Czech sentence is not bounded in theory.

Monolingual Corpora

Name and version Annotation	Sents.	Tokens
Czech National Corpus (e.g. SYN2000d) automatic m-layer, (Kocek *et al.*, 2000)	6.8M	114M
PDT 2.0 manual t-layer/manual m-layer, (Hajič, 2004a)	50k/115k	0.8M/2.0M

Parallel Czech-English Corpora

Name and version Annotation	Czech/English Sents.	 Tokens
PCEDT 1.0 (Čmejrek *et al.*, 2004) Czech/English automatic m-, a- and t-layer	22k/49k	0.5M/1.2M
CzEng 0.7 (Bojar *et al.*, 2008) automatic sentence alignment, tokenized	1.4M/1.4M	21M/23M

Table 3.2: Available Czech monolingual and Czech-English parallel corpora.

A new version of Prague Czech-English Dependency Treebank (PCEDT 2.0) is currently under development. PCEDT 2.0 will not only be about twice the size of PCEDT 1.0, but more importantly the annotation at both Czech and English t-layers will be manual. This will allow to collect reliable estimates of structural divergence at the t-layer and train deep-syntactic transfer models on highly accurate data.

3.2 Synchronous Tree Substitution Grammar

Synchronous Tree Substitution Grammars (STSG) were introduced by Hajič *et al.* (2002) and formalized by Eisner (2003) and Čmejrek (2006). They capture the basic assumption of syntax-based MT that a valid translation of an input sentence can be obtained by local structural changes of the input syntactic tree (and translation of node labels) while there exists a derivation process common to both of the languages. Some training sentences may violate this assumption because human translators do not always produce literal translations but we are free to ignore such sentences in the training.

As illustrated in Figure 3.3, STSG describe the tree transformation process using the basic unit of a **treelet pair** and the basic operation of **tree substitution**. Both source and target trees are decomposed into treelets that fit together. Each treelet can be considered as representing the minimum

Step	Tool Used
English morphological analysis (text→m)	Minnen *et al.* (2001)
English tagging (text→m)	Ratnaparkhi (1996) or Brants (2000)
English constituency parsing (m→phrase structure)	Collins (1996)
English dependencies (phrase structure→a)	hand-written rules
English tectogrammatical parsing (a→t)	rules similar to Čmejrek *et al.* (2003)
Czech morphological analysis (text→m)	Hajič (2004b)
Czech dependency parsing (m→a)	McDonald *et al.* (2005)
Czech tectogrammatical parsing (a→t)	Klimeš (2006) or Žabokrtský (2008a)
Czech tectogrammatical generation (t→text)	Ptáček and Žabokrtský (2006)

Table 3.3: Tools used for the preparation of training data and in the end-to-end evaluation.

translation unit. A treelet pair such as depicted in Figure 3.4 represents the structural and lexical changes necessary to transfer local context of a source tree into a target tree.

Each node in a treelet is either **internal** (• , constitutes treelet internal structure and carries a lexical item) or **frontier** (⌒, represents an open slot for attaching another treelet). Frontier nodes are labelled with **state labels** (such as "_Sb" or "_NP"), as is the root of each treelet. A treelet can be attached at a frontier node only if its root state matches the state of the frontier.

A **treelet pair** describes also the **mapping** of the frontier nodes. A pair of treelets is always attached synchronously at a pair of matching frontier nodes.

Depending on our needs, we can encode ordering of nodes as part of each treelet. If only local ordering is used (i.e. we record the position of a parent node among its sons), the output tree will be always projective. If we record global ordering of all nodes in a treelet, the final output tree may contain non-projectivities introduced by non-projective treelets (the attaching operation itself is assumed to be projective).

STSG is generic enough to be employed at or across various layers of annotation (e.g. an English t-tree to a Czech t-tree or an English a-tree to a Czech a-tree). Our primary goal is to transfer at the tectogrammatical layer. Other applications of STSG include e.g. text summarization (Cohn and Lapata, 2007).

STSG can be also seen as a simplification of the (Synchronous) Tree-Adjoining Grammars (TAG, Joshi *et al.* (1975)). In addition to the tree-substitution operation, TAG allows to "adjoin" a tree at an internal node as illustrated in Figure 3.5.

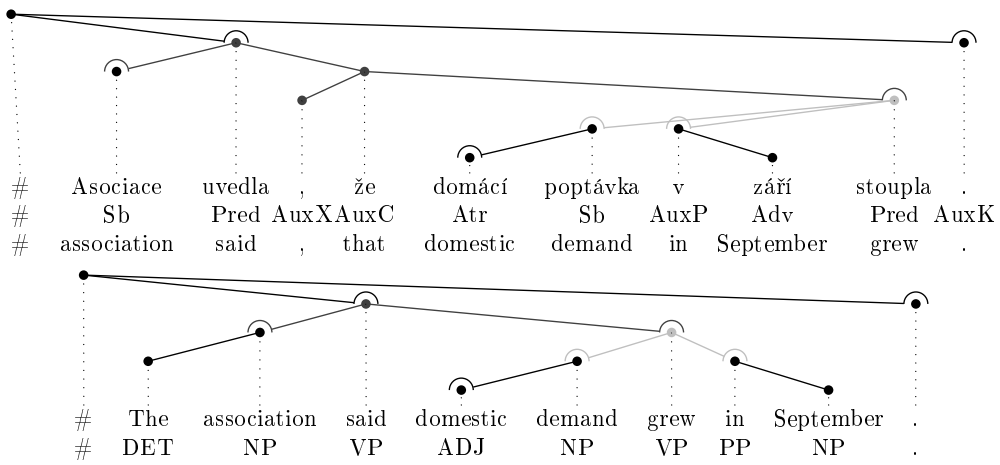


Figure 3.3: A sample pair of analytical trees synchronously decomposed into treelets. For explanation of the graphical symbols used see the text, linguistic annotation is provided for illustration purposes only.

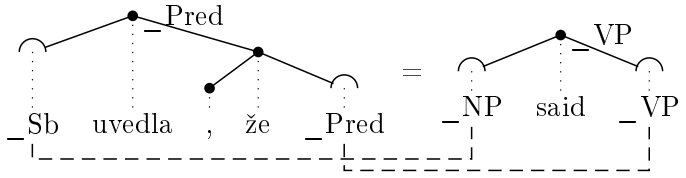


Figure 3.4: A sample analytical treelet pair.

3.3 STSG Formally

We now formally describe the core elements in STSG as motivated above to make the thesis self-contained and also because we slightly differ from the definition e.g. by Čmejrek (2006), see below.

Given a set of states Q and a set of word labels L, we define:

A **treelet** t is a tuple (V, V^i, E, q, l, s) where:

- V is a set of **nodes**,
- $V^i \subseteq V$ is a nonempty set of **internal nodes**. The complement $V^f = V \setminus V^i$ is called the set of **frontier nodes**,
- $E \subseteq V^i \times V$ is a set of directed **edges** starting from internal nodes only and forming a directed acyclic graph,
- $q \in Q$ is the **root state**,

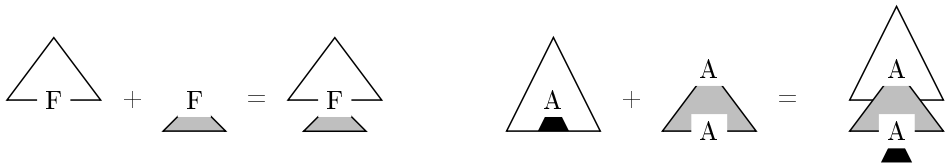


Figure 3.5: Tree substitution at a frontier node F and tree adjunction at an internal node A.

- $l : V^i \rightarrow L$ is a function assigning labels to internal nodes,
- $s : V^f \rightarrow Q$ is a function assigning states to frontier nodes.
- Optionally, some additional structure can keep track of local or global ordering of nodes.

For convenience, we will use the shorthand $t.q$ for the root state, $t.s$ for the frontier state function, and other shortcuts for all other properties of t using the same analogy.

A **treelet pair** $t_{1:2}$ is a tuple (t_1, t_2, m) where:

- t_1 and t_2 are treelets for source and target languages (L_1 and L_2) and states (Q_1 and Q_2),
- m is a 1-1 **mapping** between frontier nodes in t_1 and in t_2.

Given a starting **synchronous state** $Start_{1:2} \in Q_1 \times Q_2$, a **synchronous derivation** $\delta = \{t^0_{1:2}, \ldots, t^k_{1:2}\}$ constructs a pair of dependency trees (T_1, T_2) by:

- attaching treelet pairs $t^0_{1:2}, \ldots, t^k_{1:2}$ at corresponding frontier nodes, and
- ensuring that the root states $t^0_{1:2}.q, \ldots, t^k_{1:2}.q$ of the attached treelets pairs $t^0_{1:2}, \ldots, t^k_{1:2}$ match the frontier states of the corresponding frontier nodes.

Note that we differ from Čmejrek (2006) as we require (1) each treelet to contain at least one internal node and (2) all frontier nodes in a treelet pair to be mapped, i.e. the left and right treelets must contain the same number of frontier nodes. These two additional requirements ensure that the translation procedure (1) will not loop (by generating output treelets while not consuming anything from the input tree) and (2) will not skip any subtree of the input tree.

For the purpose of further explanation, we define the **source-side projection** source(δ) and the **target-side projection** target(δ) of a derivation δ as the trees T_1 and T_2 constructed by δ, respectively. Given a source tree T_1, we use $\Delta(T_1) = \{\delta \mid \text{source}(\delta) = T_1\}$ to denote the set of derivations δ yielding T_1 on the source side.

Note that given a tree T, not all subtrees $t \subseteq T$ can be considered as a part of (one side of) a valid (synchronous) derivation because STSG derivations have no adjunction operation. We say that a subtree t of a tree T satisfies the **STSG property**, if for every internal node $n \in t$ all immediate dependents of n in T are included in t as well, either as internal or as frontier nodes. In other words, we assume no tree adjunction operation was necessary to cover any children of n in T.

3.4 STSG in Machine Translation

Our goal is to translate a source sequence of words s_1 into a target sequence of words $\hat{s_2}$, where $\hat{s_2}$ is the most likely translation out of all possible translations s_2:

$$\hat{s_2} = \operatorname*{argmax}_{s_2} p(s_2 \mid s_1) \tag{3.1}$$

We introduce the source and target dependency trees T_1 and T_2 as hidden variables to the maximization, assuming no other dependencies except those along the pipeline indicated in Figure 3.1 (page 54):

$$\hat{s_2} = \operatorname*{argmax}_{s_2,T_1,T_2} p(T_1 \mid s_1) \cdot p(T_2 \mid T_1) \cdot p(s_2 \mid T_2) \tag{3.2}$$

Rather than searching the joint space, we break the search into three independent steps: parsing (3.3), tree transduction (3.4) and generation (3.5):

$$\hat{T_1} = \operatorname*{argmax}_{T_1} p(T_1 \mid s_1) \tag{3.3}$$

$$\hat{T_2} = \operatorname*{argmax}_{T_2} p(T_2 \mid \hat{T_1}) \tag{3.4}$$

$$\hat{s_2} = \operatorname*{argmax}_{s_2} p(s_2 \mid \hat{T_2}) \tag{3.5}$$

We mention the tools used for parsing and generation in Table 3.3 on page 60. STSG is used to find the most likely target tree $\hat{T_2}$ given T_1. Applying the Viterbi approximation we search for the most likely derivation $\hat{\delta}$

$$
\begin{aligned}
\hat{T}_2 &= \operatorname*{argmax}_{T_2} p(T_2 \mid T_1) && \text{marginalize over derivations } \delta \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta} p(T_2, \delta \mid T_1) && \text{apply chain rule} \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta} p(T_2 \mid \delta, T_1) \cdot p(\delta \mid T_1) && p(T_2 \mid \delta, T_1) = 1 \text{ because } T_2 = \text{target}(\delta) \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta} p(\delta \mid T_1) && \text{apply Fundamental Law} \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta} \frac{p(\delta, T_1)}{p(T_1)} && \text{ignore } p(T_1), \text{ constant in maximization} \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta} p(\delta, T_1) && p(\delta, T_1) = \begin{cases} p(\delta) & \text{if } \delta \in \Delta(T_1) \text{ because } T_1 = \text{source}(\delta) \\ 0 & \text{otherwise} \end{cases} \\
&= \operatorname*{argmax}_{T_2} \sum_{\delta \in \Delta(T_1)} p(\delta) && \text{approximate the sum by the largest element only} \\
&\doteq \operatorname*{argmax}_{T_2} \max_{\delta \in \Delta(T_1)} p(\delta) && \text{Viterbi approximation to search for } \delta \text{ instead of } T_2 \\
&\doteq \text{target}(\operatorname*{argmax}_{\delta \in \Delta(T_1)} p(\delta))
\end{aligned}
$$

Figure 3.6: Detailed explanation of why we are searching for the most likely derivation $\hat{\delta}$ instead of the most likely $\hat{T}_2$ given T_1.

instead and take its target-side projection, see Figure 3.6 for a step-by-step justification.

To sum up, the most likely target tree $\hat{T}_2$ given T_1 is found by searching for the most likely synchronous derivation $\hat{\delta}$ that constructs T_1 and $\hat{T}_2$:

$$\hat{T}_2 = \operatorname*{argmax}_{T_2} p(T_2 \mid T_1) \doteq \text{target}(\hat{\delta}) = \text{target}\Big(\operatorname*{argmax}_{\delta \in \Delta(T_1)} p(\delta)\Big) \tag{3.6}$$

As defined above, a derivation δ consists of a sequence of treelet pairs. When searching for $\hat{\delta}$, we thus consider all decompositions of T_1 into a set of treelets $t_1^0, \ldots, t_1^k$, expand each treelet t_1^i into a treelet pair $t_{1:2}^i$ using a treelet pair dictionary and evaluate the probability of the synchronous derivation $\delta = \{t_{1:2}^0, \ldots, t_{1:2}^k\}$. Having found the most likely $\hat{\delta}$, we return the right-hand-side tree $\hat{T}_2$ constructed by $\hat{\delta}$.

3.4.1 Log-linear Model

Following Och and Ney (2002) we further extend 3.6 into a general log-linear framework that allows us to include various features or **models**:

$$\hat{\delta} = \underset{\delta \in \Delta(T_1)}{\operatorname{argmax}} \exp\Big(\sum_{m=1}^{M} \lambda_m h_m(\delta) \Big) \tag{3.7}$$

Each of the M models $h_m(\delta)$ provides a different score aimed at predicting how good the derivation δ is. The weighting parameters λ_m, $\sum_1^M \lambda_m = 1$, indicate the relative importance of the various features and they are tuned on an independent dataset.

To facilitate efficient decoding (see Section 3.4.2 below), we require most feature functions $h_m(\delta)$ to decompose in lockstep with the derivation, i.e. to take the form:

$$h_m(\delta) = \sum_{i=0}^{k} h_m(t^i_{1:2}) \tag{3.8}$$

STSG Model

One of the most basic features is based on the STSG probability of the synchronous derivation. STSG estimates the probability of the derivation δ as the multiplication of probabilities of individual attachments. The probability of each attachment $i = 1 \ldots k$ is defined as the conditional probability of a treelet pair $t^i_{1:2}$ given the synchronous state q of the two frontiers where $t^i_{1:2}$ is attached. The frontiers' state q has to match the root state of the treelet pair $t^i_{1:2}$ so we can write the probability of the attachment as $p(t^i_{1:2} \mid t^i_{1:2}.q)$. Here is the STSG probability of a synchronous derivation:

$$p(\delta) = p(t^0_{1:2} \mid Start_{1:2}) * \prod_{i=1}^{k} p(t^i_{1:2} \mid t^i_{1:2}.q) \tag{3.9}$$

To incorporate this probability into the log-linear model, we take the log of it, defining the STSG model:

$$h_{STSG}(\delta) = \log(p(\delta)) = \log(p(t^0_{1:2} \mid Start_{1:2})) + \sum_{i=1}^{k} \log(p(t^i_{1:2} \mid t^i_{1:2}.q)) \tag{3.10}$$

Note that if $h_{STSG}(\cdot)$ were the only feature used, the log-linear model reduces to the straightforward maximization of $p(\delta)$:

$$\hat{\delta} = \underset{\delta \in \Delta(T_1)}{\operatorname{argmax}} \exp\big(h_{STSG}(\delta)\big) = \underset{\delta \in \Delta(T_1)}{\operatorname{argmax}} p(\delta) \tag{3.11}$$

Reverse and Direct Treelet Models

The STSG model assumes that the choice of a treelet pair $t_{1:2}$ depends only on the synchronous state q of the two frontiers where $t_{1:2}$ is attached.

Inspired by the common practice of statistical machine translation (Och, 2002), we include the channel model ("reverse") and "direct" conditional probabilities:

$$h_{direct}(t^i_{1:2}) = \log\big(p(t^i_2 \mid t^i_1)\big) \tag{3.12}$$
$$h_{reverse}(t^i_{1:2}) = \log\big(p(t^i_1 \mid t^i_2)\big) \tag{3.13}$$

The reverse model is justified by Bayes decomposition of $p(target|source)$[7] while the direct model empirically proves as a comparably valuable source (see e.g. Och (2002)).

N-gram Language Models

A probabilistic target-language model used to promote coherent hypotheses is a very important predictor of translation quality (see e.g. Och (2002)).

Pervasive n-gram language models estimate the probability of a sentence s as the multiplication of probabilities of all n-grams in the sentence:

$$p(s) = \prod_{i=1}^{\mathrm{length}(s)} p(w_i|w_{i-1}, \ldots, w_{i-n+1}) \tag{3.14}$$

where w_i is each word in the sentence and $w_{i-1}, \ldots, w_{i-n+1}$ are $(n-1)$ preceding words.

In the canonical mode, an STSG decoder is expected to produce an output dependency tree and thus cannot directly employ n-gram language models. However, if no structure is needed at the output (e.g. when translating to a-trees and directly reading off node labels), we can safely destroy all target-side tree structure, representing T_2 as a sequence of output words $w_1, \ldots, w_J$. Naturally, until the complete target hypothesis is constructed, we have to keep track of exact positions of yet-to-expand frontiers within the sequence of output words.

In this special case, the traditional sequence (language) model can be used, with a bit of careful delayed computation around unexpanded frontiers:

[7] $$p(target|source) = \frac{p(target)}{p(source)} p(source|target)$$

$$h_{\mathrm{LM}_n}(\delta) = \log \prod_{j=1}^{J} p(w_j | w_{j-1} \dots w_{j-n+1}) \tag{3.15}$$

We assume w_j to be set to a special out-of-sentence symbol for $j < 1$.

Binode Tree Language Model

Given an output dependency tree structure, a more natural language model estimates the probability of the sentence based on edges in the tree. As documented e.g. by Charniak (2001), such models can improve parsing accuracy.

We define binode probability of the target tree T_2 as the multiplication of probabilities of all the edges $e \in T_2$. Given the governor $g(e)$ and the child $c(e)$ of e, we can define three different probabilities, "direct", "reverse" and "joint", leading to three separate models:

$$h^{biLM}_{direct}(\delta) = \log \prod_{e \in T_2} p(g(e) \mid c(e)) \tag{3.16}$$

$$h^{biLM}_{reverse}(\delta) = \log \prod_{e \in T_2} p(c(e) \mid g(e)) \tag{3.17}$$

$$h^{biLM}_{joint}(\delta) = \log \prod_{e \in T_2} p(c(e), g(e)) \tag{3.18}$$

Additional Features

Following the common practice in phrase-based machine translation (e.g. Koehn (2004a) or Zens *et al.* (2005)), we include penalties to consider the number of treelets and words used to construct a derivation:

$$h_{treelet\ penalty}(\delta) = -|\delta| \tag{3.19}$$

$$h_{word\ penalty}(\delta) = -\sum_{i=0}^{k} |t_2^i| \tag{3.20}$$

where $|t_2^i|$ denotes the number of internal nodes in target treelet t_2^i.

3.4.2 Decoding Algorithms for STSG

The search space of all possible decompositions of input tree multiplied by all possible translations of source treelets is too large to be explored in full, efficient approximation algorithms have to be designed.

Top-Down Beam Search

The current version of our decoder implements a beam search inspired by the strategy of phrase-based decoder Moses (Koehn *et al.*, 2007). While Moses constructs partial hypotheses in a left-to-right fashion (picking source phrases in arbitrary order), our partial hypotheses are constructed top-to-bottom along with the source tree T_1 being covered from top to bottom. The algorithm, in essence very similar to the one described recently by Huang *et al.* (2006) but dating back to Aho and Johnson (1976), is outlined in Alg. 2. The main difference is that we tackle the exponential search space of tree decompositions using a pre-processing phase while Huang *et al.* (2006) use memoization.

Algorithm 2 Top-down beam-search STSG decoding algorithm.

1. For an input tree T_1 of n nodes, prepare the translation options table:
2. For each source node $x \in T_1$
3. Construct all possible treelet pairs $t_{1:2}$ where t_1 is rooted at x and covers a subtree of T_1.
4. The subtree has to satisfy the STSG property:
5. If $y \in T_1$ is covered with an internal node of t_1, all dependents of y have to be covered by t_1 as well.
6. Record only τ best possible treelet pairs rooted at x.
7. Create stacks $s_0, \ldots, s_n$ to hold partial hypotheses, stack s_i for hypotheses covering exactly i input nodes.
8. Insert the initial hypothesis (a single frontier node) into s_0.
9. For $i \in 0 \ldots n-1$
10. For each hypothesis $h \in s_i$
11. Expand h by attaching one of possible translation options at a pair
12. of pending frontiers, extending the set of covered words and
13. adding output words.
14. Insert the expanded h' (j words covered) to s_j.
15. Prune s_j if contains more that σ hyps.
16. Output the top-scoring h^* from s_n.

The first step is the construction of "translation options". For each input node $x \in T_1$, all possible treelets rooted at x are examined and if a translation of a treelet is found, it is stored as one of the translation options for x. Figure 3.7 illustrates sample translation options for the auxiliary root ("#"), the main verb "said" and the full stop ".". For conciseness, the target treelet structure is omitted in the picture as if the target output tree was directly linearized.

Figure 3.8 illustrates the second and main step, i.e. the gradual expansion

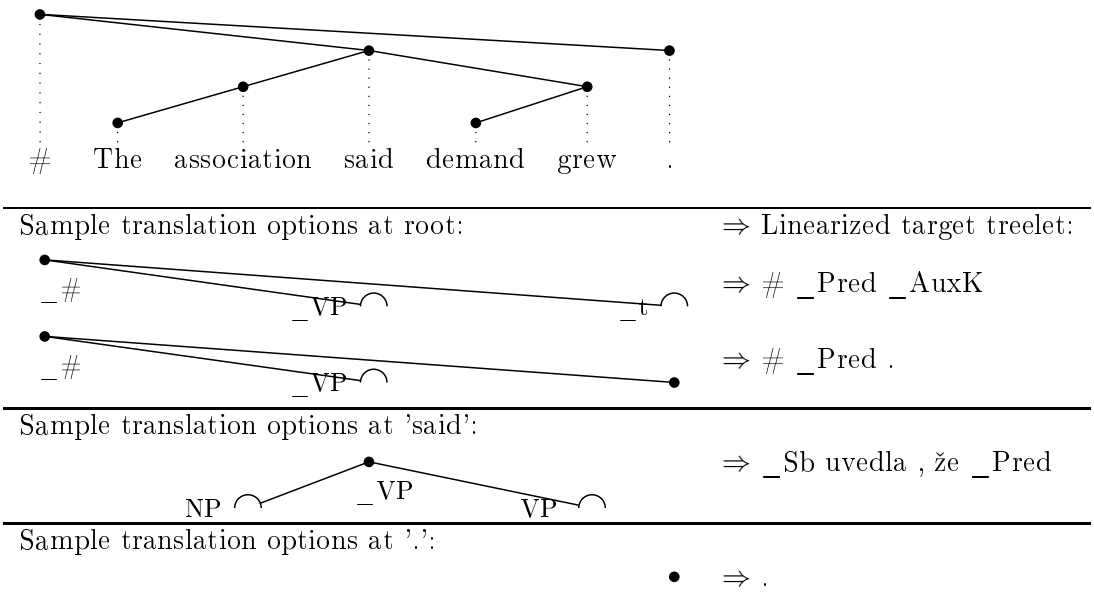


Figure 3.7: Sample translation options for translating an English a-tree to a Czech a-tree. The target structure is immediately linearized.

of a hypothesis using translation options constructed in the first step. Once all input nodes are covered (and thus no frontiers are left in the partial output), the output hypothesis is returned. In practice, we beam-search the space of derivations, studying only σ best-scoring partial hypotheses of the same number of covered input nodes. Note that each expansion is guaranteed to cover at least one more input node, so the algorithm cannot loop.

Bottom-up Dynamic-Programming Decoding Algorithm

Čmejrek (2006) presents another possible method of searching for the most probable translation T_2 of a given input tree T_1.

The most probable derivation is computed by a dynamic-programming style Alg. 3. For each node $c_1 \in T_1$ in bottom-up order and for each synchronous state $q \in Q$, we find and store the root treelet pair $t_{1:2}$ of the most probable derivation $\hat{\delta_{c_1}}$ that covers the whole subtree of T_1 rooted at c_1 and has q as the root synchronous state. The treelets are stored in arrays $A_{c_1}(q)$ and the corresponding probabilities of $\hat{\delta_{c_1}}$ are stored in $\beta_{c_1}(q)$.

The final derivation $\hat{\delta}$ covering whole T_1 is constructed by starting from $t^0_{1:2} = A_{T_1.r}(Start_{1:2})$ and recursively including all treelet pairs $t^i_{1:2} = A_{f^i_1}(q^i)$ to cover all frontiers f^i_1 (respecting the synchronous states q^i) of previously included treelets $t^0_{1:2}, \ldots, t^{i-1}_{1:2}$.

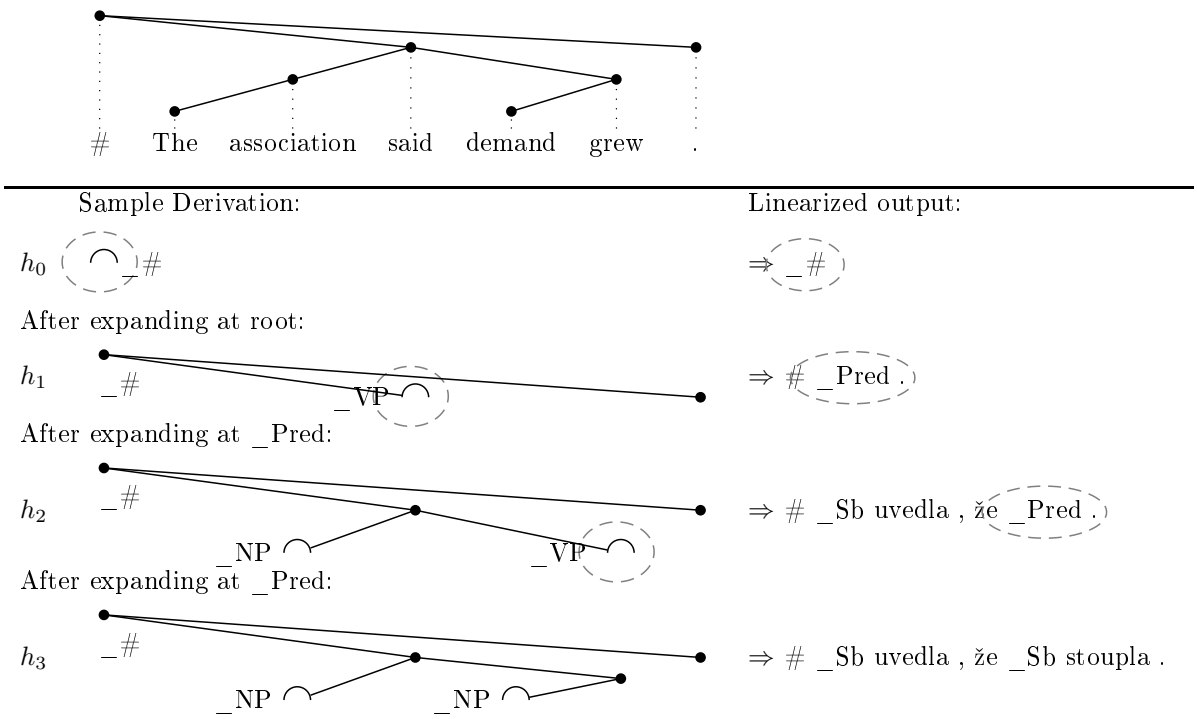


Figure 3.8: Top-down hypothesis expansion using translation options from Figure 3.7. Dashed circles indicate where treelet pairs are attached at each step.

3.5 Heuristic Estimation of STSG Model Parameters

Given a sentence-parallel treebank, we can use the expectation-maximization algorithm described by Čmejrek (2006) to obtain treelet-to-treelet alignments and estimate STSG derivation probability as defined in Eq. 3.9. Our plan is to soon adopt this method, but for the time being we restrict our training method to a heuristic based on GIZA++ (Och and Ney, 2000) word alignments. So instead of treelet-to-treelet alignments, we base our probability estimates on node-to-node alignments only.

For each tree pair in the training data, we first read off the sequence of node labels and use GIZA++ tool to extract a possibly N-N node-to-node-alignment.[8] In the next step, we extract all treelet pairs from each node-aligned tree pair such that all the following conditions are satisfied:

- each treelet may contain at most 5 internal and at most 7 frontier nodes (the limits are fairly arbitrary),

[8] GIZA++ produces asymmetric 1-N alignments, we follow standard practices to combine 1-N and N-1 alignments from two GIZA++ runs.

Algorithm 3 Bottom-up decoding algorithm for *STSG*.

```
1. for each node c_1 ∈ T_1.V in bottom-up order
2.     for each q ∈ Q let β_{c_1}(q) = −∞
3.     for each treelet t_1 that fits c_1 in a safe order
4.         while t_{1:2}=proposeNewTreeletPair(t_1)
5.                 // we have to try all possible t_2, q, m, s
6.             let prob = p(t_{1:2} | t_{1:2}.q) · ∏_{(d_1,d_2)∈m} β_{d_1}(t_{1:2}.s((d_1, d_2)))
7.             if β_{c_1}(q) < prob // found a higher scoring derivation
8.             then let β_{c_1}(q) = prob and A_{c_1}(q) = t_{1:2}
```

- each internal node of each treelet, if aligned at all, must be aligned to a node in the other treelet,
- the mapping of frontier nodes has to be a subset of the node-alignment,
- each treelet must satisfy STSG property.

All extracted treelet pairs contribute to our maximum likelihood probability estimates. In general, given a left treelet t_1, a right treelet t_2 and their respective root states q_1 and q_2, we estimate three separate models: "stsg", "direct" and "reverse":

$$h_{stsg}(t_{1:2}) = \log \frac{\text{count}(t_1, q_1, t_2, q_2)}{\text{count}(q_1, q_2)} \tag{3.21}$$

$$h_{direct}(t_{1:2}) = \log \frac{\text{count}(t_1, q_1, t_2, q_2)}{\text{count}(t_1, q_1, q_2)} \tag{3.22}$$

$$h_{reverse}(t_{1:2}) = \log \frac{\text{count}(t_1, q_1, t_2, q_2)}{\text{count}(t_2, q_1, q_2)} \tag{3.23}$$

3.6 Methods of Back-off

As expected, and also pointed out by Čmejrek (2006), the additional structural information boosts data-sparseness problem. Many source treelets in the test corpus were never seen in our training data. To make things worse, our heuristic treelet extraction method constrains the set of extractable treelet pairs by three rigid structures: source tree, target tree and the word alignment. A single error in the word alignment or parsing prevents our method from learning a treelet pair. We thus have to face not only natural divergence of sentence structures but also divergence caused by random errors in any of the automatically obtained annotations.

To tackle the problem, our decoder utilizes a sequence of back-off models, i.e. a sequence of several methods of target treelet construction and probability estimation. Each subsequent model is based on less fine-grained description of the input treelet and constructs the target treelet on the fly from independent components.

The order and level of detail of the back-off methods is fixed but easily customizable in a configuration file.

3.6.1 Preserve All

The most straightforward method is to preserve all information in an observed treelet pair. This includes:

- left and right treelet structure, including all frontiers and internals and preserving the linear order of the nodes
- full labels of left and right internals
- state labels of left and right frontiers

An example of a complete treelet pair is given in Figure 3.9.

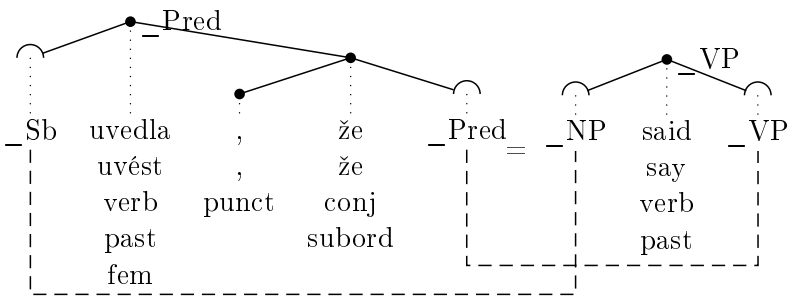


Figure 3.9: A treelet pair with all information preserved.

3.6.2 Drop Frontiers

One of significant limitations of STSG is the lack of adjunction operation. In order to handle input treelets with branching that was not seen in the training data, we collect treelet pairs while ignoring any frontiers. An example of such treelet pair is given in Figure 3.10.

Once the translation using this model is attempted, we remove all frontiers from the source treelet, map the "skeleton" to the target treelet and attach

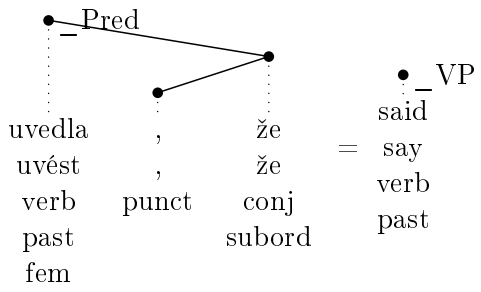


Figure 3.10: A treelet pair with no frontiers.

the required number of frontier nodes to the target tree. The position and state label of the frontiers is chosen based on a separate probabilistic model.

As a further refinement, one might think of dropping only frontiers representing adjuncts but preserving frontiers for complements. Either a valency lexicon would supply the distinction between argument and adjuncts, or we could use some heuristic such as suggested by Bojar (2004).

In the current implementation, we employ this method of back-off only in cases where the output is directly linearized. Therefore, the governing node for a frontier has not to be determined when attaching the frontier and we can use a simple model to "zip" the sequence of target internals and the sequence of target frontiers (we do not allow any reordering of the frontiers). The target label of a frontier is chosen based on the label of the source frontier.

3.6.3 Translate Word by Word

The technique of dropping frontiers cannot be used when producing output trees, unless we design a frontier re-attachment model for output treelets. However, we still need to overcome the no-adjunction limitation of STSG in this setting. A simple solution is possible, if we restrict treelet size to one internal only.

If the source treelet contains exactly one internal node, the structure of the treelet is known: the internal node is the root of the treelet and its immediate dependents are all frontiers of the treelet, see e.g. Figure 3.11.

We can easily decompose such treelets and translate independently: 1. the label of the internal node, 2. each of the frontier labels. Again, we could consider reordering of the nodes but until a satisfactory reordering model is designed, we keep the order intact.

A clear disadvantage of this back-off method is that the number of nodes cannot change in the process of translation. This poses a significant problem

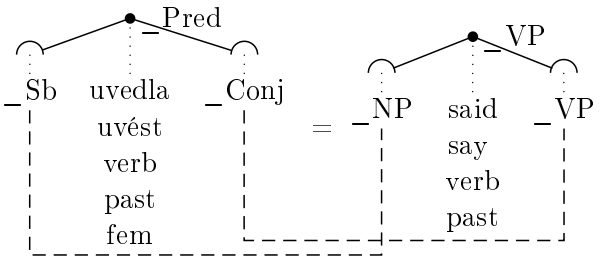


Figure 3.11: A treelet pair with one internal node in each treelet.

for transfer at the a-layer, but for transfer at the t-layer, preserving tree structure is a viable approximation (Čmejrek *et al.*, 2003).

3.6.4 Keep Word Non-Translated

In the cases where a word was never seen in the training data, the methods described so far would not provide any translation for the word, so the translation of the whole sentence would fail producing no output. As a back-off, one can either try to look up the word in a translation dictionary (possibly facing the issue of a different morphological form) or, as an ultimate rescue, keep the unknown word not translated and try to translate the rest of the sentence.

Technically, we achieve this by adding a special rule that preserves the treelet structure, copies internal labels and independently translates each of frontier labels. In practice, we prefer to restrict this method to treelets containing one internal only.

3.6.5 Factored Input Nodes

As described e.g. in Mikulová *et al.* (2006), and also indicated in Figure 3.9, internal node labels are usually not atomic values. For example, an a-node usually bears the value of word form, lemma, morphological tag (all inherited from the m-layer) and analytical function (afun) label. For t-nodes, the set of attributes is significantly larger, as attributes explicitly encode linguistic features such as verbal tense, modality, iterativeness, person, nominal gender, negation and many others.

Treating node labels as atomic and thus relying on all attributes to exactly match the input leads to severe sparse data problem. We allow to specify only a subset of input attributes ("factors") to be taken into account while searching for a treelet translation. In practice, we usually use a sequence of models, each depending on fewer and fewer input factors. For example, a

back-off model for "preserve all" as illustrated in Figure 3.9 could be based on source lemmas only. See Figure 3.12 for a hypothetical rule for Czech-to-English transfer.

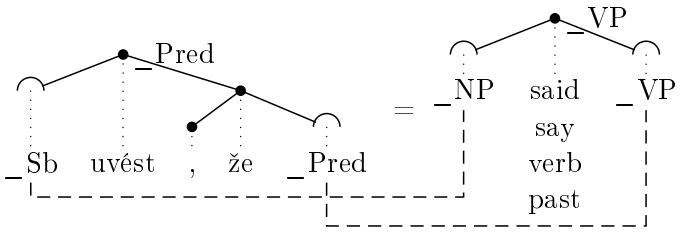


Figure 3.12: A treelet pair with source lemmas only.

3.6.6 Factored Output Nodes

Ignoring some attributes of input nodes is not sufficient as a back-off method alone. For output factors, we have no option and eventually each node has to be provided with all relevant attributes. We use the idea of "mapping" and "generation" steps from factored phrase-based translation (Koehn and Hoang, 2007), details of which are summarized in Section 4.2.4 below.

Currently, our implementation of factored models is limited to treelets containing exactly one internal. We will soon extend this to treelets of any size. However, the size and shape of the treelet (chosen according to a subset of input factors) will remain fixed until all additional output factors are constructed.

Figure 3.13 illustrates a sequence of five **decoding steps**: three **mapping steps** that convert source factors to target factors and two **generation steps** that ensure coherence of output factors. For instance, the Czech word form is translated to an English form in the first step. An independent second step translates the lemma. The third step takes all source morphological attributes and translates them to target morphological attributes. The coherence of the choices is ensured in steps 4 and 5 that bind together the output form with the lemma (4) and the form with the morphological attributes (5). It should be noted that many other configurations are possible.

In setups with multiple output factors, we apply also the language models described in Section 3.4.1 and Section 3.4.1 several times using various subsets of output factors to provide a back-off for probability estimation. For instance, even if a node pair was never seen in the exact configuration constructed in a sequence of decoding steps, the pair of node lemmas may be quite common so we wish to score it with a non-zero probability.

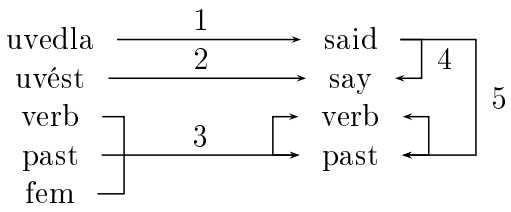


Figure 3.13: Sample decoding steps in word-for-word factored translation.

3.7 Remarks on Implementation

The STSG decoder called `treedecode` is being implemented in Mercury (Somogyi *et al.*, 1995)[9] and currently consists of about 17,000 lines of code.

Supported features, apart from methods described in previous sections, include:

- parallel execution (both training and translation phases) on Sun Grid Engine[10],
- efficient storage of translation tables using `tinycdb`[11],
- binding to IrstLM (Federico and Cettolo, 2007) for n-gram language modelling,
- disk caching of various steps of computation to speed up consecutive startups and reuse partial results upon failure (similar effects can be achieved using the technique of "checkpointing"),
- basic debugging output in Scalable Vector Graphics (SVG),
- preliminary support for minimum error-rate training using two approaches, (Och, 2003) and (Smith and Eisner, 2006a).

The source code is currently available upon request, future versions will be freely accessible on a website, released as one of the deliverables of the EuroMatrix project.

[9] `http://www.cs.mu.oz.au/research/mercury/`
[10] `http://gridengine.sunsource.net/`
[11] `http://www.corpit.ru/mjt/tinycdb.html`

3.8 Evaluating MT Quality

Estimating quality of machine translation is difficult because of many relevant criteria (e.g. output fluency or faithfulness of translation, see e.g. Dorr *et al.* (1998)) and also because many variations can be equally acceptable. Moreover, human evaluation is subjective and thus difficult to replicate for similarly performing systems unless a very large collection of judgements is created, not to mention the cost of such an evaluation.

For the daily routine of MT systems development, many automatic metrics have been proposed. Here we use one of the most common metrics, BLEU (Papineni *et al.*, 2002). Although there are metrics that achieve better correlation with humans (Callison-Burch *et al.*, 2007), such metrics are target-language dependent and have not been adapted for Czech yet.

Please note that neither absolute BLEU scores nor relative improvements are comparable unless evaluated on the very same set of source sentences and reference translations. The results reported here for English-to-Czech are thus by no means comparable to e.g. Czech-to-English MT by Bojar *et al.* (2006) or Čmejrek *et al.* (2003) evaluated on a different test set and against 4 reference translations instead of just one used here. See Bojar *et al.* (2006) for a fair comparison of those two experiments that also highlights the influence of rather subtle manipulations with the reference translations or simple rules fixing tokenization issues to significantly raise BLEU scores.

We estimate empirical confidence bounds using the bootstrapping method described by Koehn (2004b): Given a test set of sentences, we perform 1,000 random selections with repetitions to estimate 1,000 BLEU scores on test sets of the same size. The empirical 90%-confidence upper and lower bounds are obtained after removing top and bottom 5% of scores. For conciseness, we report the average of the distance between the standard BLEU value and the empirical upper and lower bound after the "±" symbol.

3.9 Empirical Evaluation of STSG Translation

In an end-to-end evaluation, we try to cover a wide range of experimental settings when translating from English to Czech, as illustrated in Figure 3.14, which is a refinement of Figure 3.1.

Our main focus is the translation from the English t-layer to the Czech t-Layer (etct). The general applicability of STSG to any dependency trees allows us to test the same model also for analytical translation (eaca) or across the layers (etca and eact). To a certain extent, our tree-based decoder can simulate a direct approach to MT (phrase-based decoding, as will be

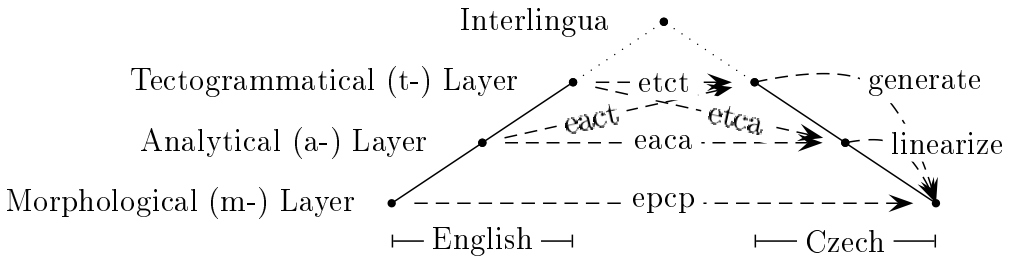


Figure 3.14: Experimental settings of syntactic MT.

discussed in Chapter 4) if we replace the dependency structure of an a-tree with a simple left-to-right chain of words ("linear tree"). The results obtained using this approach are labelled "epcp". Our phrase-based approximation epcp is bound to work worse than other phrase-based systems because we strictly follow the left-to-right order prohibiting any phrase reordering.

For each configuration, we extract treelet pairs using the heuristics described in Section 3.5, possibly employing some of the back-off techniques from Section 3.6. The EM training procedure as described by Čmejrek (2006), was not yet incorporated into our training process.

3.9.1 Experimental Results

Apart from our STSG decoder, we use several additional tools along the training and translation pipeline, as summarized in Section 3.1.4. We train our system on the Project Syndicate section of CzEng 0.5 (Bojar and Žabokrtský, 2006) (also called News Commentary corpus) and test it using the standard sets available for the ACL 2007 workshop on machine translation (WMT07[12]).

Table 3.4 reports the BLEU (Section 3.8) scores of several configurations of our system, higher scores suggest better MT quality. We report single-reference lowercased BLEU.[13]

The values in the column "LM Used" indicate the type of language model used in the experiment. An n-gram model can be applied to the output sequence of words. For setups where the final sequence of words is constructed using the generation component by Ptáček and Žabokrtský (2006) with no

[12] `http://www.statmt.org/wmt07/`

[13] For methods using the t→text generation system by Ptáček and Žabokrtský (2006), we tokenize the hypothesis and the reference using the rules from the official NIST `mteval-v11b.pl` script. For methods that directly produce sequence of output tokens, we stick to the original tokenization.

Method of Transfer	LM Used	BLEU
epcp	n-gram	10.9±0.6
eaca	n-gram	8.8±0.6
epcp	none	8.7±0.6
eaca	none	6.6±0.5
etca	n-gram	6.3±0.6
etct factored, preserving structure	binode	5.6±0.5
etct factored, preserving structure	none	5.3±0.5
eact, no output factors	binode	3.0±0.3
etct, vanilla STSG (no factors), all node attributes	binode	2.6±0.3
etct, vanilla STSG (no factors), all node attributes	none	1.6±0.3
etct, vanilla STSG (no factors), just t-lemmas	none	0.7±0.2

Table 3.4: English-to-Czech BLEU scores for syntax-based MT evaluated on DevTest dataset of ACL 2007 WMT shared task.

access to a language model, we use at least a binode LM to improve output tree coherence.

Appendix A provides examples of MT output from our "etct" method as well as from phrase-based systems described in Chapter 4.

3.10 Discussion

At the first sight, our preliminary results support common worries that with a more complex system it is increasingly difficult to obtain good results. However, we are well aware of many limitations of our current experiments as discussed below.

Within the scope of our main focus, the tectogrammatical transfer ("etct"), we see a dramatic improvement from BLEU 1.6 to BLEU 5.6. The score 1.6 is achieved using the very baseline of STSG translation: nodes including all attributes are treated as atomic units, only the maximum likelihood estimate of STSG probability (Section 3.4.1) is used and no language model is applied. Our best "etct" result scoring 5.6 uses a combination of back-off methods, including factored input and output nodes and two binode models (one less fine-grained, again as a means of back-off).

3.10.1 BLEU Favours n-gram LMs

BLEU is known to favour methods employing n-gram based language models. Empirical evidence supporting the claim can be observed in Table 3.4: an

n-gram LM gained 2 BLEU points for both "eaca" and "epcp".

In future experiments we plan to attempt two ways to tackle the problem: employing some LM-based rescoring even after the generation component (Ptáček and Žabokrtský, 2006), as well as using other automatic metrics of MT quality instead of BLEU to avoid the bias.

3.10.2 Cumulation of Errors

All components in our setup deliver only the single best candidate. Any errors will therefore accumulate over the whole pipeline. This primarily hurts the "etct" scenario where all our tools are employed.

In future, we would like to pass and accept several candidates, allowing each step in the calculation to do any necessary rescoring.

3.10.3 Conflict of Structures

Our current heuristic method of treelet extraction (Section 3.5) crucially depends on the quality of both English and Czech trees as well as the node alignment between them. A single error in any of the rigid sources may prevent the extraction of a treelet pair, not to mention natural divergence between the sentence and its translation. Precisely this reason explains the loss of performance of "eaca" compared to "epcp".

We hope that using the EM procedure (Čmejrek, 2006) will gain some recall. The current heuristic method can be also modified to accept a certain level of structure divergence, such as a certain portion of node-alignments leading out of the treelet pair. Alternatively, one could obtain not just the single best source and target tree, but a set of candidates[14] and choose such a pair of trees that matches best with the node alignments.

Ultimately, the solution lies in designing additional back-off techniques that can accommodate natural divergence appearing in Czech and English training sentences and still exploit most of the data.

Smith and Eisner (2006b) attempt to loosen the rigidity of STSG structures by defining quasi-synchronous (monolingual) grammar for target language that prefers to analyse or generate target-side sentence in alignment with the source-side tree but is not restricted to do so.

Successful syntax-based approaches to MT, e.g. Quirk *et al.* (2005) or Huang *et al.* (2006), benefit from the fact that the syntactic structure comes only from one language and is only projected to the other language according

[14] For dependency parsing, an efficient k-best parser was recently implemented by Hall (2007).

to word alignments. Although linguistic adequacy of the projected tree might suffer, much fewer structural conflicts are observed.

3.10.4 Combinatorial Explosion

In the current implementation, target-side treelets are fully built during the preparatory phase of translation option generation. Uncertainty in the many t-node attributes leads to too many treelets with insignificant variations while e.g. different lexical choices are pushed off the stack. While vital for final sentence generation (see Table 3.4), fine-grained t-node attributes should be produced only once all key structural, lexical and form decisions have been made.

3.10.5 Sentence Generation Tuned for Manual Trees

The rule-based generation system (Ptáček and Žabokrtský, 2006) was designed to generate Czech sentences from full-featured manual Czech tectogrammatical trees from the (monolingual) PDT.

Our target-side training trees are the result of an automatic analytical and tectogrammatical parsing procedure as implemented by McDonald *et al.* (2005) and Klimeš (2006); Žabokrtský (2008a), resp. Further noise is added during the tree transfer, so our final input to the generation component contains random errors in tree structure as well as missing or bad attribute values.

As the manual annotation of PCEDT 2.0 proceeds, we may be able to train the transfer system on manual Czech trees. Simultaneously, the generation component will be improved to be more robust towards malformed input.

3.10.6 Errors in Source-Side Analysis

For the purpose of source-side English analysis, we still rely on very simple rules similar to those used by Čmejrek *et al.* (2003) to convert Collins (1996) parse trees to analytical and tectogrammatical dependency trees.

We hope the English-side pipeline can be improved using recent taggers and parsers. Furthermore, the tectogrammatical analysis of English will be refined as manual English t-trees become available during PCEDT 2.0 annotation, in progress.

Alternatively, we might include some attributes based directly on a-trees in the source t-trees. This would serve as a back-off in case the a→t rules fail to provide all necessary information.

3.10.7 More Free Parameters

Last but not least, the more complex the setup is ("etct" being our most complicated design), the more free parameters there are in the system to configure. We have already mentioned many ways of replacing individual components, e.g. the parser applied or the method of treelet dictionary extraction. Moreover, each of the components in the pipeline has many options to tune its behaviour.

Despite not reflected in the error-bar figures in Table 3.4, which describe the variance due to randomness in input data, we suggest that the variance or rather room for improvement due to sub-component selection and configuration is much greater for more complex scenarios.

It is an open software engineering and management question which of the free parameters or which of the methods should be further studied.

Another drawback of the complex model is the abundance of model parameters (λ_m in the log-linear model, Section 3.4.1). The optimization method commonly used to set the parameters, so called minimum-error rate training (Och, 2003), does not converge in our setup so we stick to a default: all models equally important.

3.10.8 Related Research

More or less direct comparison can be made with the system TectoMT developed by Žabokrtský (2008b). TectoMT also uses t-layer for the transfer but instead of a generic formal model, a sequence of many heuristic steps is used. Some of the heuristics rely on probabilistic data such as a bilingual dictionary extracted automatically from CzEng 0.7, but most are rather straightforward deterministic procedures. This approach allows TectoMT to fully exploit the similarity of English and Czech t-layers and avoid the combinatorial explosion our system faces. See Table 4.5 on page 97 for human evaluation scores of TectoMT compared to phrase-based systems and Appendix A for examples and BLEU scores of "etct", TectoMT and other systems.

A method closely related to our STSG is reported by Riezler and John T. Maxwell (2006) who extract parallel snippets of LFG analyses. Their system outperforms phrased-based translation (as rated by two human judges) in a very restricted setting: the test set contains only sentences of 5 to 15 words. 44% of such sentences fall within the coverage of the core LFG grammar and human judges evaluated (a sample of) these 44% sentences. When evaluated with NIST (Doddington, 2002), an automatic n-gram-based metric similar to BLEU, phrase-based translation appears insignificantly better on the 44% in-coverage sentences and significantly better on the full test set

where a back-off LFG grammar had to be used. We can draw the following conclusion: if a sentence can be parsed by the core LFG grammar, it will be probably better translated by the grammar-based system. This fortunate and determinable occasion happens on average in 44% of sentences of 5 to 15 words; for other sentences, a phrase-based system should be used. Another possible interpretation of the experiment is that while the core of their LFG grammar allowed to achieve better translation quality, the back-off grammar was not observed to generalize better than a phrase-based system (Chapter 4) does.

3.11 Conclusion

The previous Chapter 2 was devoted to automatic acquisition of syntactic lexicons, which can serve as an valuable resource for many NLP applications. Interested in applicability of the lexicons in practice, we chose one particular task in this chapter: English-to-Czech machine translation.

We briefly reviewed approaches to MT and summarized a mathematical model of tree transformations (STSG) that fits nicely in the framework of FGD. The model is applied to convert the dependency analysis of a source sentence into the dependency analysis of the sentence in a target language.

We designed a decoding algorithm to search for the most probable translation of an input tree and implemented a first version of the decoder. Several methods of back-off have been proposed and included in the implementation. Finally, the whole pipeline of the translation process has been set up and allows for an end-to-end evaluation.

We did not get to the point where we could directly incorporate a valency lexicon into the transfer step, apart from the t-to-surface generation system (Ptáček and Žabokrtský, 2006) that uses VALLEX to choose an appropriate morphological realization of verb modifiers. However, the treelet pairs described in Section 3.2 can be seen as a form of bilingual valency frames and it would be quite straightforward to design a valency language model similar to the binode model (Section 3.4.1) promoting translations where output valencies are confirmed by the lexicon.

The empirical evaluation (Section 3.9) reveals more important problems than the lack of a valency lexicon in the transfer: the more complex setup is used, the worse results are obtained. We discussed the problems, known limitations and many open questions in Section 3.10. We also pointed out that a more complex system has more free parameters to tune and thus a greater potential for an improvement. We have to leave this for future research.

As our empirical results indicate, the current best scores were obtained using a simple phrase-based approach. That is why we explore this direct method of MT in the following chapter.

Chapter 4

Improving Morphological Coherence in Phrase-Based MT

The previous chapter was devoted to a study of a deep-syntactic MT system and one of its components, tree-to-tree transfer, in particular. Completely reversing our research priorities, we now tackle the task of MT in a very direct end-to-end fashion, employing very little of linguistic analysis.

4.1 Introduction

Best empirical results in MT are currently achieved by phrase-based systems for many language pairs.[1] Known limitations of phrase-based MT include worse quality when translating to morphologically rich languages as opposed to translating from them (Koehn, 2005) and worse grammatical coherence of longer sentences.

We participated in the 2006 summer engineering workshop at Johns Hopkins University[2] that attempted to tackle these problems by introducing separate **factors** in MT input and/or output to allow explicit modelling of the underlying language structure. The support for factored translation models was incorporated into the Moses open-source MT system[3]. Our contribution to the workshop was the design of factors improving English-to-Czech translation.

In this chapter, we discuss the experiments, focusing on one particular aspect, namely the morphological coherence of phrase-based MT output. After a brief overview of factored phrase-based MT (Section 4.2), we summarize some possible translation scenarios in Section 4.4. Section 4.5 studies the level of detail useful for morphological representation and Section 4.6 compares the results to a setting with more data available, albeit out of domain.

[1] `http://www.nist.gov/speech/tests/mt/`

[2] `http://www.clsp.jhu.edu/ws2006/`

[3] `http://www.statmt.org/moses/`

Section 4.7 provides human evaluation of our systems and Section 4.8 is devoted to a brief analysis of MT output errors.

4.1.1 Motivation for Improving Morphology

As documented in Table 3.1 on page 58, Czech has very rich morphology. The Czech morphological system (Hajič, 2004b) defines 4,000 tags in theory and 2,000 were actually seen in a big tagged corpus. (For comparison, the English Penn Treebank tagset contains just about 50 tags.) In our parallel corpus (see Section 3.1.4), the English vocabulary size is 35k distinct token types but more than twice as big in Czech, 83k distinct token types.

As we will see in the following overview of factored phrase-based MT, the model is designed to directly handle any information that corresponds 1-1 to input or output words. For morphological information, this is indeed the case (every input word form can have a lemma and a morphological tag attached), so we can hope the model will make best use of this information.

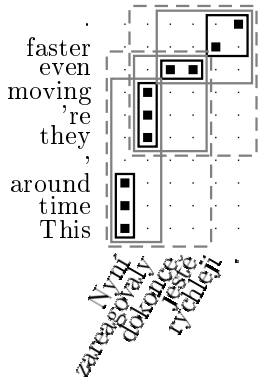
To further emphasize the importance of morphology in MT to Czech, we can compare the standard BLEU (Section 3.8) of a baseline phrase-based translation with BLEU which disregards word forms (a lemmatized MT output is compared to the lemmatized reference translation). The lemmatized BLEU represents MT quality if morphological errors are not penalized at all. The comparison gives us the theoretical margin for improving MT quality by choosing more appropriate word forms (but leaving word order and lexical selection intact). The margin amounts to about 9 BLEU points: the same MT output scores 12 points in standard BLEU and 21 points in lemmatized BLEU.

4.2 Overview of Factored Phrase-Based MT

4.2.1 Phrase-Based SMT

In statistical MT (SMT), the goal is to translate a source (foreign) language sentence $f_1^J = f_1 \dots f_j \dots f_J$ into a target language (Czech) sentence $c_1^I = c_1 \dots c_j \dots c_I$. In phrase-based SMT (e.g. Koehn (2004a), Zens *et al.* (2005)), the assumption is made that the target sentence can be constructed by segmenting source sentence into K phrases[4], translating each phrase and finally composing the target sentence from phrase translations. See Figure 4.1 for an example of phrases automatically extracted from a word-aligned sentence pair. We denote the segmentation of the input sentence into K phrases

[4]Is should be noted that the term "phrases" refers merely to a sequence of words and is not related to linguistically grounded phrases from e.g. Chomskian grammars.



faster	=	rychleji
.	=	.
This time around	=	Nyní
they 're moving	=	zareagovaly
even	=	dokonce ještě
faster .	=	rychleji .
This time around, they 're moving	=	Nyní zareagovaly
even faster	=	dokonce ještě rychleji
...	...	...

Figure 4.1: Sample word alignment and sample phrases consistent with it (not all consistent phrases have been marked).

as s_1^K. Among all possible target language sentences, we choose the sentence with the highest probability:

$$\hat{c}_1^{\hat{I}} = \underset{I,c_1^I,K,s_1^K}{\operatorname{argmax}} \{Pr(c_1^I|f_1^J, s_1^K)\} \tag{4.1}$$

4.2.2 Log-linear Model

In a log-linear model (Och and Ney, 2002), the conditional probability of c_1^I being the translation of f_1^J under the segmentation s_1^K is modelled as a combination of independent feature functions $h_1(\cdot,\cdot,\cdot), \ldots, h_M(\cdot,\cdot,\cdot)$ describing the relation of the source and target sentences:

$$Pr(c_1^I|f_1^J, s_1^K) = \frac{\exp(\sum_{m=1}^M \lambda_m h_m(c_1^I, f_1^J, s_1^K))}{\sum_{c_1^{I'}} \exp(\sum_{m=1}^M \lambda_m h_m(c_1^{I'}, f_1^J, s_1^K))} \tag{4.2}$$

The denominator in 4.2 is used as a normalization factor that depends on the source sentence f_1^J and the segmentation s_1^K only and is omitted during maximization. The model scaling factors λ_1^M are trained either to the maximum entropy principle or optimized with respect to the final translation quality measure.

In our experiments, we use the minimum-error rate training (MERT, (Och, 2003)) tuned to highest BLEU scores using a separate heldout set of data.

4.2.3 Phrase-Based Features

Most of our features are phrase-based and we require all such features to operate synchronously on the segmentation s_1^K and independently of neighbouring

segments. In other words, we restrict the form of phrase-based features to:

$$h_m(c_1^I, f_1^J, s_1^K) = \sum_{k=1}^{K} \tilde{h}_m(\tilde{c}_k, \tilde{f}_k) \tag{4.3}$$

where $\tilde{f}_k$ represents the source phrase and $\tilde{c}$ represents the target phrase k given the segmentation s_1^K.

4.2.4 Factored Phrase-Based SMT

In factored SMT, source and target words f and c are represented as tuples of F and C **factors**, resp., each describing a different aspect of the word, e.g. its word form, lemma, morphological tag, role in a verbal frame. The process of translation consists of **decoding steps** of two types: **mapping steps** and **generation steps**. If more steps contribute to the same output factor, they have to agree on the outcome, i.e. partial hypotheses where two decoding steps produce conflicting values in an output factor are discarded.

A **translation scenario** is a fixed configuration describing which decoding steps to use in which order. Figure 3.13 on page 76 illustrates one possible translation scenario, we examine several options in Section 4.4 below.

Mapping Steps

A **mapping step** from a subset of source factors $S \subseteq \{1 \dots F\}$ to a subset of target factors $T \subseteq \{1 \dots C\}$ is the standard phrase-based model (see e.g. (Koehn, 2004a)) and introduces a feature in the following form:

$$\tilde{h}_m^{\text{map}:S \to T}(\tilde{c}_k, \tilde{f}_k) = \log p(\tilde{f}_k^S | \tilde{c}_k^T) \tag{4.4}$$

The conditional probability of $\tilde{f}_k^S$, i.e. the phrase $\tilde{f}_k$ restricted to factors S, given $\tilde{c}_k^T$, i.e. the phrase $\tilde{c}_k$ restricted to factors T is estimated from relative frequencies: $p(\tilde{f}_k^S | \tilde{c}_k^T) = N(\tilde{f}^S, \tilde{c}^T)/N(\tilde{c}^T)$ where $N(\tilde{f}^S, \tilde{c}^T)$ denotes the number of co-occurrences of a phrase pair $(\tilde{f}^S, \tilde{c}^T)$ that are consistent with the word alignment. The marginal count $N(\tilde{c}^T)$ is the number of occurrences of the target phrase $\tilde{c}^T$ in the training corpus.

For each mapping step, the model is included in the log-linear combination in source-to-target and target-to-source directions: $p(\tilde{f}^T | \tilde{c}^S)$ and $p(\tilde{c}^S | \tilde{f}^T)$. In addition, statistical single word based lexicons are used in both directions. They are included to smooth the relative frequencies used as estimates of the phrase probabilities.

Generation Steps

A **generation step** maps a subset of target factors T_1 to a disjoint subset of target factors T_2, $T_{1,2} \subset \{1 \dots C\}$. In the current implementation of Moses, generation steps are restricted to word-to-word correspondences:

$$\tilde{h}_m^{\text{gen}:T_1 \to T_2}(\tilde{c}_k, \tilde{f}_k) = \log \prod_{i=1}^{\text{length}(\tilde{c}_k)} p(\tilde{c}_{k,i}^{T_1} | \tilde{c}_{k,i}^{T_2}) \tag{4.5}$$

where $\tilde{c}_{k,i}^{T}$ is the i-th words in the k-th target phrase restricted to factors T. We estimate the conditional probability $p(\tilde{c}_{k,i}^{T_2} | \tilde{c}_{k,i}^{T_1})$ by counting over words in the target-side corpus. Again, the conditional probability is included in the log-linear combination in both directions.

4.2.5 Language Models

In addition to features for decoding steps, we include arbitrary number of language models[5] over subsets of target factors, $T \subseteq \{1 \dots C\}$. We currently use standard n-gram language model:

$$h_{\text{LM}_n}^{T}(f_1^J, c_1^I) = \log \prod_{i=1}^{I} p(c_i^T | c_{i-1}^T \dots c_{i-n+1}^T) \tag{4.6}$$

While generation steps are used to enforce "vertical" coherence between "hidden properties" of output words, language models are used to enforce sequential coherence of the output.

4.2.6 Beam-Search

Operationally, Moses performs a stack-based beam search very similar to Pharaoh (Koehn, 2004a). Thanks to the synchronous-phrases assumption, all the decoding steps can be performed during a preparatory phase. For each span in the input sentence, all possible translation options are constructed using the mapping and generation steps in a user-specified order. Low-scoring options are pruned already during this phase. Once all translation options are constructed, Moses picks source phrases (all output factors already filled in) in arbitrary order, subject to a reordering limit and a probabilistic reordering cost, producing the output in left-to-right fashion and scoring it using the specified language models exactly as Pharaoh does.

[5] This might be perceived as a non-standard use of the term, because the models may contain more than just word forms. More generally, these models represent a specific case of a probabilistic sequence model.

4.3 Data Used

The experiments reported in this chapter were carried out with the News Commentary (NC) corpus as made available for the SMT workshop[6] of the ACL 2007 conference.[7]

The Czech part of the corpus was tagged and lemmatized using the tool by Hajič and Hladká (1998), the English part was tagged using MXPOST (Ratnaparkhi, 1996) and lemmatized using the Morpha tool (Minnen *et al.*, 2001). After some final cleanup, the corpus consists of 55,676 pairs of sentences (1.1M Czech tokens and 1.2M English tokens). We use the designated additional tuning and evaluation sections consisting of 1023, resp. 964 sentences.

In all experiments, word alignment was obtained using the grow-diag-final heuristic for symmetrizing GIZA++ (Och and Ney, 2003) alignments. To reduce data sparseness, the English text was lowercased and Czech was lemmatized for alignment estimation, a setup confirmed as very useful in our previous Czech-to-English MT experiments (Bojar *et al.*, 2006).

Language models are based on the target side of the parallel corpus only, unless stated otherwise.

We report BLEU (Section 3.8) scores for systems trained and tested in *case-insensitive* fashion (all data are converted to lowercase, including the reference translations), unless stated otherwise.

4.4 Scenarios of Factored Translation English→Czech

We experimented with the following factored translation scenarios.

The baseline scenario (labelled T for translation) is single-factored: input (English) lowercase word forms are directly translated to target (Czech) lowercase forms. A 3-gram language model (or more models based on various corpora) checks the stream of output word forms. The baseline scenario thus corresponds to a plain phrase-based SMT system:

English		Czech	
lowercase	⟶	lowercase	+LM
lemma		lemma	
morphology		morphology	

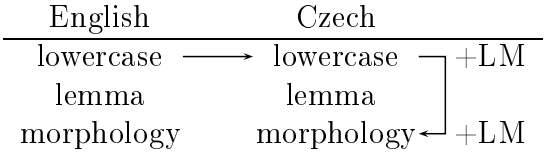
[6] `http://www.statmt.org/wmt07/`

[7] Our preliminary experiments with the Prague Czech-English Dependency Treebank, PCEDT v.1.0 (Čmejrek *et al.*, 2004), 20k sentences, gave similar results, although with a lower level of significance due to a smaller evaluation set.

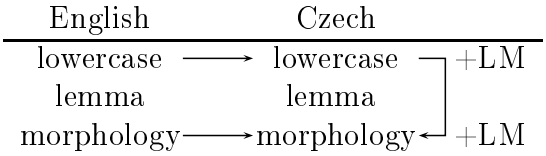
In order to check the output not only for word-level coherence but also for morphological coherence, we add a single generation step: input word forms are first translated to output word forms and each output word form then generates its morphological tag.

Two types of language models can be used simultaneously: a (3-gram) LM over word forms and a (7-gram) LM over morphological tags.

We used tags with various levels of detail, see Section 4.5. We call this the "T+C" (translate and check) scenario:

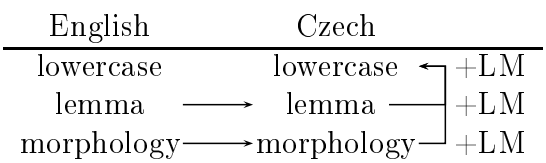


As a refinement of T+C, we also used T+T+C scenario, where the morphological output stream is constructed based on both output word forms and input morphology. This setting should reinforce correct translation of morphological features such as number of source noun phrases. To reduce the risk of early pruning, the generation step operationally precedes the morphology mapping step. Again, two types of language models can be used in this "T+T+C" scenario:



The most complex scenario we used is linguistically appealing: output lemmas (base forms) and morphological tags are generated from input in two independent translation steps and combined in a single generation step to produce output word forms.

The "T+T+G" setting allows us to use three types of language models. Trigram models are used for word forms and lemmas and 7-gram language models are used over tags:



	BLEU
T+T+G	13.9±0.7
T+T+C	13.9±0.6
T+C	13.6±0.6
Baseline: T	12.9±0.6

Table 4.1: BLEU scores of various translation scenarios.

4.4.1 Experimental Results: Improved over T

Table 4.1 summarizes estimated translation quality of the various scenarios. In all cases, a 3-gram LM is used for word forms or lemmas and a 7-gram LM for morphological tags.

The good news is that multi-factored models always outperform the baseline T.

Unfortunately, the more complex multi-factored scenarios do not bring any significant improvement over T+C. Our belief is that this effect is caused by search errors: with multi-factored models, more hypotheses get similar scores and future costs of partial hypotheses might be estimated less reliably. With a limited stack size (not more than 200 hypotheses of the same number of covered input words), the decoder may more often find sub-optimal solutions. Moreover, the more steps are used, the more model weights have to be tuned in the minimum error rate training. Considerably more tuning data might be necessary to tune the weights reliably.

4.5 Granularity of Czech Part-of-Speech Tags

As stated above, the Czech morphological tag system is very complex: in theory up to 4,000 different tags are possible. In our T+T+C scenario, we experiment with various simplifications of the system to find the best balance between richness and robustness of the statistics available in our corpus. (The more information is retained in the tags, the more severe data sparseness is.)

Full tags (1200 unique seen in the 56k corpus): Full Czech positional tags are used. A tag consists of 15 positions, each holding the value of a morphological property (e.g. number, case or gender).[8]

[8] In principle, each of the 15 positions could be used as a separate factor. The set of necessary generation steps to encode relevant dependencies would have to be carefully determined.

POS+case (184 unique seen): We simplify the tag to include only part and subpart of speech (also distinguishes partially e.g. verb tenses). For nouns, pronouns, adjectives and prepositions[9], also the case is included.

CNG01 (621 unique seen): CNG01 refines POS. For nouns, pronouns and adjectives we include not only the case but also number and gender.

CNG02 (791 unique seen): Tag for punctuation is refined: the lemma of the punctuation symbol is taken into account; previous models disregarded e.g. the distributional differences between a comma and a question mark. Case, number and gender added to nouns, pronouns, adjectives, prepositions, but also to verbs and numerals (where applicable).

CNG03 (1017 unique seen): Optimized tagset:

- Tags for nouns, adjectives, pronouns and numerals describe the case, number and gender; the Czech reflexive pronoun *se* or *si* is highlighted by a special flag.
- Tag for verbs describes subpart of speech, number, gender, tense and aspect; the tag includes a special flag if the verb was the auxiliary verb *být (to be)* in any of its forms.
- Tag for prepositions includes the case and also the lemma of the preposition.
- Lemma included for punctuation, particles and interjections.
- Tag for numbers describes the "shape" of the number (all digits are replaced by the digit *5* but number-internal punctuation is kept intact). The tag thus distinguishes between 4- or 5-digit numbers and it indicates the precision of floating point numbers.
- Part of speech and subpart of speech for all other words.

4.5.1 Experimental Results: CNG03 Best

Table 4.2 summarizes the results of T+T+C scenario with varying detail in morphological tag.

Our results confirm improvement over the single-factored baseline. Detailed knowledge of the morphological system also proves its utility: by choosing the most relevant features of tags and lemmas but avoiding sparseness,

[9]Some Czech prepositions select for a particular case, some are ambiguous. Although the case is never shown in the surface form of the preposition, the tagset includes this information and Czech taggers are able to infer the case.

	BLEU
Baseline: T (single-factor)	12.9±0.6
T+T+C, POS+case	13.2±0.6
T+T+C, CNG01	13.4±0.6
T+T+C, CNG02	13.5±0.7
T+T+C, full tags	13.9±0.6
T+T+C, CNG03	14.2±0.7

Table 4.2: BLEU scores of various granularities of morphological tags in T+T+C scenario.

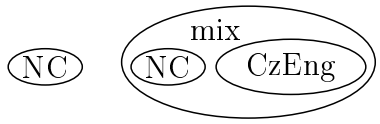
we can improve on BLEU score by about 0.3 absolute over T+T+C with full tags.

4.6 More Out-of-Domain Data in T and T+C Scenarios

In order to check if the method scales up with more parallel data available, we extend our training data using the CzEng parallel corpus (Bojar and Žabokrtský, 2006). CzEng contains sentence-aligned texts from the European Parliament (about 75%), e-books and stories (15%) and open source documentation. By "NC" corpus we denote the in-domain News Commentary corpus only, by "mix" we denote the combination of training sentences from NC and CzEng (1070k sentences, 13.9M Czech and 15.5 English tokens) where in-domain NC data amounts only to 5.2% sentences. The third option, "weighted", is a combination of NC and mix with a scaling factor α optimized in MERT (i.e. NC is included twice).

Table 4.3 gives full details on our experiments with the additional data. We varied the scenario (T or T+C), the level of detail in the T+C scenario (full tags vs. CNG03) and the size of the training corpus. We extract phrases from either the in-domain corpus only (NC) or the mixed corpus (mix). We use either one LM per output factor, varying the corpus size (NC or mix), or two LMs per output factors with weights trained independently in the MERT procedure (weighted). Independent weights allow us to take domain difference into account, but we exploit this in the target LM only, not the phrases.

The only significant difference is caused by the scenario: T+C outperforms the baseline T, regardless of corpus size. Other results (insignificantly) indicate the following observations:



$$\text{weighted} = \alpha\text{NC} + (1-\alpha)\text{mix}$$

Scenario	Phrases from	LMs	BLEU
T	NC	NC	12.9±0.6
T	mix	mix	11.8±0.6
T	mix	weighted	11.8±0.6
T+C CNG03	NC	NC	13.7±0.7
T+C CNG03	mix	mix	13.1±0.7
T+C CNG03	mix	weighted	13.7±0.7
T+C full tags	NC	NC	13.6±0.6
T+C full tags	mix	mix	13.1±0.7
T+C full tags	mix	weighted	13.8±0.7

Table 4.3: The effect of additional data in T and T+C scenarios.

- Ignoring the domain difference and using only the mixed domain LM in general performs worse than allowing MERT to optimize LM weights for in-domain and generic data separately.[10]
- CNG03 outperforms full tags only in small data setting, with large data (treating the domain difference properly), full tags perform better.

4.7 Human Evaluation

The best system described in this chapter (T+C full tags with additional CzEng data) took part in an open MT evaluation campaign carried out during ACL 2007 Second Workshop on Statistical Machine Translation[11]. Table 4.4 reproduces the results from Callison-Burch *et al.* (2007) for English to Czech MT quality. The **adequacy** scale describes how well the translation conveys the original meaning, the **fluency** reflects how grammatically correct the MT output is and **rank** shows how often would human judges prefer to get output from that particular system compared to other systems. The **constituent rank** is a new scale introduced by Callison-Burch *et al.* (2007)

[10] In our previous experiments with PCEDT as the domain-specific data, the difference was more apparent because the corpus domains were more distant. In the T scenario reported here, the weighted LMs did not bring any improvement over "mix" and even performed worse than the baseline NC. We attribute this effect to some randomness in the MERT procedure.

[11] `http://www.statmt.org/wmt07/`

System	Adequacy	Fluency	Rank	Constituent
Our T+C (cu)	0.523	0.510	0.405	**0.440**
PC Translator (pct)	**0.542**	**0.541**	**0.499**	0.381
Single-Factored Moses (uedin)	0.449	0.433	0.249	0.258

Table 4.4: Human judgements of English→Czech MT quality at ACL WMT 2007.

that tries to simplify the task of ranking hypotheses by asking the judges to rank only randomly selected sections of sentences.

Our system improved over the phrase-based baseline (provided by University of Edinburgh, uedin) and got very close to a major English-Czech commercial MT system PC Translator[12] by LangSoft (a rule-based system with a long history of development). Despite the comparison not being completely fair (PC Translator is a generic MT system while our system was trained and evaluated in the known domain of news commentaries), we consider the result very promising.

We participated with a very similar setup also in ACL 2008 WMT shared task[13] (Bojar and Hajič, 2008). The only differences were that (1) we trained our system on the recent release of CzEng 0.7 (Bojar *et al.*, 2008) which is slightly bigger, (2) we used "true-cased" data (preserve capitalization of names but drop capitalization of sentence beginnings), and most importantly (3) we included the Czech National Corpus SYN2006 (365M tokens) in a 4-gram language model over word forms and 7-gram language model over morphological tags. As documented in Table 4.5 (results from Callison-Burch *et al.* (2008)), the additional data allowed us to improve over PC Translator for in-domain setting (Commentary). In the generic domain of News, PC Translator performs better.

A somewhat surprising result of WMT08 evaluation of English-to-Czech translation is that while the systems fall into two rather distinct groups of performance, it is always a statistical and a rule-based system that form a group (our T+C and PC-Translator vs. TectoMT (Žabokrtský, 2008b) and single-factored Moses). We see that even very complementary strategies can lead to comparable MT quality, which suggest that the potential gain from systems combination may be quite high.

Examples of output of various MT systems including the recently launched Google Translate are available in Appendix A. Apart from indicating the overall state-of-the-art quality of MT, the examples also illustrate how difficult it is to compare MT systems, both manually or automatically.

[12] `http://www.translator.cz/`

[13] `http://www.statmt.org/wmt08/`

System	Commentary (in-domain)	News (out-of-domain)
Our T+C (cu-bojar)	**71.4%**	63.4%
PC Translator	66.3%	**71.5%**
TectoMT (cu-tectomt)	48.8%	49.4%
Single-Factored Moses (uedin)	48.6%	50.2%

Table 4.5: Percentage of sentences where the system was ranked better than or equal to any other system (human judgements, ACL WMT08).

Translation of	**Verb**	**Modifier**
... preserves meaning	56%	79%
... is disrupted	14%	12%
... is missing	27%	1%
... is unknown (not translated)	0%	5%

Table 4.6: Analysis of 77 verb-modifier pairs in 15 sample sentences.

4.8 Untreated Morphological Errors

The previous sections described improvements gained on small data sets when checking morphological agreement using T+T+C scenario (BLEU raised from 12.9% to 13.9% or up to 14.2% with manually tuned tagset, CNG03). However, the best result achieved is still far below the margin of lemmatized BLEU (21%), as mentioned in Section 4.1.1.

When we searched for the unexploited morphological errors, visual inspection of MT output suggested that local agreement (within 3-word span) is relatively correct but verb-modifier relations are often malformed causing e.g. a bad case for the modifier. To quantify this observation we performed a micro-study of our best MT output using an intuitive metric. We checked whether verb-modifier relations are properly preserved during the translation of 15 sample sentences.

The *source* text of the sample sentences contained 77 verb-modifier pairs. Table 4.6 lists our observations on the two members in each verb-modifier pair. We see that only 56% of verbs are translated correctly and 79% of nouns are translated correctly. The system tends to skip verbs quite often (27% of cases).

More importantly, our analysis has shown that even in the cases where both the verb and the modifier are lexically correct, the relation between them in Czech is either non-grammatical or meaning-disrupted in 56% of

Input:	Keep on investing.
MT output:	Pokračovalo investování. (grammar correct here!)
Gloss:	Continued investing. (Meaning: The investing continued.)
Correct:	Pokračujte v investování.

Input:	brokerage firms rushed out ads ...			
MT Output:	brokerské	firmy	vyběhl	reklamy
Gloss:	brokerage	$\text{firms}_{pl.fem}$	$\text{ran}_{sg.masc}$	$\text{ads}^{pl.voc,sg.gen}_{pl.nom,pl.acc}$
Correct:	brokerské	firmy	vychrlily	$\text{reklamy}_{pl.acc}$
Comprehensible:	brokerské	firmy	vyběhly	s $\text{reklamami}_{pl.instr}$

Figure 4.2: Two sample errors in translating verb-modifier relations from English to Czech.

these cases. Commented samples of such errors are given in Figure 4.2 below. The first sample shows that a strong language model can lead to the choice of a grammatical relation that nevertheless does not convey the original meaning. The second sample illustrates a situation where the system failed to choose an acceptable form for the relation between *rush out* and *ads* most probably because it backed off to a generic pattern verb-noun^{accusative}. This pattern is quite common for expressing the object role of many verbs (such as *vychrlit*, see the Correct option in Figure 4.2), but does not fit well with the verb *vyběhnout*. If the dictionary forced the system to use *vyběhnout*, a different preposition and case should have been chosen to render the output at least comprehensible (the lexical choice is still problematic, the best equivalent would probably be *vyrazily s reklamami*). While the target-side data may be rich enough to learn the generalization vyběhnout–s–*instr*, no such generalization is possible with language models over word forms or morphological tags only. The target side data will be hardly ever rich enough to learn this particular structure in all correct morphological and lexical variants: *vyběhl–s–reklamou, vyběhla–s–reklamami, vyběhl–s–prohlášením, vyběhli–s–oznámením, ...*. We would need a mixed model that combines verb lemmas, prepositions and case information to properly capture the relations.

Unfortunately, our preliminary experiments that made use of automatic Czech analytical trees to construct a factor explicitly highlighting the verb (lexicalized) its modifiers (case and the lemma of the preposition, if present) and boundary symbols such as punctuation or conjunctions and using a dummy token for all other words did not bring any improvement over the baseline. A possible reason is that we employed only a standard 7-gram language model to this factor. A more appropriate treatment is to disregard

the dummy tokens in the language model at all and use a "skipping" n-gram language model that looks at last $n-1$ non-dummy items.

4.9 Related Research

Class-based LMs (Brown *et al.*, 1992) or factored LMs (Bilmes and Kirchhoff, 2003) are very similar to our T+C scenario. Given the small differences in all T+...scenarios' performance, class-based LM might bring equivalent improvement. Yang and Kirchhoff (2006) have recently documented minor BLEU improvement using factored LMs in single-factored SMT to English. The multi-factored approach to SMT of Moses is however more general.

Many researchers have tried to employ morphology in improving word alignment techniques (e.g. (Popović and Ney, 2004)) or machine translation quality (Nießen and Ney (2001), Koehn and Knight (2003), Zollmann *et al.* (2006), among others, for various languages; Goldwater and McClosky (2005), Bojar *et al.* (2006) and Talbot and Osborne (2006) for Czech), however, they focus on translating *from* the highly inflectional language.

Durgar El-Kahlout and Oflazer (2006) report preliminary experiments in English to Turkish single-factored phrase-based translation, gaining significant improvements by splitting root words and their morphemes into a sequence of tokens. It might be interesting to explore multi-factored scenarios for different Turkish morphology representation suggested in the paper.

De Gispert *et al.* (2005) generalize over verb forms and generate phrase translations even for unseen target verb forms. The T+T+G scenario allows a similar extension if the described generation step is replaced by a (probabilistic) morphological generator.

Nguyen and Shimazu (2006) translate from English to Vietnamese but the morphological richness of Vietnamese is comparable to English. In fact the Vietnamese vocabulary size is even smaller than English vocabulary size in one of their corpora. The observed improvement due to explicit modelling of morphology might not scale up beyond small-data setting.

As an alternative option to our verb-modifier experiments, structured language models (Chelba and Jelinek, 1998) might be considered to improve clause coherence. Birch *et al.* (2007) reports improvements in sentence coherence using factored translation with CCG supertags. For languages with significant but predictable syntactic divergence such as German-to-English translation, automatic preprocessing of the word order significantly increases MT quality (Collins *et al.*, 2005). Cuřín (2006) reports improvement for Czech-to-English translation using a similar preprocessing technique focused on introducing required English auxiliary words. And surely, another op-

tion to improve output grammaticality is to employ full-featured syntax-based MT models (Wu and Wong (1998), Yamada and Knight (2002), Eisner (2003), Chiang (2005), Quirk and Menezes (2006) and our own experiments in Chapter 3 among many others).

4.10 Conclusion

Moving away from basic research of lexical acquisition (Chapter 2) and a linguistically justified but complex system of syntax-based machine translation (Chapter 3) to a goal-oriented direct method, this chapter introduced so-called phrase-based translation, currently best performing MT technique for many language pairs.

We summarized the extension of phrase-based systems to multi-factored MT and experimented with various setups of additional factors (translation scenarios), the level of detail in morphological tags and additional training data.

Our results on English-to-Czech translation demonstrate significant improvement in BLEU scores by explicit modelling of morphology and using a separate morphological language model to ensure the coherence. To our knowledge, the original experiments as described in (Bojar, 2007) were among the first to show the advantages of using multiple factors in MT. With some additional data, we were able to improve over a commercial MT system in a known domain in 2008.

Errors in expressing verb-modifier relations have been studied and a factor capturing these dependencies has been proposed. Unfortunately, this factor has yet to bring any improvement.

Chapter 5

Concluding Discussion

The underlying topic of the thesis is the relation between linguistic data and applications. We focused on creating a deep syntactic lexicon and on two methods of machine translation: a deep syntax-based MT and a shallow phrase-based MT.

To provide a larger picture, we survey available literature with a simple question in mind: Do lexicons bring an improvement to NLP applications? Not surprisingly, there is not a simple and conclusive answer to this question. Hopefully, we managed to keep a balanced view and to mediate some interesting lessons to learn from the past projects.

5.1 When Lexicons Proved to Be Useful

Litkowski (2005) gives a good overview of the current state in computational lexicography including illustrations of NLP tasks and explanations of how lexicons can be employed in them. Litkowski's main belief in lexicon utility comes from the "semantic imperative": "In considering the NLP applications of word-sense disambiguation, information extraction, question answering, and summarization, there is a clear need for increasing amounts of semantic information. The main problem facing these applications is a need to identify paraphrases, that is, identifying whether a complex string of words carries more or less the same meaning as another string." Later, he notes: "As yet, the symbolic content of traditional dictionaries has not been merged with the statistical properties of word usage revealed by corpus-based methods."

Of the many dictionary-like resources available, there seems to be only one that has been applied to a wide range of applications more or less successfully: WordNet (Fellbaum, 1998).

In some situations, lexicons are used to improve coverage (recall). For instance, WordNet can be used as a back-off to replace words not known to the system with a suitable synonym or hyperonym. In some situations, lexicons might improve the precision, such as a morphological lexicon in speech

recognition (morphological lexicon is generally more accurate than rules describing valid word forms). A lexicon can be also used as an authoritative source of terms, expressions of constructions (e.g. EuroVoc[1]). The system can then guarantee a certain level of output quality.

5.1.1 Lexicon Improves Information Retrieval

In an information retrieval system described by Woods *et al.* (1999), the addition of a morphological dictionary, taxonomic information between concepts (WordNet-like) and rules describing general entailment between words and concepts improved significantly the performance. An additional improvement was achieved by employing a morphological guesser to analyse words not listed in the lexicon. As a matter of fact, both the taxonomic (semantic) and the morphological guesser were used in an over-generation fashion: the input query was relaxed using the lexicons. All the documents that match the relaxed queries are then sorted so that documents with a closer match (less relaxation) appear on top. The lexical information is thus used to improve recall only, while the sufficient precision is ensured at no additional cost by input data.

Similar techniques are used for morphologically rich languages in search engines. An old example for Czech dates back to the ASIMUT system (Králíková and Panevová, 1990).

5.1.2 Subcategorization Improves Parsing

Subcategorization information can serve as an example where the lexicon improves the precision of the system. A parse (i.e. a syntactic analysis of a sentence) is suppressed, if the pattern of a word's modifications is not approved by a subcategorization lexicon.

As documented in (Carroll *et al.*, 1998) and cited papers, including statistics on the co-occurrence of lexical heads of phrases and the configurations of members in the phrase (i.e. complements and adjuncts) brings substantial improvements in parsing accuracy. Zeman (2002) also reports a significant improvement in parsing accuracy of his dependency-based statistical parser when subcategorization information was added. However, the absolute level of his parser's accuracy remains below modern versions of phrase-based parsers that include head-lexicalized statistics such as Collins *et al.* (1999).

More importantly, we are not aware of any published result demonstrating that subcategorization *lexicons* (built manually or automatically) would be

[1] `http://europa.eu/eurovoc/`

used in top-performing parsers.[2]

The claim we want to make is that while subcategorization information is important and it indeed helps parsing, it can be extracted automatically and most probably in a simple form tailored for the task and thus more suitable than lexicons prepared independently. In some settings though, the lexicons might provide a bigger coverage than what can be observed in the training data.

5.1.3 Lexicons Employed in MT

Liu *et al.* (2005) describe a log-linear model for word alignment where a bilingual lexicon can be added as a feature. A hand-made lexicon of word-to-word translation equivalents contributed slightly to the overall good performance of the system. The structure of the lexicon is very simple and also the evaluation is measured in terms of alignment error rate (AER) against alignments annotated by humans. It is not clear, if we would observe an improvement in an end-to-end evaluation of an MT system. (AER is known not to directly correlate with MT quality measures (Lopez and Resnik, 2006))

Fujita and Bond (2002) describe a method of augmenting a translation dictionary with subcategorization information available for similar words (other possible translation equivalents) already listed in the dictionary. The utility was evaluated on the ALT-J/E rule-based MT system (Ikehara *et al.*, 1991): based on a human judgement by a *single* native speaker, the translation quality of only about 100 evaluation sentences improved in 31% of cases and degraded in 8% of cases. Fujita and Bond (2004) report a similar experiment where available verb alternation data was used to add the missing half of the translation lexicon entry of an alternating verb. The method requires a list of verbs participating in a specific alternation, the description of the alternation in terms of valency slot changes, including changes in syntactic structure and selectional restrictions, and a seed bilingual translation dictionary. No completely new verbs are added to the dictionary, but the existing entries are augmented with the missing halves of the alternation. Evaluated by two native speakers on 124 test sentences, the augmented lexicon leads to a better translation in about 46% of sentences and to a worse translation in about 15% of cases. However, the ALT-J/E system has probably never been evaluated on a standard test set so it is difficult to assess its real usability.

Boguslavsky *et al.* (2004) describe a range of dictionaries used in ETAP-3[3]

[2] An exception is the employment of VerbaLex lexicon in a Czech parser. Hlaváčková *et al.* (2006) demonstrate a dramatic reduction in parse ambiguity thanks to VerbaLex entries. However, they do not evaluate the actual parsing accuracy.

[3] `http://cl.iitp.ru/`

(Apresjan *et al.*, 2003). Unfortunately, the MT system has probably neither been evaluated on a standard test set nor has taken part in an evaluation competition, but the authors claim and the web demo suggests that the coverage of the system is sufficiently large. Based on the Meaning-Text-Theory (Mel'čuk, 1988) and implemented as hand written rules, the system heavily depends on the quality of encoded lexicons. The applicability of ETAP-3 therefore confirms the utility of its lexicons.

5.1.4 Lexicons Help Theories

A lexicon is also an indispensable tool in refining linguistic theories. As explained above, a lexicon serves as a mapping between units on (typically) two levels of language description. Given a multi-layer linguistic theory that formally defines units at the various levels, a lexicon can prove or disprove the appropriateness of the theory. If the lexicographic work proceeds smoothly and large data is covered with lexical entries, then the theory was all right. If problems are noticed, the theory can be adjusted accordingly as e.g. in Lopatková and Panevová (2005).

5.2 When Lexicons Were Not Needed

This section surveys some practical NLP tasks that are often used to motivate the creation of lexicons. As we will see, depending on the specifics of the task and method chosen, surprisingly good results can be often achieved without any such lexicon.

5.2.1 PP Attachment without Lexicons

Calvo *et al.* (2005) conducts, to the best of our knowledge, the only experiment directly evaluating the utility of a hand-written lexicon (WordNet in particular) against a lexicon derived automatically from corpus data to solve a common task: attachment of prepositional phrases (PP).

The authors describe a method of automatically building a thesaurus and using the thesaurus as a back-off for the PP attachment problem. A comparison with a similar method based on manual (WordNet) data indicates that the results based on a manual and automatic resources are nearly identical. Higher precision scores of PP attachment are achieved without any back-off, but the coverage is very poor.

However, the task of PP attachment is notoriously hard and given the relatively low performance of both the dictionary-based and the automatic method, we cannot confidently claim superiority of any of the methods.

5.2.2 MT without Lexicons

For the time being, top performing MT systems include statistical phrase-based methods (Callison-Burch *et al.*, 2007) and in some evaluations the phrase-based systems win by a large margin.[4] These systems do not rely on any translation dictionaries but rather build them automatically, given a collection of word-aligned parallel texts. The "structure" of such lexicons is typically very simple, they contain just pairs of (sequences of) word forms in the source and target languages with no additional linguistic information, except for a co-occurrence count/probability.

Stevenson (2003) reviews the hopes of word-sense disambiguation (WSD) usefulness in various NLP tasks including MT. It seems that only very recent experiments follow Stevenson's wish: "the only way in which it can be accurately determined whether these systems [e.g. MT] will benefit from the information produced by some [WSD] component is to integrate it as part of the final system and record the change in performance." Experiments to date provide mixed results: Carpuat and Wu (2005) describe several techniques of a loose combination of a WSD and an MT system that fail to bring any significant improvement. While this particular experiment has some peculiarities[5], the same doubt on WSD utility came up in Senseval-3 panel discussions[6] in 2004. It is also worth mentioning that already Senseval-2 included "system evaluation" as one of its subgoals[7] but it does not seem that much success with WSD application has been reported in subsequent Senseval competitions.

Only recently Carpuat and Wu (2007) achieved consistent improvements by coupling the MT system with a WSD method rather tightly. One of the interesting differences between the failing and the succeeding experiments is that the latter do not rely on human-constructed lexicons of senses but rather use phrase tables extracted automatically from a parallel corpus. We can thus say that while WSD techniques can bring an improvement in MT quality, this was not yet demonstrated using human-annotated lexical data.

One of the motivations for building valency lexicons and one of the main

[4] NIST 2005 official evaluation, `http://www.nist.gov/speech/tests/`.

[5] The WSD task is used for 20 words only with 2 to 8 senses per word and there is only 37 occurrences of the words in the training data. Also, the WSD module is not used a feature in the SMT system, but rather employed in two hard ways: either in post-processing by replacing the output word with the translation equivalent suggested by WSD (this can break the cohesion of the sentence), or to prune all paths in the lattice that do not contain the target word. A finer combination of WSD and SMT would allow to tune a weight assigned to the WSD module.

[6] `http://www.senseval.org/senseval3/panels`

[7] `http://www.itri.brighton.ac.uk/events/senseval/SENSEVAL2/task-design.ps`

reasons for introducing syntax-based models to MT is the aim to produce correct valency structures of verbs and other elements in the sentence. If a word is not accompanied by all grammatically required modifiers or if there are unexpected additional modifiers, the sentence feels disfluent. Dependency grammars equipped with a valency dictionary such as we have seen in Chapter 2 should be able to identify the problem and prefer a different translation. STSG models valency explicitly, treelet pairs can be seen as bilingual valency frames.

In real world sentences though, dependency edges are relatively short (Holan, 2003) and thus can be approximated reasonably well by plain adjacency of sentence elements (words). The phrase-based approach described in Chapter 4 can thus in many cases capture and translate valency frames correctly, provided the phrase-length limit is large enough. The only real advantage of syntax-based methods is a better ability to generalize, e.g. abstract away all adjectives intervening between a verb and its object. It would be interesting to evaluate how often does such a generalization capacity promise to bring an improvement in a real MT task with fixed training and test data.

Finally, Och (2005) demonstrates that (according to current evaluation metrics) the key features of MT systems that lead to success are: (1) simplicity, such as a combination of independent features, relatively simple from the linguistic point of view, (2) minimality of design and representation, such as stemming of words, or only a few bits to represent probabilities, and (3) vast amounts of textual data. These features are somewhat contradictory to what we obtain from elaborated lexicons.

5.2.3 Question Answering without Deep Syntax

Mooney (2000) describes a system CHILL that converts questions in a natural language into Prolog queries. The answers are obtained by evaluating the query on a database. The system performs very well on restricted domains (geographical knowledge about the U.S., a thousand of restaurants in northern California or job opportunities). In the system, the deep syntactic level is simply skipped. To start working on a new domain, only a set of (a few hundreds of) sample questions and expected Prolog queries are needed as the training data. CHILL learns a shift-reduce parser for input sentences to produces directly the Prolog query. In Wong and Mooney (2007), the direct translation from plain text sentence to the Prolog query is casted as synchronous context-free grammar derivation, skipping any syntactic layer again.

As Litkowski (2005) summarizes: "From the beginning, researchers viewed this NLP task [Question Answering] as one that would involve semantic pro-

cessing and provide a vehicle for deeper study of meaning and its representation. This has not generally proved to be the case, but many nuances have emerged in handling different types of questions."

5.2.4 Summarization without Meaning and Grammaticality without Valency Lexicon

Barzilay and McKeown (2005) describe a sentence fusion technique employed in summarization of multiple source documents, the Newsblaster[8] (McKeown *et al.*, 2002). Only shallow syntactic analysis of the input text (dependency parsing) and generic knowledge collected from a larger text corpus are needed. Output sentences are generated by reusing and altering phrases from several source sentences. More specifically, a centroid sentence (an input sentence most similar to other input sentences) is selected and its dependency tree is gradually altered by adding information present only in (a majority of) other sentences and by removing information not supported by a reasonable share of other sentences. Grammaticality is ensured by keeping all modifications very conservative: information is added, only if the root node of the added subtree can be aligned to a node already present in the centroid sentence, and nodes are deleted only in a small pre-defined set of cases (such as removing components from conjunctions or removing adverbs). The lack of an explicit valency lexicon is thus compensated by making use of "valency exhibited" in input sentences.

Barzilay and McKeown (2005) also mention problems with the linearization of the output dependency structure using a large-scale unification-based text generation system FUF/SURGE. FUF/SURGE requires edges in the input dependency trees to be labelled with syntactico-semantic roles such as "manner" or "location". If the roles are added automatically (and there is no other option for machine-generated input trees), errors lead to completely scrambled output, wrong prepositions etc. Barzilay and McKeown (2005) achieve better results with a statistical linearization component, which is not only more robust to errors but also more efficient, because it can make use of phrases readily available in the data. The FUF/SURGE generation system produces every phrase from scratch. Due to limitations inherent to current n-gram based language modelling techniques, suboptimal linearizations are sometimes chosen. Once language modelling techniques are improved with respect to syntactic properties of the language, more grammatical output will be produced. (As always, language-specific issues have to be taken into account when drawing conclusions from other observations. If the tar-

[8] `http://www.cs.columbia.edu/nlp/newsblaster/`

get language were a morphologically rich language such as Czech, the language model employed in the statistical linearizer would perform significantly worse.)

5.3 Discussion

Is there a common property of the above mentioned applications that were successful without performing too deep analysis or needing advanced lexicons? In our opinion, the most important common feature of the methods is that the intelligence is left to the human.

- Grammaticality is ensured by reusing a text produced by humans (sentence fusion).
- Selection of the translation equivalent is based on the choice of a human in a similar context (MT).
- Overgeneration never hurts, if the output of the system is intersected with some man-made data (information extraction).

Why are independently designed (manual or automatic) lexicons relatively rarely used in applications? Our guess is the difficulty of adapting the formats and more importantly the difference in types of decisions an application has to make and hints a lexicon can offer.

On the other hand, we have mentioned several applications that build their own lexicons (or probabilistic tables), the features of which are very much influenced by linguistic insights incorporated in human lexicons.

Our belief is that linguistic theories provide an indispensable source of inspiration that is being slowly reflected in the design of applications. Any data produced by computational *linguists* remain difficult to reuse in practical NLP systems because they provide answers for questions the system is nowhere near to ask.

5.4 Contribution of the Thesis

The first part of the thesis (Chapter 2) examined automatic ways of constructing a valency dictionary, an important resource for various applications including rule-based or syntax-based MT. Several methods of frame extraction were designed and evaluated using a novel metric that gives a partial credit even for not quite complete frames by estimating the savings in a lexicographer's work.

The second part (Chapters 3 and 4) focused directly on linguistic data within the task of MT. First, we designed, implemented and evaluated a full-fledged syntax-based MT system. The generic engine was applied in various settings ranging from transfer at a deep syntactic layer to an approximation of an uninformed phrase-based translation. The results indicate that the best translation quality is still achieved by the most simple methods; the main reasons for this being the cumulation of errors, the loss in training data due to both natural and random syntactic divergence between Czech and English and finally a combinatorial explosion in the complex search space.

In Chapter 4 we moved to a relatively simple model of phrase-based MT and we improved its accuracy by adding a limited amount of linguistic information. While word lemmas and morphological tags can be successfully exploited by the phrase-based model thanks to their direct correspondence to the sequence of words achieving a better morphological coherence of MT output, the applicability of syntactic information remains an open research question.

The thesis contributes to the art of natural language processing and machine translation in particular by designing and evaluating:

- an automatic metric estimating the savings in a lexicographer's work;
- experiments with various methods for automatic deep valency frame acquisition based on corpus observations;
- a machine translation system with a deep syntactic transfer, including the evaluation of an end-to-end pipeline; the system can be applied also at a surface-syntactic layer;
- improved word-alignment techniques by preprocessing parallel texts, utilized in experiments reported here and fully described in Bojar *et al.* (2006);
- various configurations of factored phrase-based models for English-to-Czech translation improving target-side morphological coherence.

Moreover, we prepared and made the following data available for the research community:

- a Czech-English parallel corpus CzEng, two public releases (Bojar and Žabokrtský, 2006; Bojar *et al.*, 2008),
- manual Czech-English word-alignment data (Bojar and Prokopová, 2006), including an evaluation of inter-annotator agreement,

- Golden VALEVAL, word-sense disambiguation data from the VALEVAL experiment (Bojar *et al.*, 2005),
- a mildly cleaned-up collection of Czech-English translation dictionaries (Bojar and Prokopová, 2007).

As it tends to happen, a thesis sometimes opens more questions than it actually solves. Many suggestions on how to further improve or extend our methods were mentioned throughout the thesis. We plan to continue our research by further attempts to combine successful simple models with linguistically-informed methods.

Bibliography

Alfred V. Aho and Stephen C. Johnson. Optimal code generation for expression trees. *J. ACM*, 23(3):488–501, 1976. Cited on page 68.

ALPAC. Language and Machines: Computers in Translation and Linguistics. Technical report, Automatic Language Processing Advisory Committee, 1966. Cited on page 53.

Jurij Apresjan, Igor Boguslavsky, Leonid Iomdin, Alexander Lazursky, Vladimir Sannikov, Victor Sizov, and Leonid Tsinman. ETAP-3 Linguistic Processor: a Full-Fledged NLP Implementation of the MTT. In *MTT 2003. First International Conference on Meaning-Text Theory*, pages 279–288, Paris, June 2003. Ecole Normale Superieure. Cited on page 104.

B. T. Sue Atkins. Theoretical Lexicography and its Relation to Dictionary-making. In W. Frawley, editor, *Dictionaries: the Journal of the Dictionary Society of North America*, pages 4–43, Cleveland, Ohio, 1993. DSNA. Cited on page 33.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The Berkeley FrameNet Project. In Christian Boitet and Pete Whitelock, editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 86–90, San Francisco, California, 1998. Morgan Kaufmann Publishers. Cited on page 24, 42.

Regina Barzilay and Kathleen R. McKeown. Sentence Fusion for Multidocument News Summarization. *Computational Linguistics*, 31(3):297–328, September 2005. Cited on page 107.

Sugato Basu. *Semi-supervised Clustering: Probabilistic Models, Algorithms and Experiments*. PhD thesis, Department of Computer Sciences, University of Texas, Austin, Texas, May 2005. Cited on page 50.

Václava Benešová and Ondřej Bojar. Czech Verbs of Communication and the Extraction of their Frames. In *Text, Speech and Dialogue: 9th International Conference, TSD 2006*, volume LNAI 3658, pages 29–36. Springer Verlag, September 2006. Cited on page 30, 41, 46.

Jeff A. Bilmes and Katrin Kirchhoff. Factored language models and generalized parallel backoff. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 4–6, Morristown, NJ, USA, 2003. Association for Computational Linguistics. Cited on page 99.

Alexandra Birch, Miles Osborne, and Philipp Koehn. CCG Supertags in Factored Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 9–16, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 99.

Igor Boguslavsky, Leonid Iomdin, and Victor Sizov. Multilinguality in ETAP-3: Reuse of Lexical Resources. In Gilles Sérasset, editor, *COLING 2004 Multilingual Linguistic Resources*, pages 1–8, Geneva, Switzerland, August 28 2004. COLING. Cited on page 51, 103.

Ondřej Bojar and Jan Hajič. Extracting Translation Verb Frames. In Walther von Hahn, John Hutchins, and Christina Vertan, editors, *Proceedings of Modern Approaches in Translation Technologies, workshop in conjunction with Recent Advances in Natural Language Processing (RANLP 2005)*, pages 2–6. Bulgarian Academy of Sciencies, September 2005. Cited on page 51.

Ondřej Bojar and Jan Hajič. Phrase-Based and Deep Syntactic English-to-Czech Statistical Machine Translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, Ohio, June 2008. Association for Computational Linguistics. in print. Cited on page 96.

Ondřej Bojar and Magdalena Prokopová. Czech-English Word Alignment. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1236–1239. ELRA, May 2006. Cited on page 109.

Ondřej Bojar and Magdalena Prokopová. Czech-English Machine Translation Dictionary. Technical report, ÚFAL MFF UK, Prague, Czech Republic, April 2007. Cited on page 110.

Ondřej Bojar and Zdeněk Žabokrtský. CzEng: Czech-English Parallel Corpus, Release version 0.5. *Prague Bulletin of Mathematical Linguistics*, 86:59–62, 2006. Cited on page 78, 94, 109.

Ondřej Bojar, Jiří Semecký, and Václava Benešová. VALEVAL: Testing VALLEX Consistency and Experimenting with Word-Frame Disambiguation. *Prague Bulletin of Mathematical Linguistics*, 83:5–17, 2005. Cited on page 22, 31, 110.

Ondřej Bojar, Evgeny Matusov, and Hermann Ney. Czech-English Phrase-Based Machine Translation. In *FinTAL 2006*, volume LNAI 4139, pages 214–224, Turku, Finland, August 2006. Springer. Cited on page 77, 90, 99, 109.

Ondřej Bojar, Miroslav Janíček, Zdeněk Žabokrtský, Pavel Češka, and Peter Beňa. CzEng 0.7: Parallel Corpus with Community-Supplied Translations. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May 2008. ELRA. Cited on page 59, 96, 109.

Ondřej Bojar. Towards Automatic Extraction of Verb Frames. *Prague Bulletin of Mathematical Linguistics*, 79–80:101–120, 2003. Cited on page 49, 57.

Ondřej Bojar. Problems of Inducing Large Coverage Constraint-Based Dependency Grammar for Czech. In *Constraint Solving and Language Processing, CSLP 2004*, volume LNAI 3438, pages 90–103, Roskilde University, September 2004. Springer. Cited on page 73.

Ondřej Bojar. English-to-Czech Factored Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 232–239, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 100.

Francis Bond and Sanae Fujita. Evaluation of a Method of Creating New Valency Entries. In *Proc. of Machine Translation Summit IX (MT Summit-2003)*, pages 16–23, New Orleans, Louisiana, September 2003. Cited on page 47, 48, 50.

Francis Bond, Stephan Oepen, Melanie Siegel, Ann Copestake, and Dan Flickinger. Open source machine translation with DELPH-IN. In *Proceedings of the Open-Source Machine Translation Workshop at the 10th Machine Translation Summit*, pages 15–22, Phuket, Thailand, September 2005. Cited on page 56.

Thorsten Brants. TnT - A Statistical Part-of-Speech Tagger . In *ANLP-NAACL 2000*, pages 224–231, Seattle, 2000. Cited on page 60.

Peter F. Brown, J. Cocke, S. A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, Robert L. Mercer, and P. S. Roossin. A Statistical Approach to French/English Translation. In *Proc. of the Second International Conference on Theoretical and Methodolog-*

ical Issues is Machine Translation of Natural Languages; Panel 2: Paradigms for MT, Pittsburg, PA, 1988. Carnegie Mellon University. Cited on page 55.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai, and Robert L. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992. Cited on page 99.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. (Meta-) Evaluation of Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 136–158, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 77, 95, 105.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. Further Meta-Evaluation of Machine Translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, Ohio, June 2008. Association for Computational Linguistics. Cited on page 96.

Hiram Calvo, Alexander Gelbukh, and Adam Kilgarriff. Automatic Thesaurus vs. WordNet: A Comparison of Backoff Techniques for Unsupervised PP Attachment. In *CICLING, 5th Int. Conf. on Intelligent Text Processing and Computational Linguistics*, Mexico City, 2005. Springer Verlag. Cited on page 104.

Nicoletta Calzolari, Francesca Bertagna, Alessandro Lenci, Monica Monachini, et al. Standards and Best Practice for Multilingual Computational Lexicons & MILE (the Multilingual ISLE Lexical Entry), 2001. Available at `http://www.w3.org/2001/sw/BestPractices/WNET/ISLE_D2.2-D3.2.pdf`. Cited on page 33.

Jean Carletta. Assessing agreement on classification task: The kappa statistics. *Computational Linguistics*, 22(2):249–254, 1996. Cited on page 22.

Marine Carpuat and Dekai Wu. Word sense disambiguation vs. statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 387–394, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. Cited on page 105.

Marine Carpuat and Dekai Wu. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, 2007. Cited on page 105.

John Carroll, Guido Minnen, and Ted Briscoe. Can Subcategorisation Probabilities Help a Statistical Parser. In *Proceedings of the 6th ACL/SIGDAT Workshop on Very Large Corpora (WVLC-6)*, Montreal, Canada, 1998. Cited on page 102.

Eugene Charniak. Immediate-Head Parsing for Language Models. In *Meeting of the Association for Computational Linguistics*, pages 116–123, 2001. Cited on page 67.

Ciprian Chelba and Frederick Jelinek. Exploiting Syntactic Structure for Language Modeling. In Christian Boitet and Pete Whitelock, editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 225–231, San Francisco, California, 1998. Morgan Kaufmann Publishers. Cited on page 99.

Monique Chevalier, Jules Dansereau, and Guy Poulin. TAUM-METEO: description du système. Groupe TAUM, Université de Montréal. Montréal, Canada, 1978. Cited on page 53.

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'05)*, pages 263–270, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. Cited on page 100.

Silvie Cinková. From PropBank to EngValLex: Adapting PropBank-Lexicon to the Valency Theory of Functional Generative Description. In *Proceedings of LREC 2006*, pages 2170–2175, 2006. Cited on page 24.

Martin Čmejrek, Jan Cuřín, and Jiří Havelka. Czech-English Dependency-based Machine Translation. In *EACL 2003 Proceedings of the Conference*, pages 83–90. Association for Computational Linguistics, April 2003. Cited on page 60, 74, 77, 81.

Martin Čmejrek, Jan Cuřín, Jiří Havelka, Jan Hajič, and Vladislav Kuboň. Prague Czech-English Dependecy Treebank: Syntactically Annotated Resources for Machine Translation. In *Proceedings of LREC 2004*, Lisbon, May 26–28 2004. Cited on page 59, 90.

Martin Čmejrek. *Using Dependency Tree Structure for Czech-English Machine Translation.* PhD thesis, ÚFAL, MFF UK, Prague, Czech Republic, 2006. Cited on page 59, 61, 62, 69, 70, 71, 78, 80.

Trevor Cohn and Mirella Lapata. Large margin synchronous generation and its application to sentence compression. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 73–82, 2007. Cited on page 60.

Michael Collins and Brian Roark. Incremental parsing with the perceptron algorithm. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 111, Morristown, NJ, USA, 2004. Association for Computational Linguistics. Cited on page 35.

Michael Collins, Jan Hajič, Eric Brill, Lance Ramshaw, and Christoph Tillmann. A Statistical Parser of Czech. In *Proceedings of 37th ACL Conference*, pages 505–512, University of Maryland, College Park, USA, 1999. Cited on page 102.

Michael Collins, Philipp Koehn, and Ivona Kučerová. Clause restructuring for statistical machine translation. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 531–540, Ann Arbor, Michigan, 2005. Association for Computational Linguistics. Cited on page 99.

Michael Collins. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 184–191, 1996. Cited on page 60, 81.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, 1995. Cited on page 38.

Jan Cuřín. *Statistical Methods in Czech-English Machine Translation.* PhD thesis, ÚFAL, MFF UK, Prague, Czech Republic, 2006. Cited on page 99.

Adrià de Gispert, José B. Mariño, and Josep M. Crego. Improving statistical machine translation by classifying and generalizing inflected verb forms. In *Eurospeech 2005*, pages 3185–3188, Lisbon, Portugal, September 2005. Cited on page 99.

Ralph Debusmann and Marco Kuhlmann. Dependency grammar: Classification and exploration, 2007. Project report (CHORUS, SFB 378). Cited on page 57.

George Doddington. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In *Proc. of HLT*, 2002. Cited on page 82.

Bonnie J. Dorr and Douglas A. Jones. Role of word sense disambiguation in lexical acquisition: Predicting semantics from syntactic cues. In *COLING*, pages 322–327, 1996. Cited on page 48.

Bonnie J. Dorr and Olsen Broman Mari. Multilingual Generation: The Role of Telicity in Lexical Choice and Syntactic Realization. *Machine Translation*, 11(1–3):37–74, 1996. Cited on page 24, 42.

Bonnie J. Dorr, Pamela W. Jordan, and John W. Benoit. A Survey of Current Paradigms

in Machine Translation. Technical Report LAMP-TR-027, UMIACS-TR-98-72, CS-TR-3961, University of Maryland, College Park, December 1998. Cited on page 53, 77.

İlknur Durgar El-Kahlout and Kemal Oflazer. Initial Explorations in English to Turkish Statistical Machine Translation. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 7–14, New York City, June 2006. Association for Computational Linguistics. Cited on page 99.

Jason Eisner. Learning Non-Isomorphic Tree Mappings for Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL), Companion Volume*, pages 205–208, Sapporo, July 2003. Cited on page 59, 100.

Marcello Federico and Mauro Cettolo. Efficient Handling of N-gram Language Models for Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 88–95, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 76.

Christiane Fellbaum. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May 1998. Cited on page 24, 101.

Charles J. Fillmore, Charles Wooters, and Collin F. Baker. Building a Large Lexical Databank Which Provides Deep Semantics. In *Proceedings of the Pacific Asian Conference on Language, Information and Computation*, Hong Kong, 2001. Cited on page 24, 44.

Charles J. Fillmore. FrameNet and the Linking between Semantic and Syntactic Relations. In Shu-Cuan Tseng, editor, *Proceedings of COLING 2002*, pages xxviii–xxxvi. Howard International House, 2002. Cited on page 24, 44.

Sanae Fujita and Francis Bond. A Method of Adding New Entries to a Valency Dictionary by Exploiting Existing Lexical Resources. In *The 9th International Conference on Theoretical and Methodological Issues in Machine Translation, TMI 2002*, pages 42–53, Keihanna, Japan, March 2002. Cited on page 103.

Sanae Fujita and Francis Bond. A Method of Creating New Bilingual Valency Entries using Alternations. In Gilles Sérasset, editor, *COLING 2004 Multilingual Linguistic Resources*, pages 41–48, Geneva, Switzerland, August 28 2004. COLING. Cited on page 51, 103.

Sharon Goldwater and David McClosky. Improving statistical MT through morphological analysis. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 676–683, Morristown, NJ, USA, 2005. Association for Computational Linguistics. Cited on page 99.

Jan Hajič and Barbora Hladká. Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset. In *Proceedings of COLING-ACL Conference*, pages 483–490, Montreal, Canada, 1998. Cited on page 90.

Jan Hajič, Alexandr Rosen, and Hana Skoumalová. RUSLAN - systém strojového překladu z češtiny do ruštiny. Technical report, Výzkumný ústav matematických strojů, Prague, Czech Republic, 1987. Cited on page 17.

Jan Hajič, Eva Hajičová, Milena Hnátková, Vladislav Kuboň, Jarmila Panevová, Alexandr Rosen, Petr Sgall, and Hana Skoumalová. MATRACE – MAchine TRAnslation between Czech and English. In *Proceedings of the IBM Academic Initiative Projects Seminar, Praha, November 1992*, pages 75–82, Praha, 1992. České vysoké učení technické. Cited on page 47.

Jan Hajič, Jarmila Panevová, Zdeňka Urešová, Alevtina Bémová, Veronika Kolářová, and Petr Pajas. PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. In *Proceedings of The Second Workshop on Treebanks and Linguistic Theories*, volume 9 of *Mathematical Modeling in Physics, Engineering and Cognitive Sci-*

ences, pages 57–68. Växjö University Press, November 14–15, 2003 2003. Cited on page 23.

Jan Hajič. RUSLAN: an MT system between closely related languages. In *Proceedings of the third conference on European chapter of the Association for Computational Linguistics*, pages 113–117. Association for Computational Linguistics, 1987. Cited on page 17.

Jan Hajič. Complex Corpus Annotation: The Prague Dependency Treebank. In *Insight into Slovak and Czech Corpus Linguistics*, Bratislava, Slovakia, 2004. Jazykovedný ústav Ľ. Štúra, SAV. Cited on page 59.

Jan Hajič. *Disambiguation of Rich Inflection (Computational Morphology of Czech)*. Nakladatelství Karolinum, Prague, 2004. Cited on page 60, 86.

Jan Hajič, Martin Čmejrek, Bonnie Dorr, Yuan Ding, Jason Eisner, Daniel Gildea, Terry Koo, Kristen Parton, Gerald Penn, Dragomir Radev, and Owen Rambow. Natural Language Generation in the Context of Machine Translation. Technical report, Johns Hopkins University, Center for Speech and Language Processing, 2002. NLP WS'02 Final Report. Cited on page 59.

Jan Hajič, Jarmila Panevová, Eva Hajičová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, Marie Mikulová, Zdeněk Žabokrtský, and Magda Ševčíková Razímová. Prague Dependency Treebank 2.0. LDC2006T01, ISBN: 1-58563-370-4, 2006. Cited on page 18, 19, 23.

Keith Hall. k-best Spanning Tree Parsing. In *Proceedings of the ACL 2007*, Prague, Czech Republic, June 2007. Cited on page 80.

Dana Hlaváčková and Aleš Horák. VerbaLex - New Comprehensive Lexicon of Verb Valencies for Czech. In *Computer Treatment of Slavic and East European Languages*, pages 107–115, Bratislava, Slovakia, 2006. Slovenský národný korpus. Cited on page 22, 24.

Dana Hlaváčková, Aleš Horák, and Vladimír Kadlec. Exploitation of the VerbaLex Verb Valency Lexicon in the Syntactic Analysis of Czech. In *Text, Speech and Dialogue: 9th International Conference, TSD 2006*, volume LNAI 3658. Springer Verlag, September 2006. Cited on page 17, 103.

Tomáš Holan, Vladislav Kuboň, Karel Oliva, and Martin Plátek. On Complexity of Word Order. *Special Issue on Dependency Grammar of the journal TAL (Traitement Automatique des Langues)*, 41(1):273–300, 2000. Cited on page 57, 58.

Tomáš Holan. K syntaktické analýze českých(!) vět. In *MIS 2003*. MATFYZPRESS, January 18–25, 2003 2003. Cited on page 57, 106.

Albert Sydney Hornby. *Oxford Advanced Learner's Dictionary of Current English*. Oxford University Press, Oxford, 3 edition, 1974. Cited on page 47.

Liang Huang, Kevin Knight, and Aravind Joshi. Statistical Syntax-Directed Translation with Extended Domain of Locality. In *Proc. of 7th Biennial Conference of the Association for Machine Translation in the Americas (AMTA)*, Boston, MA, 2006. Cited on page 68, 80.

John Hutchins. The state of machine translation in Europe. In *Expanding MT horizons: proceedings of the Second Conference of the Association for Machine Translation in the Americas*, pages 198–205, Montreal, Quebec, Canada, October 1996. Cited on page 53.

John Hutchins. ALPAC: the (in)famous report. In S. Nirenburg, H. Somers, and Y. Wilks, editors, *Readings in machine translation.*, pages 131–135. The MIT Press, Cambridge, Mass., 2003. Originally published in MT News International 14, June 1996, 9-12. Cited on page 53.

John Hutchins. The first public demonstration of machine translation: the Georgetown-IBM system, 7th January 1954. Expanded version of AMTA-2004 paper.

`http://www.hutchinsweb.me.uk/GU-IBM-2005.pdf`, November 2005. Cited on page 53.

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 MT Summit III*, pages 101–106, 1991. Cited on page 51, 103.

Ray Jackendoff. *Semantic Structures*. The MIT Press, 1990. Cited on page 24, 42.

Aravind K. Joshi, Leon S. Levy, and Masako Takahashi. Tree adjunct grammars. *J. Comput. Syst. Sci.*, 10(1):136–163, 1975. Cited on page 57, 60.

Aravind K. Joshi, K. Vijay Shanker, and David Weir. The Convergence of Mildly Context-Sensitive Grammar Formalisms. Technical Report MS-CIS-90-01, University of Pennsylvania Department of Computer and Information Science, 1990. Cited on page 57.

George Karypis. CLUTO - A Clustering Toolkit. Technical Report #02-017, University of Minnesota, Department of Computer Science, November 2003. Cited on page 39.

Adam Kilgarriff. Language is never ever ever random. *Corpus Linguistics and Linguistic Theory*, 1(2):263–276, 2005. Cited on page 44.

Gilbert W. King. Stochastic Methods of Mechanical Translation. *Mechanical Translation*, 3(2):38–39, 1956. Cited on page 55.

Paul Kingsbury and Martha Palmer. From Treebank to PropBank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, Las Palmas, Spain, 2002. Cited on page 56.

Paul Kingsbury, Martha Palmer, and Mitch Marcus. Adding Semantic Annotation to the Penn TreeBank. In *Proceedings of the Human Language Technology Conference*, 2002. Cited on page 24.

Paul Kingsbury. Verb clusters from PropBank annotation. Technical report, University of Pennsylvania, Philadelphia, PA, 2004. Cited on page 48.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. Class-based construction of a verb lexicon. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 691–696. AAAI Press / The MIT Press, 2000. Cited on page 24.

Karen Kipper-Schuler. *VerbNet: A broad-coverage, comprehensive verb lexicon*. PhD thesis, University of Pennsylvania, 2005. Cited on page 24, 48.

Zdenek Kirschner and Alexandr Rosen. Apac - an experiment in machine translation. *Machine Translation*, 4(3):177–193, 1989. Cited on page 56.

Václav Klimeš. *Analytical and Tectogrammatical Analysis of a Natural Language*. PhD thesis, ÚFAL, MFF UK, Prague, Czech Republic, 2006. Cited on page 60, 81.

Jan Kocek, Marie Kopřivová, and Karel Kučera, editors. *Český národní korpus - úvod a příručka uživatele*. FF UK - ÚČNK, Praha, 2000. Cited on page 59.

Philipp Koehn and Hieu Hoang. Factored Translation Models. In *Proc. of EMNLP*, 2007. Cited on page 75.

Philipp Koehn and Kevin Knight. Empirical Methods for Compound Splitting. In *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 187–193, Morristown, NJ, USA, 2003. Association for Computational Linguistics. Cited on page 99.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic,

June 2007. Association for Computational Linguistics. Cited on page 68.

Philipp Koehn. Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models. In Robert E. Frederking and Kathryn Taylor, editors, *AMTA*, volume 3265 of *Lecture Notes in Computer Science*, pages 115–124. Springer, 2004. Cited on page 67, 86, 88, 89.

Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of EMNLP 2004*, Barcelona, Spain, 2004. Cited on page 77.

Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MT Summit X*, 2005. Cited on page 85.

Anna Korhonen. Subcategorization Acquisition. Technical Report UCAM-CL-TR-530, University of Cambridge, Computer Laboratory, Cambridge, UK, February 2002. Cited on page 30.

Květoslava Králíková and Jarmila Panevová. ASIMUT - A Method for Automatic Information Retrieval from Full Textts. *Explicite Beschreibung der Sprache und automatische Textbearbeitung*, XVII, 1990. Cited on page 102.

Geert-Jan M. Kruijff. 3-Phase Grammar Learning. In *Proceedings of the Workshop on Ideas and Strategies for Multilingual Grammar Development*, 2003. Cited on page 57.

Marco Kuhlmann and Mathias Möhl. Mildly context-sensitive dependency languages. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 160–167, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 57.

Elina Lagoudaki. Translation Memories Survey 2006: Users' perceptions around TM use. In *Proceedings of the ASLIB International Conference Translating and the Computer 28*, London, UK, November 2006. Cited on page 53.

Beth C. Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL, 1993. Cited on page 26, 41, 42, 48.

Kenneth C. Litkowski. Computational Lexicons and Dictionaries. In Keith Brown, editor, *Encyclopedia of Language and Linguistics (2nd ed.)*. Elsevier Publishers, Oxford, 2005. `http://www.clres.com/online-papers/ell.doc`. Cited on page 101, 106.

Yang Liu, Qun Liu, and Shouxun Lin. Log-linear models for word alignment. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 459–466, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. Cited on page 51, 103.

Markéta Lopatková and Jarmila Panevová. Recent developments of the theory of valency in the light of the Prague Dependency Treebank. In Mária Šimková, editor, *Insight into Slovak and Czech Corpus Linguistic*, pages 83–92. Veda Bratislava, Slovakia, 2005. Cited on page 20, 22, 25, 104.

Markéta Lopatková, Zdeněk Žabokrtský, and Václava Benešová. Valency Lexicon of Czech Verbs VALLEX 2.0. Technical Report 34, UFAL MFF UK, 2006. Cited on page 26.

Markéta Lopatková, Zdeněk Žabokrstký, and Karolína Skwarska. Valency Lexicon of Czech Verbs: Alternation-Based Model. In *Proceedings of LREC 2006*, pages 1728–1733. ELRA, 2006. Cited on page 26.

Markéta Lopatková, Zdeněk Žabokrtský, and Václava Kettnerová. *Valenční slovník českých sloves*. Univerzita Karlova v Praze, Nakladatelství Karolinum, Praha, 2008. In cooperation with Karolína Skwarska, Eduard Bejček, Klára Hrstková, Michaela Nová and Miroslav Tichý. Cited on page 22, 26.

Markéta Lopatková. Valency in the Prague Dependency Treebank: Building the Valency Lexicon. *Prague Bulletin of Mathematical Linguistics*, 79–80:37–60, 2003. Cited on page 24.

Adam Lopez and Philip Resnik. Word-Based Alignment, Phrase-Based Translation: What's the Link? In *Proc. of 7th Biennial Conference of the Association for Machine Translation in the Americas (AMTA)*, pages 90–99, Boston, MA, August 2006. Cited on page 103.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. Non-Projective Dependency Parsing using Spanning Tree Algorithms. In *Proceedings of HLT/EMNLP 2005*, October 2005. Cited on page 60, 81.

Kathleen R. McKeown, Regina Barzilay, David Evans, Vasileios Hatzivassiloglou, Judith L. Klavans, Ani Nenkova, Carl Sable, Barry Schiffman, and Sergey Sigelman. Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster. In *Proceedings of HLT 2002*, 2002. Cited on page 107.

Igor A. Mel'čuk. *Dependency Syntax - Theory and Practice*. Albany: State University of New York Press, 1988. Cited on page 56, 104.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. The NomBank Project: An Interim Report. In A. Meyers, editor, *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pages 24–31, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics. Cited on page 56.

Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolářová, Lucie Kučová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Magda Razímová, Petr Sgall, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, and Zdeněk Žabokrtský. Annotation on the tectogrammatical level in the Prague Dependency Treebank. Annotation manual. Technical Report 30, ÚFAL MFF UK, Prague, Czech Rep., 2006. Cited on page 18, 23, 74.

Eleni Miltsakaki, Rashmi Prasad, Aravind Joshi, and Bonnie Webber. The Penn Discourse TreeBank. In *Proceedings of Fourth International Conference on Language Resources and Evaluation, LREC 2004*, 2004. Cited on page 56.

Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223, 2001. Cited on page 60, 90.

Raymond J. Mooney. Learning for Semantic Interpretation: Scaling Up without Dumbing Down. In James Cussens and Sašo Džeroski, editors, *Learning Language in Logic, LLL'99*, volume LNAI 1925, pages 57–66, Berlin Heidelberg, 2000. Springer-Verlag. Cited on page 106.

Thai Phuong Nguyen and Akira Shimazu. Improving Phrase-Based SMT with Morpho-Syntactic Analysis and Transformation. In *Proceedings of the 7th Conference of the Association for Machine Translation of the Americas*, pages 138–147, August 2006. Cited on page 99.

Sonja Nießen and Hermann Ney. Toward hierarchical models for statistical machine translation of inflected languages. In *Proceedings of the workshop on Data-driven methods in machine translation*, pages 1–8, Morristown, NJ, USA, 2001. Association for Computational Linguistics. Cited on page 99.

Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, 2007. Cited on page 57.

Brian Oakley. To do the right thing for the wrong reason, the Eurotra experience. In *MT Summit V Proceedings*, Luxembourg, July 1995. Cited on page 53.

Franz Josef Och and Hermann Ney. A Comparison of Alignment Models for Statistical Machine Translation. In *Proceedings of the 17th conference on Computational linguis-*

tics, pages 1086–1090. Association for Computational Linguistics, 2000. Cited on page 70.

Franz Josef Och and Hermann Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *ACL*, pages 295–302, 2002. Cited on page 64, 87.

Franz Josef Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, 2003. Cited on page 90.

Franz Joseph Och. *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. PhD thesis, RWTH Aachen University, 2002. Cited on page 66.

Franz Josef Och. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of the Association for Computational Linguistics*, Sapporo, Japan, July 6-7 2003. Cited on page 76, 82, 87.

Franz Joseph Och. Statistical Machine Translation: Foundations and Recent Advances. Tutorial at MT Summit 2005, September 2005. Cited on page 106.

Stephan Oepen, Erik Velldal, Jan Tore Lønning, Paul Meurer, Victoria Rosén, and Dan Flickinger. Towards Hybrid Quality-Oriented Machine Translation. On Linguistics and Probabilities in MT. In *Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI-07)*, Skövde, Sweden, 2007. Cited on page 56.

Karel Oliva. A Parser for Czech Implemented in Systems Q. *Explicite Beschreibung der Sprache und automatische Textbearbeitung*, XVI, 1989. Cited on page 17.

Karel Pala and Pavel Ševeček. Valence českých sloves. In *Sborník prací FFBU*, pages 41–54, Brno, 1997. Cited on page 24, 47.

Karel Pala and Pavel Smrž. Building Czech Wordnet. *Romanian Journal of Information, Science and Technology*, 7(1-2):79–88, 2004. Cited on page 24.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, 2005. Cited on page 42.

Jarmila Panevová. *Formy a funkce ve stavbě české věty [Forms and functions in the structure of the Czech sentence]*. Academia, Prague, Czech Republic, 1980. Cited on page 19, 20, 34.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *ACL 2002, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, 2002. Cited on page 77.

Carl J. Pollard and Ivan A. Sag. *Head-driven Phrase Structure Grammar*. University of Chicago Press, Chicago, 1994. Cited on page 56.

Maja Popović and Hermann Ney. Improving Word Alignment Quality using Morpho-Syntactic Information. In *Proceedings of COLING 2004*, Geneva, Switzerland, August 23–27 2004. Cited on page 99.

Paul Procter, editor. *Longman Dictionary of Contemporary English*. Longman Group, Essex, England, 1978. Cited on page 48.

Jan Ptáček and Zdeněk Žabokrtský. Synthesis of Czech Sentences from Tectogrammatical Trees. In *Proc. of TSD*, pages 221–228, 2006. Cited on page 17, 22, 60, 78, 80, 81, 83.

J. Ross Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, 1986. Cited on page 38.

J. Ross Quinlan. Data Mining Tools See5 and C5.0, 2002. `http://www.rulequest.com/see5-info.html`. Cited on page 38, 48.

Christopher Quirk and Arul Menezes. Dependency Treelet Translation: The Conver-

gence of Statistical and Example-Based Machine-Translation? *Machine Translation*, 20(1):43–65, 2006. Cited on page 100.

Chris Quirk, Arul Menezes, and Colin Cherry. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 271–279. Association for Computational Linguistics, 2005. Cited on page 80.

Adwait Ratnaparkhi. A Maximum Entropy Part-Of-Speech Tagger. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*, University of Pennsylvania, May 1996. Cited on page 60, 90.

Stephen D. Richardson, William B. Dolan, Arul Menezes, and Monica Corston-Oliver. Overcoming the Customization Bottleneck Using Example-Based MT. In *Proceedings of the workshop on Data-driven methods in machine translation*, pages 1–8, Morristown, NJ, USA, 2001. Association for Computational Linguistics. Cited on page 56.

Stefan Riezler and III John T. Maxwell. Grammatical Machine Translation. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 248–255, Morristown, NJ, USA, 2006. Association for Computational Linguistics. Cited on page 82.

Alexandr Rosen, Eva Hajičová, and Jan Hajič. Derivation of underlying valency frames from a learner's dictionary. In *Proceedings of the 14th conference on Computational linguistics*, pages 553–559, Nantes, France, 1992. Association for Computational Linguistics. Cited on page 47.

Alexandr Rosen. In Defense of Impractical Machine Translation Systems. `http://utkl.ff.cuni.cz/~rosen/public/pognan.ps.gz`, 1996. Cited on page 53.

Pavel Rychlý and Pavel Smrž. Manatee, Bonito and Word Sketches for Czech. In *Proceedings of the Second International Conference on Corpus Linguisitcs*, pages 124–131, 2004. Cited on page 29, 44, 50.

Anoop Sarkar and Daniel Zeman. Automatic Extraction of Subcategorization Frames for Czech. In *Proceedings of the 18th International Conference on Computational Linguistics (Coling 2000)*, Saarbrücken, Germany, 2000. Universität des Saarlandes. Cited on page 49.

Sabine Schulte im Walde. *Experiments on the Automatic Induction of German Semantic Verb Classes*. PhD thesis, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, 2003. Published as AIMS Report 9(2). Cited on page 48, 49.

Jiří Semecký and Petr Podveský. Extensive Study on Automatic Verb Sense Disambiguation in Czech. In Petr Sojka, Ivan Kopecek, and Karel Pala, editors, *TSD*, volume 4188 of *Lecture Notes in Computer Science*, pages 237–244. Springer, 2006. Cited on page 22.

Jiří Semecký. *Verb Valency Frames Disambiguation*. PhD thesis, Charles University, Prague, 2007. Cited on page 35.

Petr Sgall, Eva Hajičová, and Jarmila Panevová. *The Meaning of the Sentence and Its Semantic and Pragmatic Aspects*. Academia/Reidel Publishing Company, Prague, Czech Republic/Dordrecht, Netherlands, 1986. Cited on page 18.

Hana Skoumalová. *Czech syntactic lexicon*. PhD thesis, Univerzita Karlova, Filozofická fakulta, 2001. Cited on page 24, 47.

David A. Smith and Jason Eisner. Minimum-Risk Annealing for Training Log-Linear Models. In *Proceedings of the International Conference on Computational Linguistics and the Association for Computational Linguistics (COLING-ACL), Companion Volume*, pages 787–794, Sydney, July 2006. Cited on page 76.

David A. Smith and Jason Eisner. Quasi-Synchronous Grammars: Alignment by Soft Projection of Syntactic Dependencies. In *Proceedings of the HLT-NAACL Workshop on Statistical Machine Translation*, pages 23–30, New York, June 2006. Cited on page 80.

Zoltan Somogyi, Fergus Henderson, and Thomas Conway. Mercury: An Efficient Purely Declarative Logic Programming Language. In *Proceedings of the Australian Computer Science Conference*, pages 499–512, Glenelg, Australia, February 1995. Cited on page 76.

Mark Stevenson. *Word Sense Disambiguation: The Case for Combinations of Knowledge Sources*. CSLI Publications, 2003. Cited on page 26, 33, 105.

Markéta Straňáková-Lopatková and Zdeněk Žabokrtský. Valency Dictionary of Czech Verbs: Complex Tectogrammatical Annotation. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002)*, volume 3, pages 949–956. ELRA, 2002. LN00A063. Cited on page 17, 22.

David Talbot and Miles Osborne. Modelling Lexical Redundancy for Machine Translation. In *Proc. of COLING and ACL 2006*, pages 969–976, Sydney, Australia, 2006. Cited on page 99.

Bernard Vauquois. La traduction automatique à Grenoble. Document de linguistique quantitative 24. Dunod, Paris., 1975. Cited on page 54.

Jean Véronis. Sense tagging: does it make sense? In *Corpus Linguistics by the Lune: a festschrift for Geoffrey Leech*. Peter Lang, 2003. Published version of a paper presented at the Corpus Linguistics'2001 Conference, Lancaster, U.K. Cited on page 22, 41.

Piek Vossen. Introduction to EuroWordNet. *Computers and the Humanities, Special Issue on EuroWordNet*, 32(2–3), 1998. Cited on page 24.

Yuk Wah Wong and Raymond Mooney. Learning Synchronous Grammars for Semantic Parsing with Lambda Calculus. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 960–967, Prague, Czech Republic, June 2007. Association for Computational Linguistics. Cited on page 106.

William A. Woods, Lawrence A. Bookman, Ann Houston, Robert J. Kuhns, Paul Martin, and Stephen Green. Linguistic knowledge can improve information retrieval. Technical Report TR-99-83, Sun Labs, December 1999. Cited on page 102.

Dekai Wu and Hongsing Wong. Machine translation with a stochastic grammatical channel. In *COLING-ACL*, pages 1408–1415, 1998. Cited on page 100.

Kenji Yamada and Kevin Knight. A decoder for syntax-based statistical mt. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 303–310, Morristown, NJ, USA, 2002. Association for Computational Linguistics. Cited on page 100.

Mei Yang and Katrin Kirchhoff. Phrase-Based Backoff Models for Machine Translation of Highly Inflected Languages. In *EACL 2006*, April 2006. Cited on page 99.

Zdeněk Žabokrtský and Markéta Lopatková. Valency Frames of Czech Verbs in VALLEX 1.0. In *Frontiers in Corpus Annotation. Proceedings of the Workshop of the HLT/NAACL Conference*, pages 70–77, May 6, 2004 2004. Cited on page 24.

Zdeněk Žabokrtský, Václava Benešová, Markéta Lopatková, and Karolina Skwarská. Tektogramaticky anotovaný valenční slovník českých sloves. Technical Report TR-2002-15, ÚFAL/CKL, Prague, Czech Republic, 2002. Cited on page 24.

Zdeněk Žabokrtský. *Valency Lexicon of Czech Verbs*. PhD thesis, Faculty of Mathematics and Physics, Charles University in Prague, 2005. Cited on page 21.

Zdeněk Žabokrtský. Tecto MT. Technical report, ÚFAL/CKL, Prague, Czech Republic, 2008. In prep. Cited on page 60, 81.

Zdeněk Žabokrtský. TectoMT: Highly Modular Hybrid MT System with Tectogrammatics Used as Transfer Layer. In *Proc. of the ACL Workshop on Statistical Machine Translation*, page In print, Columbus, Ohio, USA, 2008. Cited on page 82, 96.

Daniel Zeman. Can Subcategorization Help a Statistical Parser? In *Proceedings of the 19th International Conference on Computational Linguistics (Coling 2002)*, Taibei, Tchajwan, 2002. Zhongyang Yanjiuyuan (Academia Sinica). Cited on page 102.

Richard Zens, Oliver Bender, Sasa Hasan, Shahram Khadivi, Evgeny Matusov, Jia Xu, Yuqi Zhang, and Hermann Ney. The RWTH Phrase-based Statistical Machine Translation System. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, pages 155–162, Pittsburgh, PA, October 2005. Cited on page 67, 86.

Yi Zhang, Timothy Baldwin, and Valia Kordoni. The Corpus and the Lexicon: Standardising Deep Lexical Acquisition Evaluation. In *Proceedings of the ACL 2007 Workshop on Deep Linguistic Processing*, pages 152–159, Prague, Czech Republic, June 2007. Cited on page 30, 51.

Le Zhang. Maximum Entropy Modeling Toolkit for Python and C++. `http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html`, 2004. Cited on page 35, 38.

George Kingsley Zipf. The Meaning-Frequency Relationship of Words. *Journal of General Psychology*, 3:251–256, 1945. Cited on page 26.

Andreas Zollmann, Ashish Venugopal, and Stephan Vogel. Bridging the Inflection Morphology Gap for Arabic Statistical Machine Translation. In Robert C. Moore, Jeff A. Bilmes, Jennifer Chu-Carroll, and Mark Sanderson, editors, *HLT-NAACL*. The Association for Computational Linguistics, 2006. Cited on page 99.

Appendix A

Sample Translation Output

A.1 In-Domain Evaluation

This section illustrates the performace of various MT systems on articles from Project Syndicate.[1] We can talk about "in-domain" evaluation for our systems (etct and two configurations of Moses), because other texts from the same source are part of our training data.

Because both the original and the reference translations are publicly available on Project Syndicate website, we can speculate whether e.g. Google Translate had an opportunity to train on parts of this particular test set.

Source text, WMT 08 Commentary Test

Berlusconi at Bay

. . . Fifteen years later, Signor Berlusoni understood that the Italian state's monopoly of television would not survive and jump-started what became Italy's main privately owned media group. But you don't win in TV and the real estate business without the right political connections. On both occasions, Berlusconi outwitted his competitors by siding with the Socialists, at the time the rising stars of Italian political life. His long association with Bettino Craxi, Milan's most influential politician in the 1970's and Italy's prime minister through much of the 1980's, started early. On the other hand, political connections do not make a politician.

A Field Guide to Israeli Hawks

Prime Minister Ariel Sharon's announcement that he plans to dismantle Jewish settlements in the Gaza Strip, as well as some settlements in the West Bank, has shocked and caught people off guard both in Israel and around the world. Many denounced Sharon's plan as a trick. But that surprise was wrong-headed from the start. Despite the way it often looks to outsiders, debates in Israel about the future of the occupied territories have never been confined to hawks and doves. Like everything in Israel, the process is more complicated, especially where the hawks are concerned. Basically, there are two species of Israeli hawks: call one kind ideological and the other strategic.

[1] `http://www.project-syndicate.org/`

etct, factored output nodes, preserve structure BLEU 4.98%

Berlusconi v zálivu.

...Fifteen let už signoru berlusonus chápal, že italský stát je monopol televize, nakonec, nežije a, nastartuje začne, co se stalo, itálie je hlavní soukromá vlastní mediální skupina. Však vám je, zvítězí v televizi a skutečného nemovitosti obchodu správných politických vztahů. Oba příležitostí berlusconi outwit její konkurence stranění socialisté v době rostoucích hvězd italského politického života. Její dlouhodobá vazba bettino craxus milán je nejvlivnější politik 70 je a itálie je ministerská vláda velké 80 je začne brzy. Na druhé straně politické konexe jsou politik.

Oblasti vede izraelských jestřábů.

Premiér ministr ariel šaron je oznámení, že je hodlá odstranit židovských osad gaza strip jak, tak, některá urovnání na západním břehu, je, šokoval a chytili lidé gardy jak v izrael a po světě. Mnohém odsoudil, šaron je, plán trik. A to překvapení bylo správné , vede od začátku. Way, její často vypadá cizinců, se debaty izrael o budoucnosti okupovaných území nikdy týkaly jestřábů a holubičky. Všechna v izrael proces je složitější, především kde se jestřábi týkají. Vlastní jsou two druhové izraelských jestřábů : výzvou jistou takovou ideologické a další strategické.

TectoMT BLEU 9.28%

Berlusconi na zálivu.

...O patnáct roků později pán Berlusoni rozuměl, že by se italský státní monopol televize nepřežil a jump-started, co se stalo hlavní soukromě vlastní sdělovací skupinou Itálie. Ale nevyhrajete se v TV a ve skutečném realitním obchodu bez pravých politických spojení. Na obou příležitostech Berlusconi přelstil svoje konkurenty obklad se socialisty na času rostoucí hvězdném italského politického života. A jeho dlouhého sdružení s nulou politikem v roce 1970 Craxiho Milán nejvlivnější ministerský ministerský Itálie velké 80. let výchozích rané. Na jiné ruce politická spojení neučiní politika.

Polní vodítko k Izraelcům jestřábům.

Ministerské ministerské Ariel oznámení Sharonu, že plánuje demontáž židovských urovnání v Gaze Gazy i jako některá urovnání ve West bance šokovalo a chytilo lidi dozorce obojím v Izraeli a po světě. Mnoha vypověděl plán Sharonu jako trik. Ale toto překvapení bylo wrong-headed ze začátku. Navzdory způsobu, to často vypadá k cizincům, debaty v Izraeli o budoucnosti okupovaných území nikdy nebyly omezeny k jestřábům a k holubům. Jako vše v Izraeli proces je složitější, že se zejména kde jestřábi jsou týkáni. Vlastně jsou dva druhy izraelských jestřábů: hovor jeden věcný ideologický a jiné strategický.

PC Translator 2007 BLEU 8.48%

Berlusconi v úzkých

. . . Patnáct let později, pán Berlusoni rozuměl , že italský státní monopol na televizi by ne by nepřežila a strmý start co stal se Itálií- ovo hlavní v soukromém vlastnictví mediální skupina. Ale vy nevyhrajete v TV a obchodu nemovitostí bez pravých politických spojení. Na obou příležitostech, Berlusconi přechytračený jeho konkurenti výhybkou se Socialists, v době rostoucí hvězdy italského politického života. Jeho dlouhé spojení s Bettino Craxi, Milan- ovo nejvlivnější politik v 1970- ovo a Itálie- ovo ministerský předseda skrz velkou část 1980- ovo, začít brzo. Na druhé straně, politické spojení neudělají politika.

A polní průvodce po izraelských jestřábech

Ministerský předseda Ariel Sharon- ovo oznámení že on plánuje rozebrat židovská osídlení v Pásmu Ghazy, stejně jako nějaká osídlení na západním břehu, šokovali a chycené lidi mimo stráž v Izraeli a kolem světa. Mnoho odsuzovaného Sharon- ovo plánu jak trik. Ale to překvapení bylo zarputilé od začátku. Navzdory cestě to často vzhlíží k outsider, diskuse v Izraeli o budoucnosti obsazených oblastí nikdy nebyly uvězněné v jestřábech a skokách. Jako všechno v Izraeli, proces je víc komplikovaný, zvláště kde jestřábové se týkají. Základním způsobem, tam jsou dva druzi izraelských jestřábů: volat jednoho druha ideologického a další strategického.

Moses T+C, CzEng 0.7 data only BLEU 14.64%

Berlusconi v koutě

. . . Patnáct let nato, přičemž Berlusoni signor italského státního monopolu televize, a to by nebylo přežít a impulsem začalo to, co se stalo v Itálii v hlavním soukromém vlastnictví mediální skupiny. Ale cožpak zvítězit v televizi a nemovitosti podnikatelské bez patřičné politické konexe. V obou případech Berlusconi outwitted soupeřům tím, že stranění socialistů, v době rostoucí hvězdy italského politického života. Jeho dlouhá spojitost se Bettino Craxi, Milan 's nejvlivnější političkou v sedmdesátých a italského premiéra po většinu osmdesátých let, začala brzy. Na druhé straně však politické konexe nejsou vytvořit politik.

Směřování k izraelským jestřábům oblasti,

Ministerského předsedy Ariela Šarona "je oznámení, že plány na odstranění židovských osad v pásmu Gazy, stejně jako některé osady na západním břehu, má zaskočen a chycen lidí mimo gardy, a to jak v Izraeli a po celém světě. Mnozí odsoudili Šaronův plán jako trik. Ale to, že překvapení se mýlí hlavou od začátku. Navzdory tomu, aby byla často vypadá pro nezasvěcené debaty v Izraeli o budoucnosti na okupovaných územích nebyly nikdy neomezují na jestřábi a holubi. Jako vše v Izraeli, proces je mnohem složitější, zvláště pokud jestřábi jsou znepokojeni. V zásadě existují dva druhy izraelských jestřábů: volání jednoho druhu ideologické a ostatní strategické.

Moses T+C, LM from SYN2006 BLEU 15.91%

Berlusconi v zálivu

...O patnáct let později, signor Berlusoni pochopili, že italský státní monopol televize by to nepřežije a skok - začalo to hlavní, co se stalo v Itálii v soukromém vlastnictví mediální skupiny. Ale vy zvítězit v televizi a realitní činnost bez patřičné politické konexe. V obou případech Berlusconi outwitted soupeřům tím, že stranění socialistů, v době stoupající hvězdy italského politického života. S Bettino Craxi jeho dlouhé sdružení, Milan 's nejvlivnějším politikem v sedmdesátých a italský premiér po většinu z osmdesátých let, začal brzy. Na druhé straně však politické konexe nejsou vytvořit politikem.

Vodítko pro oblasti izraelských jestřábů

Premiéra Ariela Šarona "je oznámení, že plány na odstranění židovských osad v pásmu Gazy, stejně jako některé osady na západním břehu, byl šokován a chytil od lidí, kteří střeží jak v Izraeli a na celém světě. Mnozí odsoudil Šaronovu plánu jako trik. Ale to překvapení bylo špatné - stojí od samého začátku. Navzdory tak, jak to často vypadá pro cizince v Izraeli, debaty o budoucnosti na okupovaných územích nikdy nebyly omezeny na jestřábi a holubi. Jako vše v Izraeli, proces je mnohem složitější, a to zejména v případě, že jestřábi jsou znepokojení. V zásadě existují dva druhy izraelských jestřábů: volání jednoho druhu ideologické a další strategické.

Google Translate, as of May 15, 2008 BLEU 21.14%

Berlusconi v úzkých

...Patnáct let poté, Signor Berlusoni pochopil, že italský státní monopol na televizní by nepřežili a jump-začalo to, co se stala Itálie v soukromém vlastnictví, hlavní mediální skupiny. Ale nemusíte vyhrát v TV a realitní činnost bez právo politické spojení. V obou případech Berlusconi outwitted jeho konkurenti o vlečka s socialistů, v době stoupající hvězdy na italského politického života. Jeho dlouhé spolupráci s Bettino Craxi, Milan nejvlivnějším politikem v 1970 a Itálie premiér prostřednictvím mnohem z roku 1980 se začal brzy. Na druhou stranu, politické propojení nenuťte politikem.

Pole Průvodce izraelské Hawks

Ministerský předseda Ariel Sharon je oznámení, že plány likvidace židovských osad v pásmu Gazy, stejně jako některých osad na západním břehu Jordánu, byl šokován a ulovené lidí z stráže, jak v Izraeli a po celém světě. Mnoho vypovězena Šaronův plán jako trik. Ale to překvapení bylo špatné-čele od začátku. Navzdory tomu, že způsob, jak to často vypadá na outsidery, diskuze v Izraeli o budoucnosti okupovaných území, nebyla nikdy omezena na jestřábi a holubice. Stejně jako všechno v Izraeli, že celý proces je mnohem složitější, zejména pokud se jedná o jestřábi. V zásadě existují dva druhy izraelských jestřábů: Výzva jednoho druhu ideologických a jiných strategických.

Reference translation

Berlusconi v úzkých

. . . O patnáct let později signor Berlusconi pochopil, že se italský státní televizní monopol neudrží, a chopil se příležitosti, která dala vzniknout největší italské mediální skupině v soukromých rukou. V televizním a realitním byznysu ovšem nemůžete vítězit bez správných politických styků. V obou případech Berlusconi vyzrál nad svými konkurenty tím, že stranil socialistům, tehdejší stoupající hvězdě italského politického života. Velmi brzy začalo jeho dlouholeté přátelství s Bettinem Craxim, nejvlivnějším milánským politikem 70. let a italským ministerským předsedou po většinu 80. let. Na druhé straně platí, že politické konexe nevytvoří politika.

Klíč k určování izraelských jestřábů

Prohlášení ministerského předsedy Ariela Šarona, že hodlá odstranit židovské osady z pásma Gazy a některé osady ze Západního břehu Jordánu, šokovalo a zaskočilo lidi jak v Izraeli, tak po celém světě. Mnozí Šaronův plán odsoudili jako úskok. Ona překvapenost ale byla od počátku pomýlená. Navzdory tomu, jak se věc často jeví cizincům, vnitroizraelské debaty o budoucnosti okupovaných území se nikdy neomezovaly na jestřáby a holubice. Tento proces, jako všechno v Izraeli, je složitější, obzvláště co se jestřábů týče. V zásadě existují dva druhy izraelských jestřábů: jednomu říkejme ideologický a druhému strategický.

A.2 Out-of-Domain Evaluation

This sections illustrates the performance of various MT systems on news text. For our contributions (etct and two setups of Moses), we can talk about evaluation out of the original domain, because no texts from a similar source or of a similar type are available in our training data.

As this particular test set was translated on demand for the purposes of WMT 08, we can be nearly sure that none of the third-party systems had access to the reference translations.

Source text, WMT 08 News Test

Food: Where European inflation slipped up

The skyward zoom in food prices is the dominant force behind the speed up in eurozone inflation. November price hikes were higher than expected in the 13 eurozone countries, with October's 2.6 percent yr/yr inflation rate followed by 3.1 percent in November, the EU's Luxembourg-based statistical office reported. Official forecasts predicted just 3 percent, Bloomberg said. As opposed to the US, UK, and Canadian central banks, the European Central Bank (ECB) did not cut interest rates, arguing that a rate drop combined with rising raw material prices

and declining unemployment would trigger an inflationary spiral. The ECB wants to hold inflation to under two percent, or somewhere in that vicinity.

New Russia-Ukraine gas row fears

A fresh gas price dispute is brewing between Ukraine and Russia, raising the risk that Russian exports of the fuel to western Europe may be affected. Most of Russia's gas exports to the European Union (EU) are piped through Ukraine and any row between the two nations is keenly watched. Kiev has warned that if Moscow raises the price it has to pay for the gas it will charge Russia higher transit fees. A previous dispute between the two last year reduced supplies to EU states.

etct, factored output nodes, preserve structure BLEU 3.36%

Food :, když kde evropská inflace zakopla.

Skyward zoom potravin cen je dominantní síla rychlosti vysoké eurozone inflace. Listopadu cenové zvýšení bylo vyšší, než očekával ve 13 eurozone zemích, říjen 2.6 procenta yr / yr inflace míra šel 3.1 procenta v listopadu, unie je lucembursko až, založený statistický úřad report. Představitel odhady předpovídal pouhé 3 procenta, bloomberg řekl. Odmítal usa, británie a kanadských centrálních bank evropská centrální banka ecb omezí úrokové sazby, tvrdil, že míry pokles, spojuje rostly hrubé materiálu ceny a klesala zaměstnanosti by vyvolá inflační spirálu. Ecb chce, má inflaci two procenta a ona v tomto okolí.

Nová rusko ukrajiny plynu řada se obává.

Nového plynu ceny sporu, je, brewing mezi ukrajinou a mezi rusko zvýšil riziko, že ruské vývozy paliva západní evropa mohly ovlivňovat. Největší rusko je plynu exporty evropské unie eu vzdušné ukrajiny a jakákoli řada mezi two zeměmi naléhavě sleduje. Kyjev varoval, že, moskva zvýší cenu, je má, platí plynu, její zaplatí rusko vyšší dopravy poplatky. Poslední spor mezi two posledním rokem snížil zdroje eu států.

TectoMT BLEU 6.94%

Potravina: kde evropská inflace klopýtla.

Skyward, že se zvětší, v cenách potraviny je dominantní platnost za rychlostí nahoru v eurozóně inflaci. Že zvýšení listopadu ceny byla vyšší, než se očekával ve 13 eurozónách zemích s říjnem 2,6 desetiprocentní yr/yr inflační sazbou následující 3,1 procentem v listopadu, Luxembourg-based statistický úřad EU uvedl. Že úředník předpovědi předpověděl právě 3 procenta, Bloomberg řekl. Že se stavěl proti USA proti UK a proti kanadským centrálním bankám, Evropan centrální banka (ECB) nesnížila úrokové sazby člověk, že by paušální kapka kombinovaná růst surových cen materiálu a poklesem nezaměstnanosti vyvolala inflační spirály. ECB chce držet inflaci k pod dvěma procentu nebo někde v této blízkosti.

Nové e plynové řádky strachy.
Čerstvé plynové cenové sporné, že je pivo mezi Ukrajinou a mezi Ruskem zvýšení, riziko že ruské vývozy paliva do západní Evropy mohou být ovlivněny. Největší vývozů Ruska plynu do Evropana svazu (EU) je píchnut Ukrajinou a jakýkoli řádek mezi dvěma národy pronikavě je sledován. Kyjev varoval, že, pokud Moskva zvýší cenu, že to má zaplatit za plyn, to bude účtovat Rusko vyšší tranzitní poplatky. Předchozí spor mezi dvě posledním rokem snížil dodávky na EU státy.

PC Translator 2007 BLEU 8.41%

Jídlo: Kde evropská inflace klopýtla
K nebi najet transfokátorem potravinové ceny je dominantní síla za rychlostí nahoru v eurozone inflaci. Listopad zvýšení cen byla vyšší než očekávaný v 13 eurozone zemích, s říjnovým 2.6 procent yr/yr míry inflace následované 3.1 procent v listopadu, EU- ovo Luxembourg - based statistický úřad ohlásil. Oficiální předpovědi předpovídaly jen 3 procent, Bloomberg řekl. Jak protichůdný k US, UK, a kanadské ústřední banky, Evropská centrální banka (ECB) ne řeže úrokové sazby, argumentování ten přepočítací pokles v kombinaci se stoupáním surovina ceny a sestupná nezaměstnanost spoušť inflační spirála. ECB Chci držet inflaci pod dva procent, nebo kdesi v tom sousedství.
Nová Russia - Ukraine plynová řada bojí se
A čerstvý plynový cenový spor vaří mezi Ukrajinou a Ruskem, pěstování riziko ty ruské exporty paliva západní Evropa může být ovlivněný. Většiny ruských plynových exportů k Evropské unii (EU) jsou vedení potrubím skrz Ukrajinu a nějaký /každý /žádný řada mezi dvěma národy je nadšeně sledovaný. Kyjev varoval že jestli Moskva zvedne cenu, kterou to musí platit za plyn, který to bude účtovat Rusku vyšší poplatky tranzitu. A předchozí spor mezi dvěma minulým rokem snížené dodávky EU stojí.

Moses T+C, CzEng 0.7 data only BLEU 9.75%

Jídlo: kam evropská inflace sklouzla nahoru,
K tomu, že vzlétl ještě výše přiblížit ceny potravin je dominantní silou v pozadí urychlí v eurozóně inflace. Listopadové cenové zvýšení bylo vyšší, než se očekávalo, že v říjnu 13 země eurozóny, s tím, že 2, 6 procenta Yr / Yr míru inflace následovaná 3, 1 procenta v listopadu, EU a Lucembursko - založený statistický úřad ohlásil. Oficiální předpovědi předpověděly právě, 3 procent, Bloomberg řekl. Na rozdíl od USA, Británii a kanadskou centrální banky, evropská centrální banka (ECB), nikoliv snížit úrokové sazby, a tvrdí, že sazby klesnou spojovány s rostoucími cenami surovin a klesající nezaměstnanosti vyvolává inflační spirále. ECB si chce udržet inflaci, aby se podle dvou procent, nebo někde v těchto místech.
Nové Rusko - Ukrajina plynu obává pořadí.

A čerstvé ceny plynu bublají spor mezi Ukrajinou a Ruskem, zvýší riziko, že ruský export paliva pro západní Evropu, může být ovlivněn. Většina ruských vývozů plynu do evropské unie (EU) je zaveden prostřednictvím Ukrajiny a každá řada mezi oběma zeměmi je naléhavě sledoval. Kyjev upozornila, že pokud Moskva zvýší cena, která má platit za plyn bude účtovat Rusko vyšších tranzitních poplatků. A předchozí spor mezi těmito dvěma loni sníží dodávky do členských států EU.

Moses T+C, LM from SYN2006 BLEU 11.93%

V případě potravin sklouzla do evropské inflace:
Vzhledem k tomu, že vzlétl ještě výše přiblížit ceny potravin je dominantní silou za urychlení v eurozóně inflace. V listopadu byly vyšší než očekávané zvýšení ceny v oblasti země eurozóny, 13. října 's 2, 6 procenta s Yr / Yr míru inflace 3, 1 procenta, následované v listopadu, EU v Lucemburku statistický úřad hlásí založený. Jen 3 procenta oficiální předpovědi předpověděl, Bloomberg řekl. Na rozdíl od amerických, britských a kanadských centrálních bank, evropská centrální banka (ECB), nikoliv snížit úrokové sazby, a tvrdí, že sazby klesnou spolu s rostoucími cenami surovin a klesající nezaměstnanosti vyvolalo inflační spirále. ECB chce udržet inflaci do dvou procent, nebo někde v těchto místech.
Nové Rusko - Ukrajina plynu obává řádek.
A čerstvé ceny plynu bublají spor mezi Ukrajinou a Ruskem, zvýší riziko, že ruský vývoz paliva do západní Evropy, může být ovlivněn. Většina ruských vývozů plynu do evropské unie (EU) jsou píšťala přes Ukrajinu a každý řádek mezi oběma národy je naléhavě střežen. Kyjev již varoval, že pokud Moskva zvyšuje cenu, která se má platit za plyn bude účtovat vyšší tranzitní poplatky v Rusku. A předchozí spor mezi dvěma v loňském roce snížené dodávky pro státy EU.

Google Translate, as of May 15, 2008 BLEU 12.82%

Strava: Tam, kde se evropské inflace strčila do
Na obloze zoom cen potravin, je dominantní silou urychlení inflace v eurozóně. Listopad růst cen byl vyšší než se očekává v 13 zemích eurozóny, v říjnu na 2,6 procenta r / r míra inflace následoval o 3,1 procenta v listopadu, EU, Lucembursko-založené statistického úřadu hlášena. Úřední prognózy předpovědět jen 3 procenta, Bloomberg řekl. Na rozdíl od USA, Velké Británii, a kanadské centrální banky, Evropská centrální banka (ECB) nebyla snížení úrokových sazeb a tvrdil, že míra poklesu v kombinaci s rostoucí ceny surovin a klesající nezaměstnanosti by podnítit inflační spirály. ECB chce držet inflaci pod dvě procenta, nebo někde v blízkosti.
Nové Rusko-Ukrajina plynový řádku obavy
A fresh cen zemního plynu je pivovarské spor mezi Ukrajinou a Ruskem, a tím zvýšit riziko, že ruský vývoz paliva do západní Evropy může být ovlivněna. Většina z ruského vývozu zemního plynu do Evropské unie (EU) je propojen přes Ukrajinu

a jakékoli řádku mezi oběma národy je horlivě sledoval. Kyjev má varoval, že pokud Moskva se zvyšuje cena, kterou musí zaplatit za benzín, že Rusko bude účtovat vyšší poplatky za tranzit. Předchozí spor mezi dvěma posledním roce snížena dodávky do států EU.

Reference translation

Inflace v Evropě poskočila kvůli potravinám

Zrychlující se inflace naměřená v eurozóně je způsobena především neustálým růstem cen potravin. Listopadový růst cen ve 13 zemích eurozóny byl nad očekávání vyšší, po 2,6 procenta v říjnu byla zaregistrována roční inflace 3,1 procenta, oznámil lucemburský statistický úřad Unie. Oficiální předpověď předpokládala pouze 3 procenta, sdělila agentura Bloomberg. Na rozdíl od americké, britské a kanadské emisní banky Evropská centrální banka (ECB) nesnížila základní úrokovou sazbu s tím, že snížení by spolu se zvyšujícími se cenami surovin a klesající nezaměstnaností vedlo ke vzniku inflační spirály. ECB by ráda udržela míru inflace pod dvěma procenty, ovšem v jejich blízkosti.

Obavy z nové hádky o plyn mezi Ruskem a Ukrajinou

Mezi Ruskem a Ukrajinou právě probíhá spor o ceny zemního plynu, a tak se zvyšuje riziko toho, že mohou být ovlivněny ruské dodávky tohoto paliva do západní Evropy. Většina ruského paliva vyváženého do Evropské unie (EU) je vedena potrubím přes Ukrajinu a jakýkoliv spor mezi těmito dvěma zeměmi je ostře sledován. Kyjev varoval, že pokud Moskva zvedne Ukrajině ceny plynu, bude Rusku účtovat vyšší tranzitní poplatky. Předchozí spor mezi těmito dvěma minulý rok snížil dodávky do států EU.

List of Figures

List of Tables