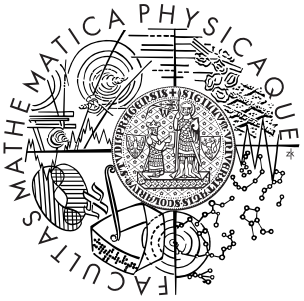
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Matěj Korvas

Empirické meze vybraných modelů strojového překladu

Empirical Limits of Selected Machine Translation Models

Ústav formální a aplikované lingvistiky

Vedoucí bakalářské práce: RNDr. Ondřej Bojar, PhD.

Studijní program: Obecná informatika

2010

Děkuji Karolíně, která o mě všemožně pečovala, když práce nebrala konce. Bez ní bych ji stěží někdy dopsal. Děkuji svému vedoucímu za bezvadné vedení a neustálou nápomoc. Otevřel mi nové obzory. Také děkuji rodině za skvělé zázemí, a v neposlední řadě všem lidem z Ústavu formální a aplikované lingvistiky za nesmírnou ochotu i toleranci.

Prohlašuji, že jsem svou bakalářskou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce a jejím zveřejňováním.

V Praze dne 5. 8. 2010 Matěj Korvas

Contents

Název práce: Empirické meze vybraných modelů strojového překladu
Autor: Matěj Korvas
Katedra (ústav): Ústav formální a aplikované lingvistiky
Vedoucí bakalářské práce: RNDr. Ondřej Bojar, PhD.
E-mail vedoucího: bojar@ufal.mff.cuni.cz

Abstrakt: Práce navrhuje míry, kterými lze kvantifikovat možnosti a náročnosti různých překladových modelů. Snaží se popsat vlastnosti překladových modelů na obecné rovině, nezávisle na vnitřním způsobu jejich fungování. Na extrémních případech stanovuje empirické meze modelů vzhledem k navrženým mírám a dále ověřuje použitelnost měr na skutečných překladových modelech. Cílem práce je poskytnout představu, které překladové modely mají přirozeně snazší práci za zachování dobrých výsledků. Důraz je kladen na volbu základní jazykové jednotky, na kterou model rozkládá vstupní text.
Klíčová slova: strojový překlad, měření náročnosti překladových modelů, dosažitelnost referenčního překladu

Title: Empirical Limits of Machine Translation Models
Author: Matěj Korvas
Department: Institute of Formal and Applied Linguistics
Supervisor: RNDr. Ondřej Bojar, PhD.
Supervisor's e-mail address: bojar@ufal.mff.cuni.cz

Abstract: In this thesis, we design measures to quantify capabilities and complexity of various translation models. We strive to describe properties of translation models in general, without regards to inner workings of the given translation model. On extreme cases, we determine empirical limits of translation models with regard to the designed measures and also prove usability of the measures with real translation models. The aim of this thesis is to give an idea of which translation models have naturally easier work to do, while keeping good quality of translation. The emphasis is put on choice of the basic language unit into which the translation model tries to decompose the input text.
Keywords: machine translation, measuring complexity of machine translation, reachability of reference translation

Chapter 1

Introduction

This thesis focuses on machine translating. To begin with, let us first describe what we mean by the word *translating*.

1.1 Scheme of translation

Translating is the process of expressing the same meaning in another language. Humans who speak both the source and target language of a particular translation problem usually have good idea about what the right translation is. However, one should realize that there are always more than one way how to express a given meaning in a given language, and that no possible counterpart of a given source language sentence in the target language has exactly the same meaning. We believe that even for the simplest sentences, there is a difference between its source and target form, caused by possible connotations or different usage distribution, which are in turn caused by the languages differing as a whole.

We will use the parallel Czech-English corpus CzEng 0.9 [2] for our experiments, which consists of pairs of segments stated to have the same meaning. But, before we draw any conclusions from results of the experiments, we need to be aware of several facts about the actual relation between Czech and English segments in the corpus and about expected relation between a Czech sentence and its translation into English (or vice versa).

1.1.1 Translation as a relation

In any translation, a meaning is necessarily present (although certain class of translation models successfully ignores it). This is the meaning that should be common to the sentences in both languages. As noted above, meanings of the sentences necessarily differ. Moreover, some other sentences in the target language have meanings similar to given translations, and some of them may even be better translations for the given source sentence.

Alternatively, this idea can be shown at the Vauquois triangle. We claim that the analysis and generation are rather general relations than functions. Moreover, not even the top point of the

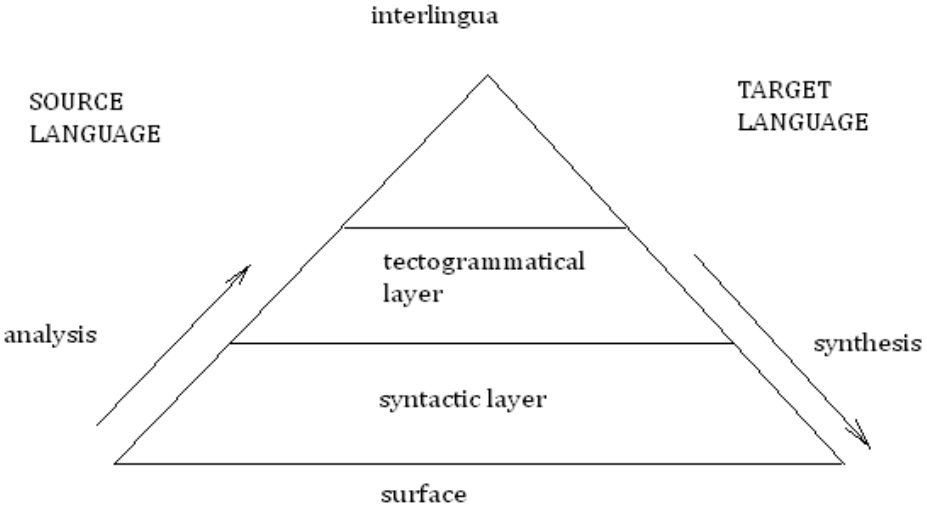


Figure 1.1: Vauquois triangle

triangle can be seen as unique on the way from the source sentence to the target one—we are given only the surface form of the sentence, which can have several meanings.

1.1.2 Corpus as a (random) sample

While the whole process of translation is fuzzy, even more randomness is brought by available resources describing the languages, by corpora. Ideally, properties of translation models should be determined based on whole probability space of both the languages, containing all possible translation pairs together with the probability that they will occur in a "random" pair of parallel texts.

However, the probability space mentioned above can be approximated by a (large enough) sample. That is what parallel corpora present. Bearing the fact about translation being a general relation in mind, we should see that any parallel corpus is a (random) sample of segments in the source language, assigning a sample of their translations to them. The corpus, as such, provides only a sample of length 1 for each space of possible translations for a given source segment. However, it can be factored by the source segments, which yields, generally, larger samples for each source segment.

Ideally, corpora would present *random* samples. Unfortunately, we cannot assume this about the real corpus we will use. They are biased, generally speaking, to written texts which are easily available in large amounts. Apart from that, they may contain errors caused by mistakes in deciding about particular pairs of segments, whether they were originally meant as translations of each other. However, we will neglect this noise with the excuse that we will use data which are subject to the same noise, both for testing and for training.

1.2 Scheme of translation model

Before we proceed to formulating the question for this thesis, let us first briefly describe general construction of a translation model we will assume.

There are two basic types of translation models—rule-based and statistical. The basic difference between them is that former ones come with language-specific knowledge already implemented, while latter ones have implemented only general rules for how to acquire the knowledge from data. We will only examine the latter ones, as there is no dependency of rule-based models on training data, and thus nothing to examine for us.

The work a *statistical translation model* (or, hereafter, just TM) has to perform when applied to a translation problem, can be decomposed into phases:

1. Learn from training data.
   (a) Analyze each pair of segments into basic *units*.
   (b) Figure out possible mutual correspondence of respective units (a.k.a. *alignment*).
   (c) Mark the found corresponding pairs of units with probability that they really correspond to each other in meaning.
2. Calibrate parameters on (another) training data.
3. Use the outcome of first two phases to find the best translation for the given text.

In this procedure, some points are of special importance for us. First, we expect every TM to produce its *translation lexicon* after the phase 1, which may get trimmed after the later phase 2. The lexicon stores all language-specific knowledge the TM has. Second, the term *unit* used in 1a will be fundamental in our research. Basically, we will use the concept of language unit to classify TMs, and then to describe their various properties with respect to the unit they use.

We assume that all TMs follow this sequence of phases. Some TMs may skip some of the phases, especially the trivial ones we will present.

1.3 Question for this thesis

With the presumptions formulated above, we can now specify precisely the question for this thesis.

We will use a parallel Czech-English corpus to measure performance of various TMs. We will try to determine empirical limits on reachable quality of translation for those TMs, as well as uncertainty in getting the translation. All results will be functions of size of the corpus used for training and of the basic language unit used by that particular TM.

Different basic units lead to different view of the ⟨segment in the source language⟩–⟨segment in the target language⟩ relation: some may correlate well with the surface, thus allowing for easy analysis and synthesis, while leaving the transfer phase (as illustrated in Fig. 1.1) complicated, while other can get nearer to the structure of meaning, thereby making the relation of the TMs internal representation and the surface form more complicated, with the benefit of providing simpler relation

for the transfer. Some TMs might even use such units that neither correlate well with the surface form, nor do they make the transfer easier. We will try to reflect this property of different TMs using our measures.

For us to be able to measure the TMs and to compare the results, we need to find measures which are appropriate and well interpretable. The measures for adequacy of translation should provide simple characteristics of the complicated relation of space of translations as seen by the TM and by the corpus. Measures for uncertainty should present a counterweight for the former ones.

1.4 Outline of the research

First, we will find some measures which we expect to be suitable for our task. Then, we will run experiments with trivial TMs which we will construct for that single purpose. It will *a)* give us an idea about extremal values for measures we are going to use; *b)* help us modify the measures when needed; and also *c)* allow us to measure some parameters of the underlying corpus we use. The last point can hold true thanks to easy interpretability of results from experiments with trivial models.

After having set some basic framework using toy TMs, we will proceed to more elaborate ones and apply measures obtained from the initial experiments to them. We will include TMs based on as different language units as possible. Main categories of TMs we want to examine include *1.* flat phrase translation models which use for transfer predominantly the lowest level of abstraction in Vauquois triangle (see Fig. 1.1), and *2.* translation models with transfer at a deeper syntactic level.

1.5 Overview of experiments conducted

The table 1.1 summarizes all experiments we actually conducted. There were experiments we were planning to conduct, but we did not conduct them eventually. We did not measure later developed measures with the model Sen, which we had already finished experimented with at that time. These measurements are approximated using the TM Pas. We did not experiment with t-factored translation with longer phrases, as intended, for its high computational requirements. The translation lexicon for Tfact got unfortunately removed when the disk space allocated for our experiments ran out, and the IV-n measure was not measured for Tfact for technical hurdles. The IV-n is also not measured for Pas nor Sen, because at the time IV-n was included, data for those models were not easily accessible any more.

	Extreme			Phrase			Factored				Hypothetical	
							Afact			Tfact		
	Sen	Pas	Ltr	1	3	10	1	3	10	1	Aedge	Tedge
lexicon size	✓	✓	✓	✓	✓	✓	✓	✓	✓		–	–
IV-n			✓	✓	✓	✓	✓	✓	✓		✓	✓
OOV(TM)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–
reachability	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–
dec. entropy	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–
TM perplexity		✓	✓	✓	✓	✓	✓	✓	✓	✓	–	–

Table 1.1: Overview of conducted experiments. Each column corresponds to a type of TM and each row to a measure applied to it. The numbers 1, 3, and 10 express maximum length of phrases used for translating. The abbreviations in names of TMs, as well as the measures applied, are described in detail in the following text. Points marked with the sign "✓" correspond to experiments that we conducted. The sign "–" is at combinations of rows and columns that do not describe any possible experiment (e.g. lexicon size cannot be measured without having an actual TM). Empty fields correspond to experiments that we did not conduct.

Chapter 2

Developing measures

Before we start with experiments, we have to have prepared the measures to apply. We will discuss their choice in this chapter.

2.1 Level of abstraction

We believe that a well designed TM should fit well to any given corpus, learning specific properties of the language with certainty, but not with size of its translation lexicon, proportional to size of the corpus. The TM should be able to abstract from repeated patterns.—Rather than remembering all permissible pairs of language units, it should extract rules for them. This requirement is based not only on aesthetic feeling; the rate of growth of the translation lexicon has practical impacts on complexity of the translation process, on the time needed to compute the translation. Based on this argument, we will use *size of the translation lexicon* as another measure.

It could be discussed whether the lexicon may be measured in the form the TM uses it, because it may contain also some information only useful for faster working with it. However, we will disregard this issue, presuming that an index or any other subsidiary structure included in the lexicon is negligible in comparison with amount of the raw useful information. Bearing this in mind, we will focus rather on the rate of growth of the lexicon than on its absolute size.

Another simple but useful measure for ability to abstract, or generality, is the *out of vocabulary rate* (or, hereafter, just *OOV*). We expect models based on different language units to differ significantly with regard to OOV.

We are also going to employ a generalization of OOV. Besides counting only numbers of units out of vocabulary, we are going to obtain a histogram of how many occurrences a particular unit had in the training data. We will call this measure *IV-n*, with n as a variable for the number of occurrences of the unit in training data. OOV is then a special case of IV-n for $n = 0$. This measure, IV-n, in contrast to OOV, will be only obtained without actually using any TM for translating—it will be a purely empirical measure, describing the relation of training and testing data.

These two measures of TM's ability to abstract need to be complemented by other measures, that would discard TMs with high ability to abstract for the price that the abstraction is not very useful for the task. The ideal TM should be tailored to the structure of natural languages, able to draw

knowledge from training data which is successfully reusable for a sample of pairs of segments that have in common with the training data the structure of the language. That is, it should perform well on unseen data while having reasonably abstracted from the training data. Hence, we will always compare size of the translation lexicon to measures of applicability, in particular to reachability and TM perplexity.

2.2 Reachability and TM perplexity

Probably the most easy-to-obtain parameter of each TM are statistics of *reachability of reference translations*. Not only are they easily available, but also they can tell a lot about performance of a TM. Therefore, we are going to measure them for each of the TMs.

Rate of reachable pairs of segments can be viewed as an instance of a measure of recall—, and because recall cannot provide a universal measure, having to be accompanied by precision, we will also define a correlate of precision. Once we can measure both of these, we can judge the appropriateness of any given TM.

Instead of choosing a constant number of best candidate translations and measuring their fidelity one by one, we will need to measure the TM's view of space of translations generally, as a whole. In accordance with the categoricity of our measure of recall, also this measure of precision shall be categorical in comparing to reference translations. So, the property which can well distinguish TMs that have comparable rate of reachable sentences, is *certainty* in choosing the correct translation—TMs allowing virtually for any string to be the correct translation would have the rate of reachability insuperable, but for the price of never being certain.

Note that we used the phrase *correct translation* in the previous paragraph, despite that we had claimed that no translation can be considered the correct one. In some cases, a TM might be certain in choosing the reference translation, whereas it should not be that certain, according to the corpus evidenceThus, the measure of precision has to be readjusted to this fact. It has to consider the vector of probabilities that the corpus predicts for all possible translations (when factorized according to the source segments) and compare these probabilities to those proposed by the TM. The measure we are looking for is Kullback-Leibler divergence, defined as

$$\sum_x P_T(x) \log \frac{P_T(x)}{P_M(x)}, \tag{2.1}$$

where, in our case, x is a possible translation of the fixed source unit, P_T is its probability derived from data of the corpus ("true probability"), and P_M is its probability as proposed by the TM ("model probability").

The Kullback-Leibler divergence is measured in bits[1]. Commonly, it is transformed to another quantity, called *perplexity*:

$$2^{-\sum_x P_T(x) \log P_M(x)}. \tag{2.2}$$

(See [4].) Perplexity can be more natural to use, because its scale is linear rather than logarithmic, with regard to the intuitive property of dissimilarity.

[1] if one chooses 2 as the base for log

With the corpus fixed, KL-divergence and perplexity map to each other one-to-one, as the term $(\sum_x P_T(x) \log P_T(x))$ (the corpus entropy) in the following equation is constant:

$$\sum_x P_T(x) \log \frac{P_T(x)}{P_M(x)} = \sum_x P_T(x) \log P_T(x) - \sum_x P_T(x) \log(P_M(x)). \tag{2.3}$$

(The term $-\sum_x P_T(x) \log(P_M(x))$ is called *cross entropy*.) Therefore, as long as we are using the same corpus, we will use either of the measures, depending on which one yields more useful values in the particular situation.

N.B. Perplexity is directly dependent on the measure of cross entropy and vice versa. However, we are going to measure *perplexity of TM* compared to the corpus, and a measure of *entropy of decision* (see 2.3) which is independent on the perplexity of TM. In the following, we will always use the terms *TM perplexity* and *cross entropy* to refer to the mutual measure between the TM and the corpus, and the term *entropy of decision* to refer to the inner property of the TM.

Unfortunately, both the measures, KL-divergence and perplexity, have a feature which makes them unsuitable for our purpose without some adaptation—they are not defined for $P_M = 0$. Hence, we have either to restrict their use to only those translations which are allowable according to a given TM, or we need to adjust the measure to avoid computing $\log 0$.

While the first option would be much simpler and cleaner, it would mean discarding probably a large portion of translation pairs from the evaluation, particularly all such pairs where the corpus knows a translation (one or more) for their source segment that the TM does not know. Such pairs of segments would then neither be counted in the OOV measure, nor would they be counted in the measure of KL-divergence or perplexity. We decided not to permit such an inaccuracy.

Therefore, we devised an algorithm to measure perplexity also for those pairs of segments that are not measurable using the standard definition of perplexity.

2.2.1 Modified TM perplexity

We base the measure of perplexity on its standard definition, adjusting it in the special cases of $P_M = 0$. The adjustment aims at quantifying the information which the TM's best proposed candidate for the translation[2] lacks in comparison to the translation provided by the corpus. We also take into account any information the TM's best proposed candidate has surplus.

Formulated this way, the problem can be seen as measuring the edit distance, which is what we eventually do. We employ a weighted Levenshtein distance applied for whole basic language units (rather than letters). We define the operations

- insert a unit (***I***),
- delete a unit (***D***),

[2]Be it an empty string if the TM cannot propose any candidate at all.

viewed in the direction from the translation proposed by the TM to the translation in the corpus. We consider missing information a more several deficiency than including an excessive word (thus leaning slightly on side of the recall, in the precision–recall distinction). We derive the amount of information present in a language unit from its frequency in the corpus, as the amount of information needed to identify it among all units in the corpus. Inspired by the translation tool Moses [3] that we use for our experiments, we express the weight of each unit u_x as a log-probability:

$$w_{\boldsymbol{I}}(u_x) = \log\left(\frac{n(u_x)}{|C|}\right) - 1, \tag{2.4}$$

where $n(u)$ is number of occurrences of u in the corpus C and $|C|$ is total number of units in the corpus. Subtracting 1 approximates the amount of information missing in the corpus from the whole language in use, which the TM should ideally know, too. This weight applies to units which are missing from the reference translation, therefore it is indexed with the $\boldsymbol{I}$.

The cost for $\boldsymbol{D}$ can be approximated by a simple arithmetic mean of costs for all words in the reference translation:

$$w_{\boldsymbol{D}}(u_x) = \frac{1}{\text{tgtLength}} \sum_{u \in \text{tgt}} \left(\frac{n(u)}{|C|}\right), \tag{2.5}$$

where "tgt" is the reference translation, u is a unit in that translation and "tgtLength" length of "tgt" measured in the language units used.

We expect that costs for $\boldsymbol{I}$ will be higher, as TMs will probably leave out rather rare words, which are more costly—more than an average word in the reference translation. The correction for corpus incompleteness is another measure to put more emphasis on the recall.

Now, having set up for measuring the probability of translations not reachable by the TM using weighted Levenshtein distance, we simply supply these probabilities in the formulae 2.1 and 2.2 for P_M.

2.3 Entropy of decision

Finally, having accounted for ability to reach good enough translations by reachability and close-fitting to structure of natural languages by TM perplexity, we want to define also some measures to quantify uncertainty associated with performing the translation using the chosen TM. To express it using the conception of analysis and synthesis as relations, we want to measure expansiveness of these relations, weighted by probability of transitions over their different edges.

What we are going to employ for this aim, is *entropy of decision*, as a compound measure for a set of edges weighted with probabilities. Let us define the entropy of decision as

$$-\sum_{f_i} P(e_i|f_i) \log P(e_i|f_i) \tag{2.6}$$

for each language unit f_i from the source segment, where e_i is its possible image (the target language unit this translates into) and $P(e_i)$ the probability from view of the TM that the source language unit should be translated into e_i.

The entropy of decision corresponds to the uncertainty of the TM during the translation. It should not be confused with TM perplexity, defined above, which describes rather the relation of the TM and the corpus.

Because we will want to compare TMs working with different kinds of language units, we are going to count the entropy of decision for every input unit and sum them up to yield *total entropy of decision* for the whole segment f:

$$-\sum_{f_i \in f} \sum_{e_i} P(e_i|f_i) \log P(e_i|f_i). \tag{2.7}$$

This approach neglects all other uncertainty associated with the decoding, such as employing language models, which assign different probabilities to different orderings of the output units, as well as lexical weigthing, honored by the Moses decoder, or any feature functions. We have chosen to measure TMs by their very acquisition to the whole translation pipeline, rather than by their ability to perform well with other components co-employed in the translation process.

Regarding the phases of analysis and synthesis, it should be noted yet that we have decided to work with lowercased forms of segments from the corpus, thus always staying at least at this level of abstraction. We believe that it enables to compare different TMs more accurately, as differences stemming from different errors in truecasing are rather caused by differring performance of truecasers than of the TMs.

2.4 Summary

To make the final set of measures more synoptic, we recapitulate them here, in the table 2.1, together with abbreviations we are going to use for them hereinafter. The measured values are cited in chapters about the respective TMs either in tables, or as plotted data, apart from IV-n, which has its own chapter.

Table 2.1: Summary of measures

Name	Abbreviation	Description
Reachability vs. TM perplexity		
Reachability	rbl	Reachability of the reference translation by the TM. Analogous to recall.
Modified TM perplexity	TM-ppl	Measures differences in probability spaces of translations between the TM and the corpus. Analogous to precision.
TM cross entropy	TM-xent	Binary logarithm of the modified TM perplexity.
Level of abstraction		
Size of the lexicon	|Lex|$[B]$ / |Lex|[phr]	Size of the translation lexicon, either absolute size in bytes, or number of phrases it contains.
OOV of the TM	OOV	Rate of source segments from which not all units occur in the TM's vocabulary.
Empirical IV-n	IV-n	Number of instances of units in the source side of the testing data that occur in the training data n times. The difference between IV-0 and OOV is, that IV-0 is counted directly from the training data, whereas OOV depends on the TM's extracted vocabulary.
Uncertainty		
Entropy of decision	Ent	Entropy of decision when using the TM. Also measured on a per-unit base.

Chapter 3

Extreme translation models

3.1 Translating sentence-wise (Sen)

As mentioned above, we have started experimenting with a trivial TM, which we developed for that single purpose. The TM we measure merely collects pairs of whole sentences it finds in training data. It is able to translate only exactly the same phrase it saw in training data, provided it saw also the right translation with it there. The only abstraction performed is lowercasing.

3.1.1 Characteristics of the sentence-wise approach

The language unit this model uses is a whole sentence[1]. We have chosen it, as it is the only language unit readily provided with the corpus we have, whose instances in source and target languages are already mapped to each other in the one-to-one fashion. This model completely ignores any linguistical structures behind given sentences of characters (or, bytes), not to mention the meaning.

3.1.2 Technical setup of the experiments

For comparing sentences from the training part of a corpus with sentences from its testing part and counting number of those which are same, not much is needed over the corpus itself. Rather than an actual piece of software, fractions of the corpus of different size served as our TM, with several GNU utilities and a database management system on top of them.

We ran our experiments on the corpus CzEng 0.9. In accordance with instructions from authors of CzEng, we used first 8 sections of total 10 for training. We did not use the 9^{th} section, reserved for testing during development, at all. And we used the 10^{th} section for evaluation of the model. Most data we utilized come from the plaintext format of the corpus, only counts of tokens are derived from its "export format".

[1] . . .or any segment, depending on what the corpus contains

tr[sen.]	rbl[sen.]	rbl (sen.)
1,000	10,338	1.2 %
3,162	22,680	2.8 %
10,000	33,751	4.2 %
31,623	48,267	6.0 %
100,000	68,359	**8.5 %**
316,228	93,874	11.6 %
1,000,000	131,037	16.3 %
3,162,278	187,881	**23.4 %**

Table 3.1: Reachability for the model Sen. The column headers mean: 'tr' = "size of training data"; 'rbl' = "reachability"; 'sen' = "sentences/segments".

tr[sen.]	tr[en. t.]	tr[cz. t.]	rbl (en. t.)	rbl (cz. t.)
1,000	11,376	9,923	0.5 %	0.5 %
3,162	35,460	31,111	0.9 %	0.9 %
10,000	115,663	101,775	1.4 %	1.3 %
31,623	365,211	321,871	2.1 %	2.0 %
100,000	1,153,029	1,019,940	3.2 %	3.2 %
316,228	3,672,385	3,247,100	5.0 %	5.0 %
1,000,000	11,598,831	10,253,485	8.1 %	8.3 %
3,162,278	36,700,477	32,444,389	13.7 %	14.0 %

Table 3.2: Reachability for the model Sen, each segment weighted by number of words it contains. The new column headers mean: 'en. t.' = "English tokens"; 'cz. t.' = "Czech tokens".

3.1.3 Expectations

As outlined earlier, we expect this model to be an extreme one. It should be most error-free of all TMs (provided they cannot take advantage from context). Therefore, it will probably have the highest score in choosing the right translation, and the same for entropy of decision, given the testing part of the corpus is consistent enough with its training part. On the other hand, without the TM making any abstractions, we expect it to be the most specific one, with a very high OOV rate.

The main purpose of the model is to provide us with extremal values of the quantities we will measure. If they turn out to be less extreme than we expect, we will be able to relativize values obtained by measuring other models.

3.1.4 Results

The results from our experiments with translating sentence-wise are captured in the tables 3.1 and 3.2.

After having performed a few first measurements, we were quite surprised with the reachability of translations. Already with as few as ten thousand pairs of segments in training data, roughly

Table 3.3: Size of the lexicon for the model Sen. Training data are expressed in number of segment pairs, size of the lexicon is measured in kilobytes.

training data	|Lex|[kB]
1,000	221
3,162	675
10,000	2,137
31,623	6,671
100,000	20,913
316,228	65,803
1,000,000	206,546
3,162,278	648,640

8.5 % of reference translations become reachable[2] (see Table 3.1). And from all the testing data, over 23 % of the pairs can be seen literally in the training data. This called for explanation.

The reason for such high reachability can be either poor correspondence of probability distribution of sentences in text, as implied by the corpus, with the true distribution (which, we believe, is not recurring to that extent). Or, there might be short segments strongly represented in the corpus, which are more likely to repeat literally than long sentences. To find the actual reason, we measured the reachability one more time, counting number of reachable tokens in the sentences, rather that just count of the sentences. To figure out number of tokens in each segment, we utilized the "export format" part of Czeng, which already comes tokenized.

Results of measurements of the reachability in number of tokens can be seen in the last two columns of the second part of the table 3.2. By comparing this table with the table 3.1, we can see that percentages in the table for tokens are approximately half the corresponding values in the first table. This verifies the expectation that shorter sentences are reachable more often than longer ones.

Despite this positive findings, we peeked into the data to see what kind of segments are repetitive, while not too short. We have found that there are many paragraphs mainly from the EU law, which are obviously each time literally the same, coming from different sources. We draw a conclusion from it, that the corpus is unnaturally regular, fairly non-random.

Regarding the complement measure to reachability, as declared in the section 2.2, either the TM perplexity or KL-divergence, we were not able to calculate it in reasonable time using the basic tools we used for this model. This is unfortunate, as if we knew results for this measure, we could find out about diversity of the corpus—whether it contains rather many different target segments for each source segment, or just a few of them.

The entropy of decision for this TM is summarized in fig. 3.1. The result reflects our prior expectations, that is that the entropy of decision is very low. Note that even after the TM sees 10^6 segment pairs in the training corpus, its decision is approximately equally uncertain as a coin flip.

[2]Note that reachability was evaluated only once for both directions in this TM. This is because whenever the reference translation is reachable in one direction, it had to be present literally in the training data, and therefore it is reachable also in the opposite direction.

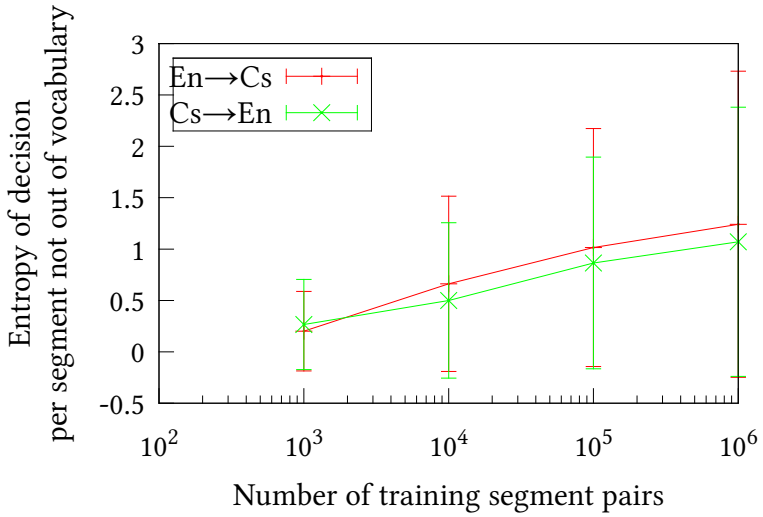
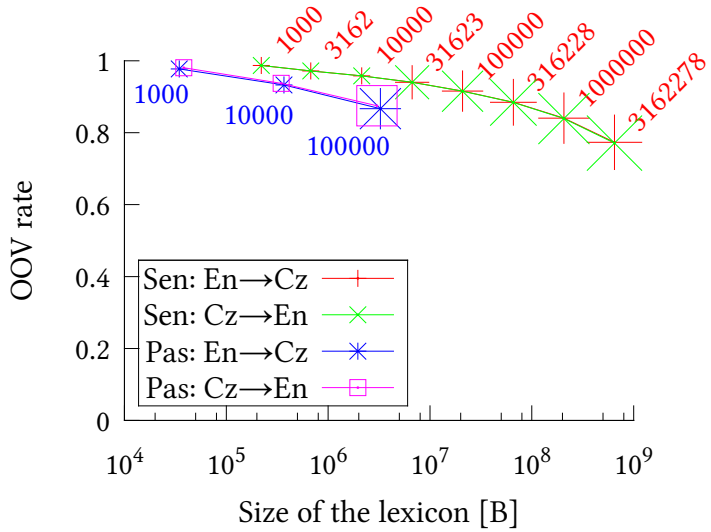


Figure 3.1: Entropy of decision for the TM Sen. The middle tick shows the average value, the bars delimit the standard deviation of the measured sample. It is not an error that they reach into negative values; they merely visualize an approximation of the standard deviation of the real random measure.

Figure 3.2: Comparison of the models Sen and Pas: Size of the lexicon vs. OOV rate, roughly corresponding to level of abstraction vs. applicability of the abstraction. The labels next to the points express number of pairs of segments in training data.



3.2 Translating sentence-wise using a statistical model (Pas)

To ensure coherence of the results we obtain from different approaches, we measured the sentence-wise TM once again, using a phrase-based statistical TM. We pasted words in each segment in one long word by substituting the character '@' for each space. This way, we simulated the sentence-wise translation for a phrase-based TM.

We expected to get the same results as with the other method. However, the results from our hypothetical TM from the previous section did not happen to be absolutely identical to those we obtained using a phrase-based TM over pasted segments. We do not print the tables for the measures both models are almost the same in again. The interesting differences are captured in the figure 3.2. You can note that the phrase based TM (Pas) somehow performs better than the exact applying pairs seen in training data literally. Not only does it save the information more effectively—which one could expect—, but it can reach slightly more reference translations given the same training data. By comparing the sets of translations reachable by the model Sen and by Pas, we found that the Moses tool substitutes the original segment as the translation where it does not know any better. Surprisingly, copying the source segment proved to be the right answer in a number of cases.

If not with respect to size of the lexicon, at least with respect to all other relevant parameters, Pas behaves almost the same as Sen. Based on this presumption, we have measured the appropriateness (TM perplexity or cross entropy) of the TM for Pas only, not for Sen. The results are captured in the figure 3.3. We can take these values as an example of very good ones, because this TM is

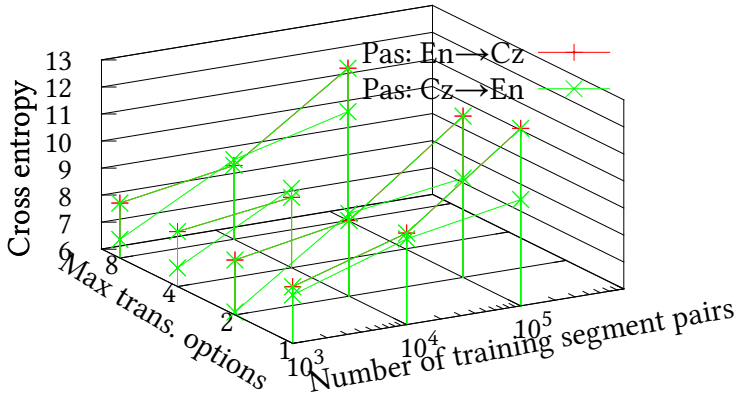


Figure 3.3: Average cross entropy of the model Pas for different decoding options. Note that setting the maximum translation options per input span does not improve the TM's performance almost at all. That indicates that the TM does not know too many good translations for any of source phrases.

usually maximally sure with the translation, as possible, unless the source segment was out of its vocabulary. Recall, though, that source segments out of vocabulary are not evaluated with respect to cross entropy, TM perplexity or KL-divergence. It is also mostly the out-of-vocabulary segments' credit that with less training data, the cross entropy is lower. In this context, importance of the OOV value over cross entropy is prominent.

What is worth noting at figures 3.3 and 3.4, is that the cross entropy tends to get rather worse with increasing size of training data, than better. It may be caused by growing vocabulary of the TM, which makes its OOV score lower (better) for the price of reducing its overall appropriateness for cases where it is applicable. Another point about the results is that given a Czech segment, having more morphology, being more informative, on input makes the decision easier.

3.3 Translating letterwise (Ltr)

The second extreme we examine to set boundary values for our measures, is choosing single letter for the basic language unit. Because letters are not aligned one-to-one in the corpus, we already need to employ a statistical model which makes up (non-trivial) alignments for language units in the training segments. Therefore, we use a phrase-based TM adjusted for phrases build up from single-letter words. We include a special word for the space character, too, as we consider it important to be able to transfer boundaries of words correctly.

With scaling down to such miniature unit as a letter, the computational costs of the translation dramatically increase. Therefore, range of both training and testing data need to get reduced accordingly (we scaled down from tens of thousands and millions of segments to a few dozens).

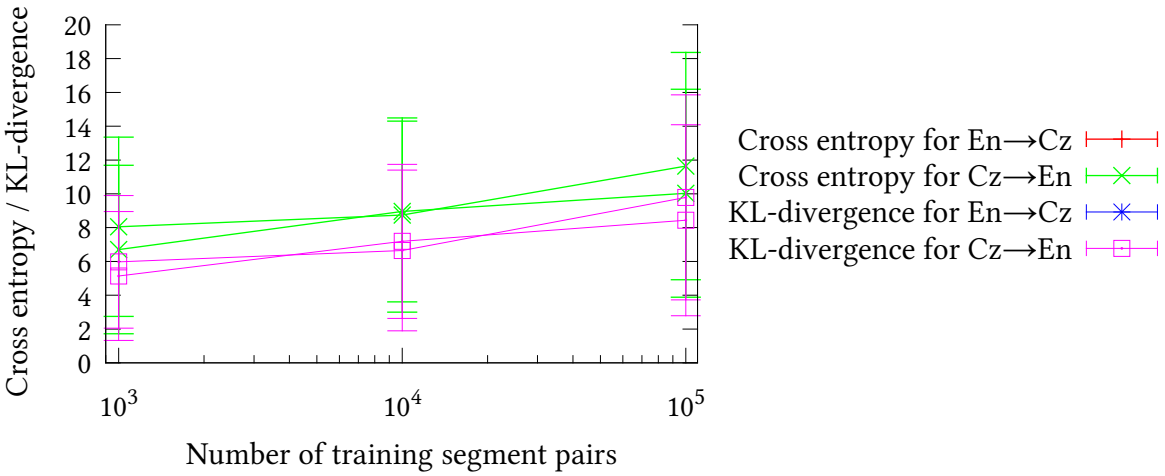


Figure 3.4: Closer look at the Pas TM's precision results after setting the maximum translation options per input span to 8 and distortion limit to 0 (which should allow for pretty good results). The vertical bars show the standard deviation of measured values for testing data. The large standard deviation shows that some of the translations proposed by the TM are in perfect agreement with the corpus evidence, while others are rather bad, reaching about the level of the top mark. (Caused by a bug, which could not be recovered at the time, the lines for both translation directions are unfortunately drawn in the same style.)

3.3.1 Expectations

We expect this TM to set the upper bound for perplexity, as well as the lower bound for size of the lexicon. In other words, this TM is expected to abstract highly from provided information while not being able to use the knowledge efficiently. It will also probably have an extremely high entropy of decision.

3.3.2 Results

The results for this TM were obtained in two runs, which differred in setting of the maximum translation options per input span. In the first run, the results were generated with this parameter set to 4 or 8. They brought a surprise that not a single reference translation was reachable with them. This was already the first sign of a bad choice of the language unit, showing that the units from the two languages cannot be well aligned, that their correspondence is in almost no relation to the correspondence of meanings.

However, after finding so, we made the second run of translations, with the maximum translation options limit set high enough to cover most of all common characters, namely to 50. This adjustment had the expected effect—some of the reference translations became reachable. The reachability with these second-run TMs is summarized in the table 3.4.

One hundred of training sentences might sound like too little, but when all of them are cut into as many pieces, as number of letters they are composed of, it presents a lot of information for the

tr[sen]	dir	rbl[sen]	rbl
100	cs⟶en	21,966	2.7 %
100	en⟶cs	11,187	1.4 %

Table 3.4: Reachability for the translation model Ltr used with <maximum translation options per span> set to 50. The columns contain size of the training data (in sentences/segments), direction of the translation, number of reachable sentences (out of 802,392), and the reachability rate.

tr[sen]	ttl	dl	dir	Cross entropy [b]		
				avg±stdev	min	max
10	4	4	cs⟶en	51.6±32.6	*0*	120.3
10	4	4	en⟶cs	50.6±24.3	10	104.8
10	8	0	cs⟶en	52.8±33.4	*0*	122.8
10	8	0	en⟶cs	51±21.5	2.1	91.8
18	8	0	cs⟶en	53.4±34.8	*0*	129.3
18	8	0	en⟶cs	72±44.6	7.2	185.4
32	4	4	en⟶cs	81±48.6	7.2	198
32	8	0	cs⟶en	54.5±34.1	*0*	134.6
32	8	0	en⟶cs	79.7±49.2	5.7	200.2
100	50	4	cs⟶en	38.5±32.6	*0*	300.7
100	50	4	en⟶cs	27.8±34.5	3.6	327

Table 3.5: Cross entropy of the TM Ltr. The column headers mean: 'tr[sen]' = "number of training sentences/segments; 'ttl' = "maximum number of translation options per input span"; 'dl' = "limit on distortion—reordering—of the translation".

MT system. Despite that, the reachable number of sentences is very low. Our prior expectation was that almost every translation is reachable, but it turns out that complexity of finding the right translation among all possible is too high.

If the translation using letterwise approach was not as tardy as it was, we would perform more sounds into the space of possible experiment setups, to find more about trends the translation follows. Because this was not possible, we can only study a few other values we obtained, that describe the other properties.

The table 3.5 lists measured values regarding perplexity of this TM, to complete the picture of poor reachability of reference. It demonstrates high inappropriateness of a TM to the translation problem. The highest values of cross entropy of the TM and the corpus reach 300 bits—that is the amount of information that would have to be added to gain the one, certain translation for the given source sentence.

Apart from the absolute values of the cross entropy, one more thing is astounding about the results. The zeros as minimal values for the cross entropy should mean that some source Czech segment was translatable with no doubts into one and only one English segment. That is not virtually possible with this TM. Moreover, the zero cross entropy got measured with a pretty long sentence. We realize that this must have been caused by a bug in the software, but we, unfortunately, have

tr[sen]	ttl	dl	dir	|lex|[kB]	oov
10	4	4	cs⟶en	12	2.6 %
10	4	4	en⟶cs	13	4.2 %
10	8	0	cs⟶en	12	2.6 %
10	8	0	en⟶cs	13	4.2 %
18	8	0	cs⟶en	31	2.8 %
18	8	0	en⟶cs	40	2.6 %
32	4	4	en⟶cs	74	1.5 %
32	8	0	cs⟶en	68	1.8 %
32	8	0	en⟶cs	74	1.5 %
100	50	0	cs⟶en	288	0.4 %
100	50	0	en⟶cs	305	0.2 %

Table 3.6: OOV rate for the translation model Ltr. The columns contain size of the training data (in sentences/segments), maximum translation options per input span, distortion limit, direction of the translation, and the OOV rate.

not found the cause.

On the other hand, the results from measurements of OOV turned out just as expected. The OOV was at low 2.6 % already from 10 segments of training data, and after first 100 segments, only each 500^{th} unit—character—was unknown to the TM. These values were reached by the TM with using mere 300 kB. The best OOV score of Pas was reached much earlier by Ltr, with less than 12 kB of memory for storing the translation lexicon. All the results are listed in Table 3.6.

The table 3.7 summarizes measurements of the decision entropy for the TM Ltr. The first four columns are the same as in the previous tables, for specifying the translation settings. Following columns list the new information. From those, of a special interest is the third last column, describing decision entropy per unit. When the unit is a character/letter, as in this case, if we neglect the TM's ability to translate the letters from the source segment's letters, this value corresponds to the entropy rate of an English, or Czech text, which is known to have approximately these values. (See [5].)

tr[sen]	ttl	dl	dir	entropy			entropy per unit		
				avg±stddev	min	max	avg±stddev	min	max
10	4	4	cs⟶en	39.3±32.3	4.5	180.9	1.2±0.6	0	2.9
10	4	4	en⟶cs	34.4±25.9	7.6	135.8	1.2±0.8	0	3.6
10	8	0	cs⟶en	42.4±36.2	5.1	199.6	1.3±0.7	0	2.9
10	8	0	en⟶cs	37.9±30.4	7.0	161.1	1.3±0.9	0	4.1
18	8	0	cs⟶en	46.4±40.6	4.2	216.7	1.4±0.8	0	3.2
18	8	0	en⟶cs	58.9±48.8	12.8	266.8	1.5±0.9	0	3.6
32	4	4	en⟶cs	60.0±46.1	15.0	272.7	1.6±0.8	0	3.5
32	8	0	cs⟶en	47.3±37.1	4.5	207.1	1.5±0.9	0	3.2
32	8	0	en⟶cs	65.4±51.5	15.7	309.2	1.7±0.9	0	4.0
100	50	0	cs⟶en	90.7±68.4	7.3	475.2	2.3±1.3	2.4	0
100	50	0	en⟶cs	102.4±74.7	16.6	384.2	2.3±1.3	0	5.3

Table 3.7: The entropy and entropy per unit for the TM Ltr.

Chapter 4

Phrase translation models (Phr)

Having finished setting appropriate measures and checking them at extreme TMs, we advanced to more real examples of TMs and language units they use. However, these TMs consumed still more resources to produce the output than, for example, models for translating letterwise. Therefore, both training and testing data were smaller for the experiments that employed the TM.

4.1 Setup of the experiments

We used sets of 1,000 and 10,000 segment pairs of training data for training phrase translation models with the maximal length of phrases saved into the translation lexicon restricted to 1, 3, and 10 words, in subsequent experiments. After training the models, they were tuned using MERT (minimum error rate training) on equally sized tuning data. For obtaining the translations, we used the Moses decoder again. The number of segments eventually translated by the models varies from about 25,000 to 800,000, as translating the whole testing section of CzEng took a few days of CPU, and thus proved intolerable. (Full listing of numbers of translated segments for all TMs can be found in Chapter A.1 in the appendix.) Fortunately, the reachability was much quicker to calculate, so it is always evaluated on the whole testing section.

4.2 Results

All the experiments were run using the setting TTL (maximum translation options per input span) = DL (distortion limit) = 4. The maximum phrase length (abbreviated as 'plen' below) used for the translation was varied among values 1, 3, and 10. We expected the proposed measures to respond to the changing 'plen' parameter.

The measured results are listed in the table 4.1.

tr[sen]	1,000	1,000	1,000	1,000	10,000	10,000	10,000	10,000	10,000
dir	en→cs	en→cs	cs→en	en→cs	cs→en	en→cs	cs→en	en→cs	en→cs
plen	1	3	3	10	1	1	3	3	10
lex[kB]	41	246	243	896	286	280	2,577	2,583	10,873
lex[#]	1,922	9,837	9,742	29,730	13,116	12,986	100,579	100,345	364,317
oov	32.6 %	30.9 %	43.6 %	30.8 %	25.5 %	14.2 %	22.8 %	13.0 %	12.9 %
rbl[1]	2.3 %	3.2 %	3.3 %	3.3 %	6.6 %	5.8 %	9.3 %	9.1 %	9.8 %
Entropy of decision									
avg	3.2	5.5	3.8	6.5	2.7	6.7	6.1	10.4	11.1
stddev	2.7	4.4	4.1	7.9	2.3	5.7	5.3	8.0	10.1
min	0	0	0	0	0	0	0	0	0
max	21.3	37.3	28.6	*116.6*	47.7	59.2	72.3	77.0	192.5
Entropy of decision per unit									
avg	0.8	1.2	1.1	1.4	0.7	1.1	1.3	1.6	1.7
stddev	0.7	0.8	0.8	1.0	0.6	0.7	0.8	0.8	0.9
min	0	0	0	0	0	0	0	0	0
max	2.0	3.5	3.4	5.1	2.0	2.0	4.1	4.0	5.7
Cross entropy									
avg	80.9	64.0	11.4	24.9	23.2	13.2	24.4	8.2	28.9
stddev	71.3	63.8	9.9	16.9	17.6	19.4	20.1	16.3	44.0
min	2.8	1.6	0.7	2.1	3.5	0.1	3.3	0.0	0.3
max	909.3	902.5	102.4	181.1	226.0	247.2	214.2	229.4	488.0

Table 4.1: Results from the measurement of the phrase TMs. The first column describes meaning of the corresponding row. The abbreviations read this way: 'tr[sen]' = "number of segments/sentences in the training data"; 'dir' = direction of the translation; 'plen' = "maximum length of phrases saved into the translation lexicon"; 'lex[kB] = size of the translation lexicons in kilobytes; 'lex[#]' = size of the translation lexicons in number of items; 'oov' = "out-of-vocabulary rate"; 'rbl rate' = "rate of reachable reference translations"; 'avg' = "arithmetic mean"; 'stddev' = "standard deviation"; 'min' = "minimum value"; 'max' = "maximum value".

4.3 Analysis of the results

Let us first describe constingness of this TM, as implied by the table of results, and how it depends on the translation settings. By the constingness, we understand amount of resources needed by the translation, which is reflected prevalently by the size of the translation lexicon (for memory requirements) and partly by the entropy of decision (for its impact on CPU and other dynamic resources).

When we experimented with the TMs Pas and Sen, which store whole segment pairs in the translation lexicon, we made the prediction that they would use up most space for the lexicon. However, the results from experiments with Phr contradict the prediction. When training on 10,000 sentences, Phr uses up to 11 MB of memory for the translation lexicon, whereas Sen needed only 2 MB. It is obviously caused by Phr learning overlapping phrases from the training data, which in turn causes it to remember each word up to 10 times, in the case of phrases 10 words long. However, we believe that with training data growing further beyond the size we experimented with, there occur unseen whole segments much more often then unseen phrases of a fixed maximum length. Hence, we predict that the Sen's translation lexicon would overgrow that of Phr at some time. For the maximum phrase lengths 1 and 3, sizes of the translation lexicons are nearing those for Sen already in the setting with 1,000, resp. 10,000 training sentences.

Regarding the entropy of decision, it stays very low when measured per unit, but it also stays relatively low when evaluated for the whole segments. Its variation is too big to rule out possible discrepancy between the measured average and its actual mean value, but with this many translated data (as listed in Table A.1) the average values become fairly reliable. Note, though, that this parameter can be directly influenced by setting parameters to the decoder. Therefore, its plausibly low value should be understood rather as reflecting setting the decoder to be plausible, than as a surprising result of this TM. Regardless of this fact, the differences in the entropy between the two directions of translation are noteworthy. The TM provides about 3 b or 4 b more of decision entropy, on average, for the 1,000 training sentences, respectively 10,000 training sentences settings. That means that this TM sees Czech as a richer language than English (having learned it from a few thousand sentences only). It is certainly caused partly by the rich Czech morphology, but probably to the same extent also by the Czech free word order which accounts for the TMs uncertainty in aligning corresponding phrases correctly.

The second group of properties concerns the success of the TM in finding a good translation. First important measure to look at is the OOV rate. Also here we can note a significant difference between the directions of translation. Czech proved more complex again, as it did regarding the entropy of decision. It had the OOV rate almost twice as big as English in the experiments with 10,000 training sentences.

In contrast to the OOV is reachability of reference translations, which was always slightly higher for translating to English from Czech than otherwise, in the pairs of experiments with the same settings. This also testifies for Czech being a more complex language in the view of this TM. Regarding the absolute values of these measures, they are rather poor. Not more than 10 % of reference translations were reachable, even using as regular data source as we found CzEng is (in the section 3.1.4). However, compared to the extreme models of Sen or Pas, which had reachability of 4.2 % with 10,000 training sentences, the phrase based TM performed much better. We can observe a sig-

nificant improvement between experiments that used 1,000 training sentences and those that used 10,000 training sentences. The improvement is most notable with maximum phrase length set to 3.

Unlike the training data size, the maximum phrase length is not that essential for reaching better results. But it proves very important. As we have mentioned in the above text, adding to the phrase length does not damage the entropy of decision noticeably. Or, the other way around, even when keeping the entropy of decision low, and thus saving dynamic computational resources, we can use longer phrases and get more good translations. The exact measured improvement can be best tracked by the experiments that we conducted with all maximum phrase length settings (i.e. 1, 3, and 10). From these series of experiments, we get the results $\langle 2.3\ \%, 3.2\ \%, 3.3\ \% \rangle$, and $\langle 5.8\ \%, 9.1\ \%, 9.8\ \% \rangle$, for 1,000 training sentences, and 10,000 training sentences, respectively. These sequences lead us to expect the ideal setting somewhere over phrases of length up to 10, although we cannot tell how the relation will develop further. We expect longer phrases to impose a burden at both the training and decoding phase, which might increase the complexity of the translation, as well as consume part of space of partial translation hypotheses, that should be better occupied by hypotheses made up of shorter phrases. This is a matter for future measurements. The possibility to success in translating as a parameter of decoder settings is well examined in [1].

As the last measured parameter, we will analyze the cross entropy. At first sight, we can notice it varies largely—both with regards to the average in different experiment settings, and with regards to variation in each particular setting. The second mentioned phenomenon can only be explained by the TM having come across source sentences of largely variable demandingness, some of which were well learned by the TM, but some of which were rather alien. The variation across different experiment settings is of more interest. In the 1,000 training sentences setting, the only translation from Czech to English outperforms all translations in the other directions sovereignly. This is apparently due to its high OOV score which causes many source sentences not to be measured with respect to OOV at all. The same cause can be seen behind differences of translating from Czech to English and in the other direction with the setting of 10,000 training sentences and the maximum phrase length of 1 or 3 words.

Specially surprising is the value measured for the last experiment, translation based on 10,000 training sentences from English to Czech, with maximum phrase length 10. Its cross entropy got much higher than those of experiments with shorter phrases, while the OOV rate stays almost the same. This was most probably caused by pruning translation paths to the actual best translation candidates (by *search errors*, as described in [1]). N.B. that cross entropy, unlike reachability, was evaluated using unconstrained translation, while reachability, in which this setting reached a high rating, was measured using a constrained translation. The relatively high score in reachability, and relatively low score in the cross entropy at the same time means that the model allowed for translating the sentences rightly, but did not score individual translations well (not in good accordance to the corpus evidence).

Chapter 5

TMs using additional linguistic information (Afact and Tfact)

This section is devoted to factored translation, i.e. translating words annotated with various tags. We performed two kinds of factored translation—one using annotations at the morphological level (or a-level), and the second one using annotations at the tectogrammatical level (or t-level). All these experiments were again conducted using the MT system Moses. The decoder parameters were set the same as in experiments with Phr, and the maximum phrase length was varied, as in experiments with Phr. The measures were evaluated on a part of the 10^{th} section of CzEng, size of which is listed in the table A.1, but reachability was evaluated using the whole section.

5.1 Setup of translations at the a-level

For translating with morphological information available, we picked a translation scheme that was linguistically appealing: we translated separately a-lemmata and morphological tags, employing a language model at the target side of both of these layers, and from the target-side lemmata and morphology, we generated back the surface forms.

5.2 Results for Afact

The measured results are listed in the tables 5.1 and 5.3.

5.3 Analysis of the results

Let us first describe the constingness of the TMs again. We can notice that the translation lexicon was somewhat smaller for the factored translation (Afact) than for the unfactored (Phr). For instance, approximately 1.4 MB for Afact (10,000 training sen., max. phrase length 3, both directions) compares to approximately 2.6 MB for Phr with the same settings. On the other hand, the lexicon of Afact holds more entries than that of Phr with the same settings of the experiment. For

tr[sen]	1,000	1,000	1,000	1,000	1,000	1,000
dir	en→cs	cs→en	en→cs	cs→en	en→cs	cs→en
plen	1	1	3	3	10	10
lex[kB]	55	39	174	157	564	539
lex[#]	2,539	2,561	16,421	16,222	53,319	53,435
oov	31.0 %	42.1 %	29.9 %	40.4 %	29.7 %	40.1 %
rbl	2.2 %	2.7 %	3.0 %	2.2 %	3.5 %	3.3 %

Entropy of decision						
avg	5.2	3.0	7.2	5.2	7.9	5.8
stddev	3.8	2.3	5.6	3.9	8.2	7.8
min	0	0	0	0	0	0
max	35.6	25.9	43.8	55.0	*85.1*	72.4

Entropy of decision per unit						
avg	1.1	0.9	1.5	1.4	1.6	1.5
stddev	0.7	0.7	0.9	0.8	1.0	1.1
min	0	0	0	0	0	0
max	2.0	2.0	4.1	3.6	5.3	5.0

Cross entropy						
avg	60.2	15.7	10.4	28.5	14.2	19.1
stddev	65.1	14.4	12.4	21.7	14.1	19.6
min	0.4	0.5	0.1	0.8	0.4	0.7
max	573.2	223.8	137.9	187.9	148.6	169.0

Table 5.1: Results for the TM Afact trained on 1,000 segment pairs.

the settings used for the previous example, the lexicons contain about 150,000, and 100,000 entries, respectively. [2] That reflects that the factored model sees less different language units in the training data (therefore the translation table is smaller), but stores them more times each (into the three tables: translation, reordering, and generation); therefore the lexicons contain more entries than they do for non-factored models.

We would like to make an estimate of the translation lexicon size for growing training data, but, unfortunately, we have not had enough resources and time to run experiments on more different sizes of training data, thus we have too little data to make the estimate. However, we expect that with the abstraction Afact involves, the translation lexicons would grow slower depending on the size of the training data, than those of Phr. So far, we can merely compare the rate of growth of the lexicons depending on the maximum phrase length allowed. In such a comparison, Phr shows slightly slower growth. In other words, there are less individual lemmata and annalytical tags seen

[2] The entries counted include those in the translation table only (not the entries from either reordering table or from the generation table).

tr[sen]	10,000	10,000	10,000	10,000[1]	10,000	10,000	1,000,000	1,000,000
dir	en→cs	cs→en	en→cs	cs→en	en→cs	cs→en	en→cs	cs→en
plen	1	1	3	3	10	10	1	1
lex[kB]	335	197	1,448	1,309	6,339	6,170	6,064	3,068
lex[#]	12,630	12,713	151,775	152,636	639,224	646,442	380,592	378,356
oov	12.5 %	18.2 %	11.6 %	16.9 %	11.5 %	16.9 %	1.8 %	1.8 %
rbl	4.8 %	6.1 %	8.0 %	4.1 %	8.8 %	6.6 %	6.9 %	–
Entropy of decision								
avg	9.6	6.2	14.2	8.5	15.0	9.9	15.9	14.7
stddev	8.0	4.8	11.1	6.3	12.4	9.2	16.1	14.3
min	0	0	0	0	0	0	0	0
max	107.2	91.6	162.9	53.4	199.9	99.4	126.1	135.9
Entropy of decision per unit								
avg	1.5	1.2	2.1	1.7	2.2	1.8	1.5	1.6
stddev	0.6	0.6	0.8	0.8	0.9	1.0	0.5	0.6
min	0	0	0	0	0	0	0	0
max	2.0	2.0	4.3	3.9	5.6	6.5	2.0	2.0
Cross entropy								
avg	28.4	22.8	17.9	9.2	15.3	31.0	23.8	–
stddev	31.3	22.8	25.6	16.0	24.1	55.0	42.4	–
min	0.4	0.5	0.2	0.1	0.2	0.2	0.0	–
max	425.1	287.8	335.5	206.6	303.6	750.0	486.4	–

Table 5.3: Results for the TM Afact trained on 10,000 and 1,000,000 segment pairs. Fields marked with "–" correspond to experiments we have not measured, particularly for their high computational requirements.

in the training data, but they can be combined in more ways into 3-grams and 10-grams, than the whole tokens from the surface level.

Regarding the entropy of decision, we observed a moderate increase compared to Phr. However, as a pleasing trifle, Afact lowered the maximum value of the entropy of decision in the setting 1,000 tr. sen., en→cs, plen 10, from Phr's 116.6 b to 72.4 b, even getting better OOV rate. Otherwise, the entropy of decision is a bit higher also on a per unit base, where it reaches over 2 b (on average).

Let us now analyze the Afact's results with regard to its success. In the first measure, OOV rate, we can see an improvement compared to Phr. The improvement is not very clear with 1,000 training segments, while with 10,000 training segments it becomes notable. To instantiate this difference, compare the experiments using 10,000 training segments and maximum phrase length of 3 tokens. Phr has OOV rate for these settings $\langle 13.0\ \%, 22.8\ \% \rangle$ for the directions en→cs, and cs→en,

respectively, while Afacts reaches better $\langle 11.6\ \%, 16.9\ \% \rangle$ for the same experiments. The differences in OOV rate are most notable for the direction cs⟶en, that is for covering the Czech vocabulary. This exactly agrees with the characteristics of Czech as of a language with rich morphology, which factorization of the model prevalently copes with. As the best result from experimenting with Afact we can see pushing the OOV rate as low as to 1.8 % with models trained on 1 million segment pairs. (The phrase based TM would surely also reach a lower score if trained on more data, but as the experiments with 10,000 training segments indicate, it would most probably not reaach this low OOV rate.)

In contrast to improvement of the OOV rate, the reachability decreased about 1 % on average compared to Phr. The difference was again almost unobservable with 1,000 training segments, where Afact even reached a better score for translating en⟶cs with phrases up to 10 tokens long (3.5 %, compared to 3.3 %). The reachability is again increasing as we loose the maximum phrase length constraint. A highly interesting phenomenon that can be seen in the results from experiments with max. phrase length 3 and higher, is that the relation of reachability for English and Czech got inverse, compared to all preceding results. When the reachability got lower overall, it should be perceived rather as getting the cs⟶en translation worse than getting the en⟶cs translation better. Without further research, we cannot tell surely what caused this turnover. We would expect the Czech segments to be harder to generate (or that there would be a larger 'vocabulary' of them), which the results disprove. We can only draw the conclusion that the factored model coped better with the richer (or, maybe, more thoroughly annotated) language than with the second one.

A very surprising result is the fall in reachability of cs⟶en translation using 10,000 training segment pairs, from maximum phrase length of 1 to 3. The same fall is present also in the parallel experiments based on 1,000 training segment pairs. The only explanation we can supply is the translation lexicon being probably populated with bigrams and trigrams that blocked unigrams, otherwise useful for the reachability, from getting either into the lexicon or into consideration by the decoder.

Let us analyze the cross entropy now. We can note the unpleasantly high maximum values for every experiment. In such extreme cases, the translation brings rather a (countable) loss of information, rather than its increase, from view of the target language speaker. On the other hand, in the cases where cross entropy reaches its minimum, the TM brings only a negligible amount of misinterpretation into the translation, compared to the corpus evidence.

The other characteristics of the cross entropy distribution are more informative, though. The experiment with 10,000 tr. seg., cs⟶en, max. phr. len. 3 turns out to be a special case again, for its low value not only for the reachability, but also for the cross entropy. The reachability was measured using the whole 10^{th} section of CzEng, but the cross entropy was extracted from translating only 1,119 segments (as listed in Table A.1), hence it presents a less reliable value. However, the parallel experiment using Phr instead of Afact performed similarly with respect to the cross entropy, so we should not disregard this measured value.

The overall trends in the cross entropy show it has its minimum when the maximum phrase length is set to 3. The increase with using longer phrases can be explained by search errors again, as with Phr.

The dependency of which one of the translation directions performs better with regard to the

cross entropy, on the experiment setup or on the OOV rate do not seem to be easily explainable. Several types of errors can come into play (described in [1]), and without more experiments set up more different ways, we cannot surely distinguish the causes of different performance.

5.4 Setup of translations at the t-level

For translating with tectogramatical annotations at hand, we tried to adhere to those relations we consider especially useful for translating. We presupposed that a node in the tectogrammatical dependency parse tree has its governing node as one of its principal properties. We also assumed that the deep-order distance plays an important role. Therefore, we substituted the tuple $\langle <$ deep-order distance $>, <$ governing node's functor $>\rangle$ for the link to the governing node originally present in the "export format" of CzEng 0.9, assigning a special value to it for the root node.

Then, we choosed a translation scheme we wanted to be representative for translating at the tectogrammatical level. It included translating separately the lemmata plus the valency frame[3], the above described joined data about the governing node, and the rest (formemes and various attributes, refer to [2]). After the translation, we did not perform any synthesis or generation of the surface form, as we considered the output of the translation as bearing more information than its possible synthesis into surface. Thus the experiments abstract from discrepancies potentially caused by errors in the synthesis. However, we have not imposed any penalties for the TM not having finished its actual work of translating, as we have not set up any framework for doing so yet. We would also first need more results from translating to be able to benefit from imposing the penalties.

5.5 Results for Tfact

The results are listed in Table 5.4.

5.6 Analysis of the results

We have not run enough experiments with Tfact, unfortunately, to produce more interesting results. This TM proved computationally one of the most demanding, and we also gave the TMs analyzed earlier higher priority. Therefore, only incomplete results are available, only for two modest experiment settings, and only measured using 1,000, and 2,000 testing segments for the translation en⟶cs, and cs⟶en, respectively.

We can note that the translation lexicons for Tfact contain more entries than for example those for Phr. Tfact uses three translation tables, therefore the expected number of entries in its lexicon would be three times number of entries in the comparable Phr's translation lexicon. But the Tfact's lexicon contains somewhat more than that. It implies that the t-lemmata, the combined information about the governing node, or the combined formemes are not as repetitive as the simple surface

[3]Although the valency frame attribute was left empty at least by first several thousands of sentences in CzEng.

tr[sen]	1,000	1,000
dir	en⟶cs	cs⟶en
plen	1	1
lex[kB]	77	77
lex[#]	7,009	7,044
oov	56.7 %	60.1 %
rbl rate	0.03 %	0.006 %

Entropy of decision		
avg	5.1	5.1
stddev	3.1	3.1
min	2.0	1
max	15.9	20.0

Entropy of decision per unit		
avg	2.0	2.0
stddev	0.0	0.1
min	1.5	1
max	2.0	2.0

Table 5.4: Results for the TM Tfact.

forms. This refers to training on data containing 1,000 segment pairs, though, so it is not a very crucial observation. So do not we consider the absolute size of the lexicons very informative.

The other measures prove that this TM would need more training data. Almost 60 % of all testing data were out of vocabulary, and only a few segments were eventually reachable. This did not help the TM be certain, either. Conversely, the entropy of decision was at the highest level seen till now already for this small setup.

All in all, we found the tectogrammatical annotation very specific and probably accurate, but also too demanding for the translation system to use.

Chapter 6

IV-n for all TMs

The results from the measurement of the IV-n are collected in this chapter for better readability, for not losing the context with results for other TMs. The chapter is divided into sections for each TM for which this measure was evaluated. In each section, there are the measured data plotted with a title naming the corresponding (hypothetical) TM, number of segment pairs it used (might use) for training, and the language it is evaluated for[1].

IV-n was defined in the section 2.1. For the reader to understand the plots more easily, we shall describe their different areas and their meaning briefly. The axis named "Times seen in training data" corresponds to the extent the given phrase is known. The left-most coordinate, "Times seen in training data" = 0, corresponds to the empiric OOV. (It is often not plotted as a point in the plot for having to large value.) The second variable axis, named "Phrase length", distinguishes length of phrases the measure is evaluated for. The rearmost line in the plot corresponds to unigrams, the third from rear to trigrams, the foremost line corresponds to phrases of length 10. The value axis, "Times used", expresses total number of occurrences of phrases with the given length, seen in the data the corresponding number of times, in the 10^{th} section of CzEng.

The values were obtained by first counting occurrences in the training data and, with the table of counts, going through the 10^{th} section of CzEng and adding one for each $1..10$-gram to the appropriate number.

6.1 Translating letterwise (Ltr)

These results are not of very importance for machine translating, but they are useful to check the method, and they also tell something about the regularity of both languages at the level of groups of a few letters.

We can notice two basic tendencies in this kind of plots:

1. The curve for unigrams continues furthest to high values of the "Times seen in training data" axis. The curves for higher n-grams go lower.

[1]Unlike OOV, we can immediately make use of IV-n for the language it is evaluated for, whether it is the source language for a particular translation, or the target language. It is not subject to any aligning and phrase filtering, as one-sided models may be.

2. There forms a "bowl" (a local minimum) on individual lines in the plot sometimes.

We will talk about these tendencies later.

What is noticeable about the results for Ltr, is the overall high extent of having known the language unit from the training data (the curves are fairly high above 0, but, most importantly, they spread far to right).

6.2 Phrase translation (Phr)

In the plots for the TM Phr, we can see that the testing data (the 10^{th} section of CzEng) was very similar to the training data, as even the nearest curves cover a pretty significant area with "Times seen in training data" relatively high. We claim that the high values (causing the curves to resemble a bowl) correspond to the same sentences that came to the corpus from different sources (but they have the same origins), as it is highly improbable for a set of 10-grams to occur this number of times[2] and not to come from the same sentence.

6.3 A-factored translation (Afact)

For the a-factored translation, we have counted the IV-n measure for each of the translating (or, precisely, "mapping") step: for translation of the lemmata, for translation of the morphology and for the generation of the final surface forms. Each such translation is then influenced by the IV-n for the lemmata and morphology at the source side and by the IV-n for the tuple $\langle \text{lemma}, \text{morphology} \rangle$ (called "lemmor" in captions of the plots) at the target side.

In the measured data, we can observe inarguable improve from the corresponding plots for translating in unfactored fashion. The curves spread much more to the right side. Another noteworthy observation regards the difference between English and Czech morphological tags. The plot for English ones is very spare. That reflects the English morphological tagset being much smaller than the Czech one.

The conclusion to be drawn from these plots speaks for use of factored models, at least without abundance of training data.

6.4 Edges of the a-tree (Aedge)

This section and the following one are devoted to a hypothetical TM, or rather to a mere feature of linguistic data in the corpus. The implied TMs could not translate a single sentence. By including these two experiments, we want to measure useability of tree-like structures for machine translating in practice, or, more precisely, characterize the two types of trees (a-trees and t-trees) with respect to machine translating.

In these experiments, we extract tuples of nodes from a-trees[3] and t-trees[4] and measure the IV-n

[2] Namely, it was 5379.

[3] Parse trees at the a-level (annalytical).

[4] Parse trees at the t-level (tectogrammatical).

for such pairs. The nodes in the edges retain all their original linguistic information (i.e. lemmata and tags) for the experiment. The edges we operate on are sorted into segments the same way their dependent nodes were originally in the corpus. This allows us to compute IV-n also for longer phrases than just for single edges.

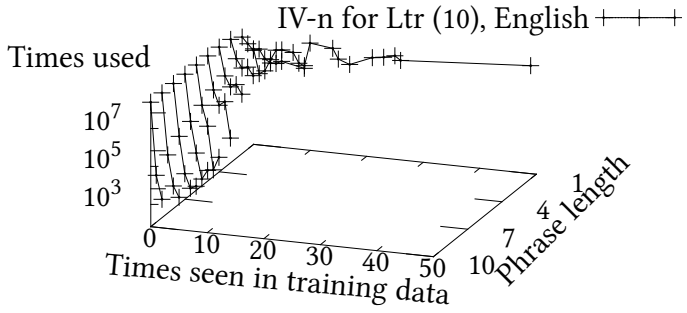
As you can see in the figure 6.5, surprisingly, also longer phrases of edges could be taken as the basic language unit for a TM, as they display a high enough extend of repetitiveness. The shape of the plots for 10,000 training segments is very similar to plots for the lemma-morphology combination for Afact. That is natural, as both concepts, the a-edge, and the lemma-morphology combination, are also very similar.

The last two plots in this section display the slice of the other plots, taken at where "Phrase length" = 1, for large training data. They demonstrate the bowl-like appearance of this kind of plots, and give a clue what it is caused by. Let us describe the plot from left to right, in correspondence to extent of knowing a particular a-edge. There existed a lot of a-edges, that would be useful for translating the test section, but they were not seen in the training data at all. A number of a-edges was very useful in the testing data, while having been seen only a few times in the training data (this corresponds to the descending part of the plot). Some common a-edges were present in both the data (the middle part of the plot). Finally, some very common a-edges occured many times in both the data (the last, ascending part of the plot).

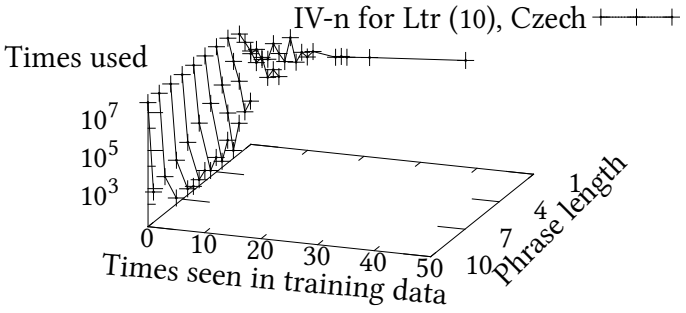
6.5 Edges of the t-tree (Tedge)

Generally, we have not, unfortunately, experimented with the t-layer much, because it came to the t-layer only after having explored the simpler models. Therefore, also the IV-n measurements were only few for this layer.

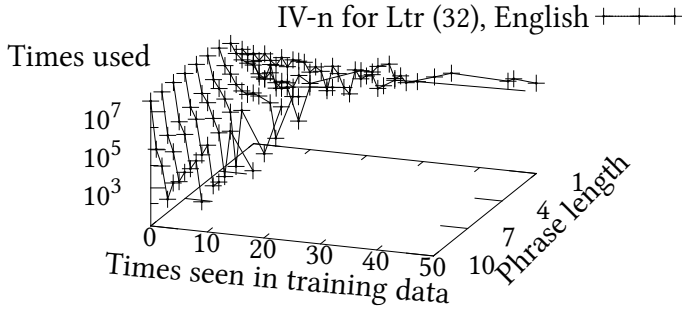
In the plots in Figure 6.6, you can note both the phenomenons mentioned in the introduction to this section. The results were somewhat interesting in that the longer phrases were always unreachable. That means that applying a phrase based model to t-level annotated corpus cannot yield very good results, because too many phrases would be just out of vocabulary.



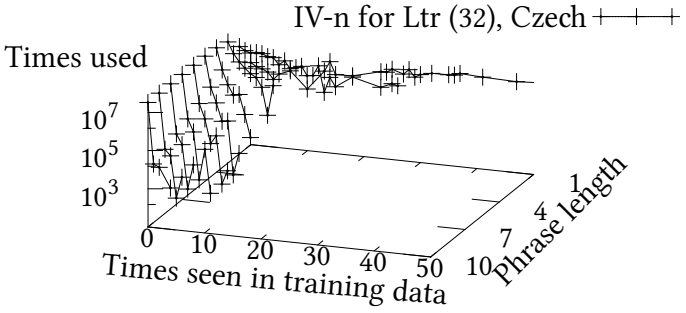
(a)



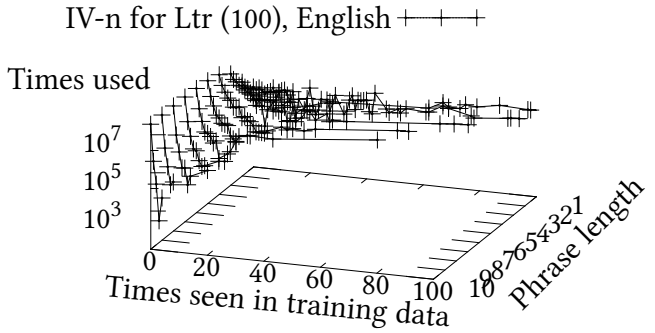
(b)



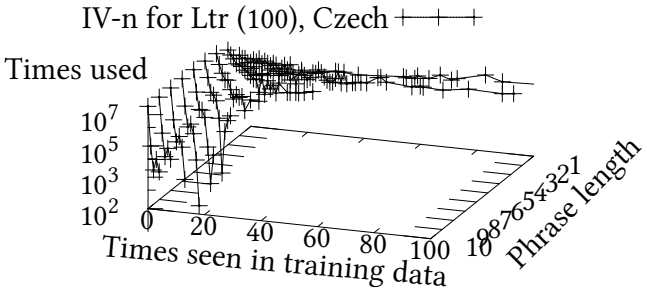
(c)



(d)



(e)



(f)

Figure 6.1: IV-n for the TM Ltr.

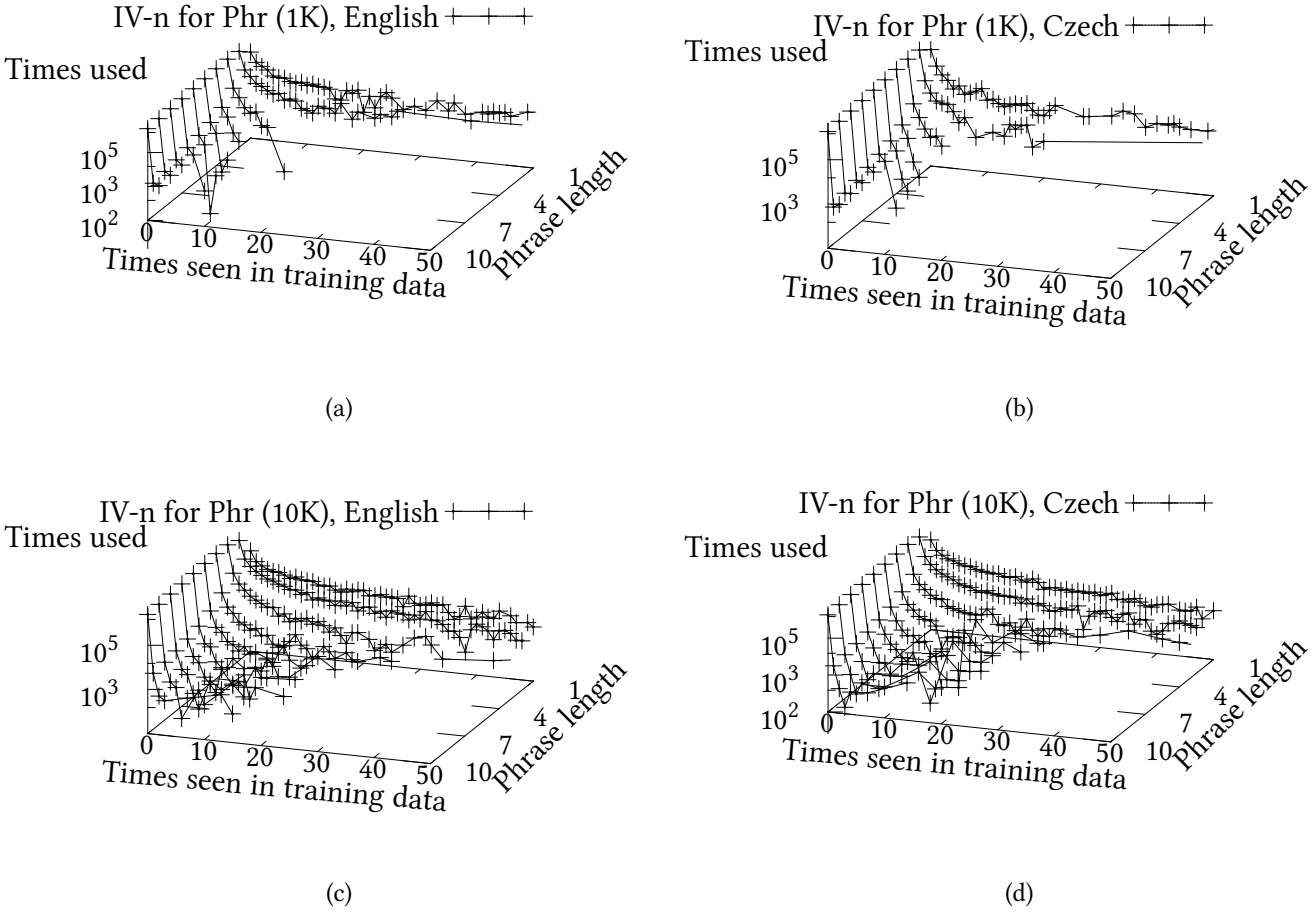
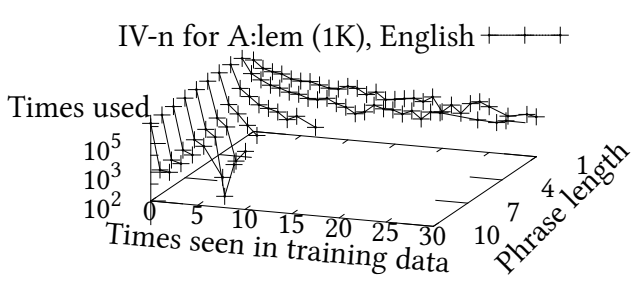
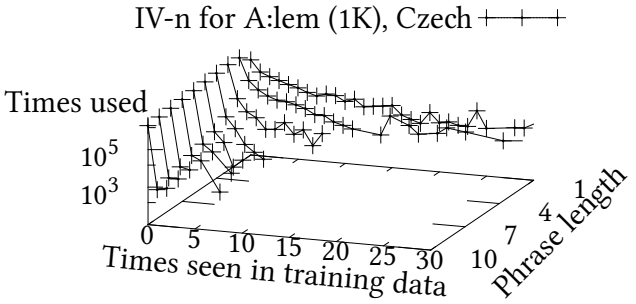


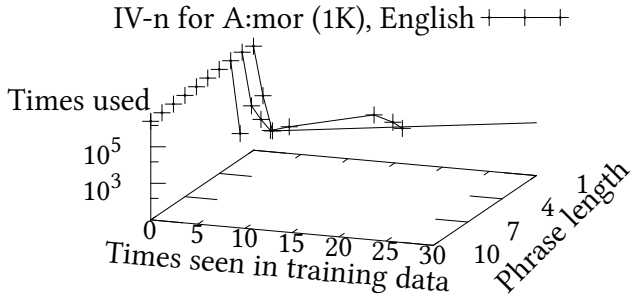
Figure 6.2: IV-n for the phrase TM.



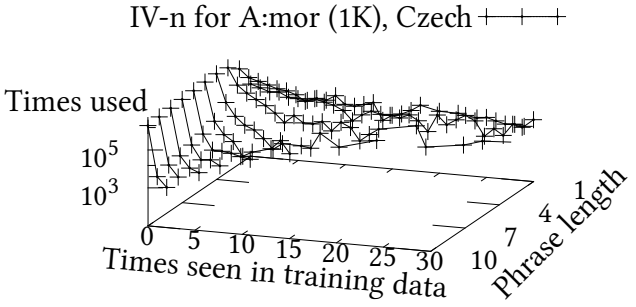
(a)



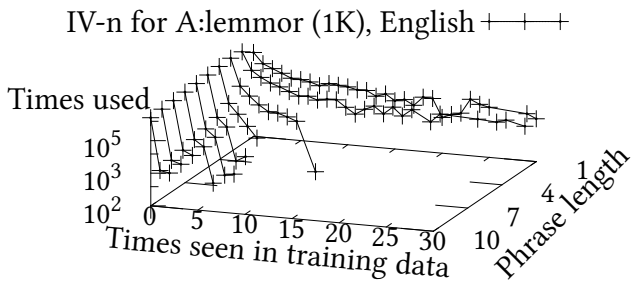
(b)



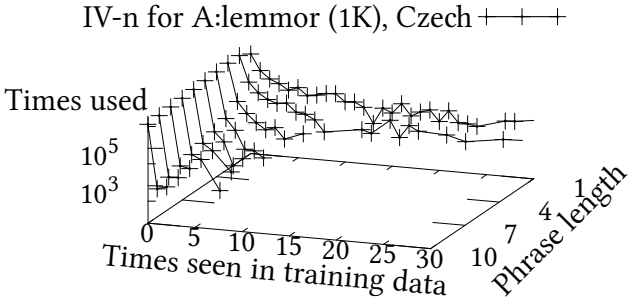
(c)



(d)

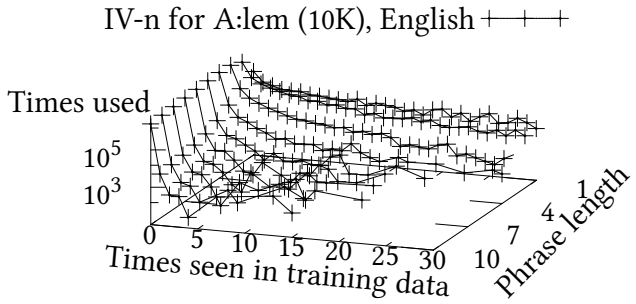


(e)

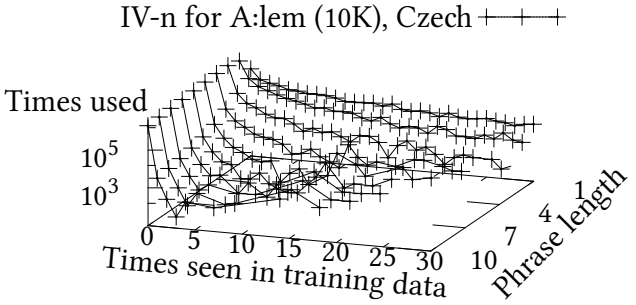


(f)

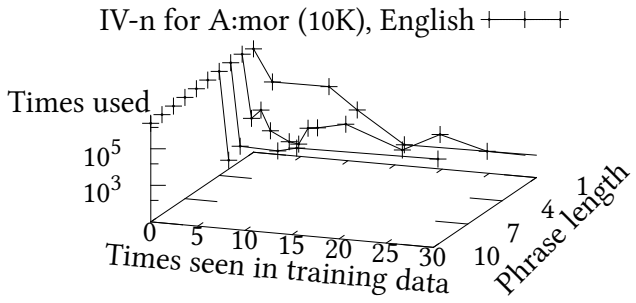
Figure 6.3: IV-n for the a-factored TM.



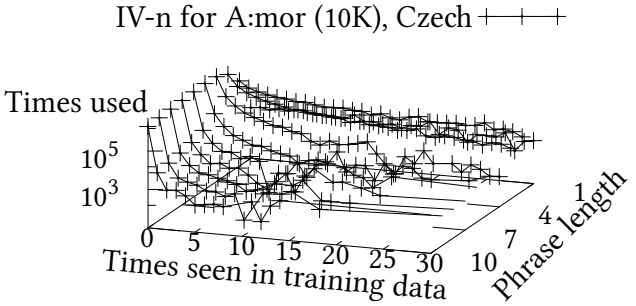
(a)



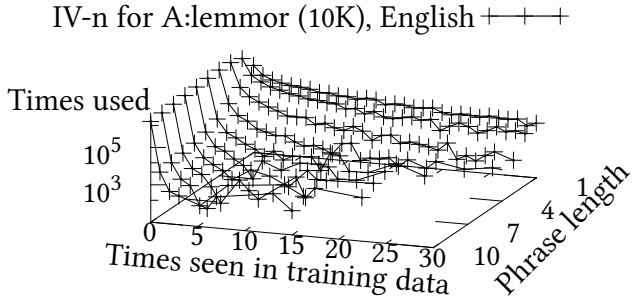
(b)



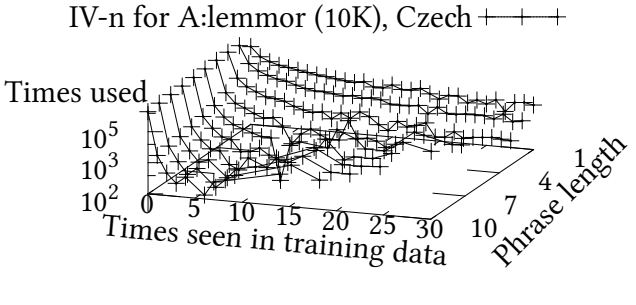
(c)



(d)



(e)



(f)

Figure 6.4: IV-n for the a-factored TM, continuation.

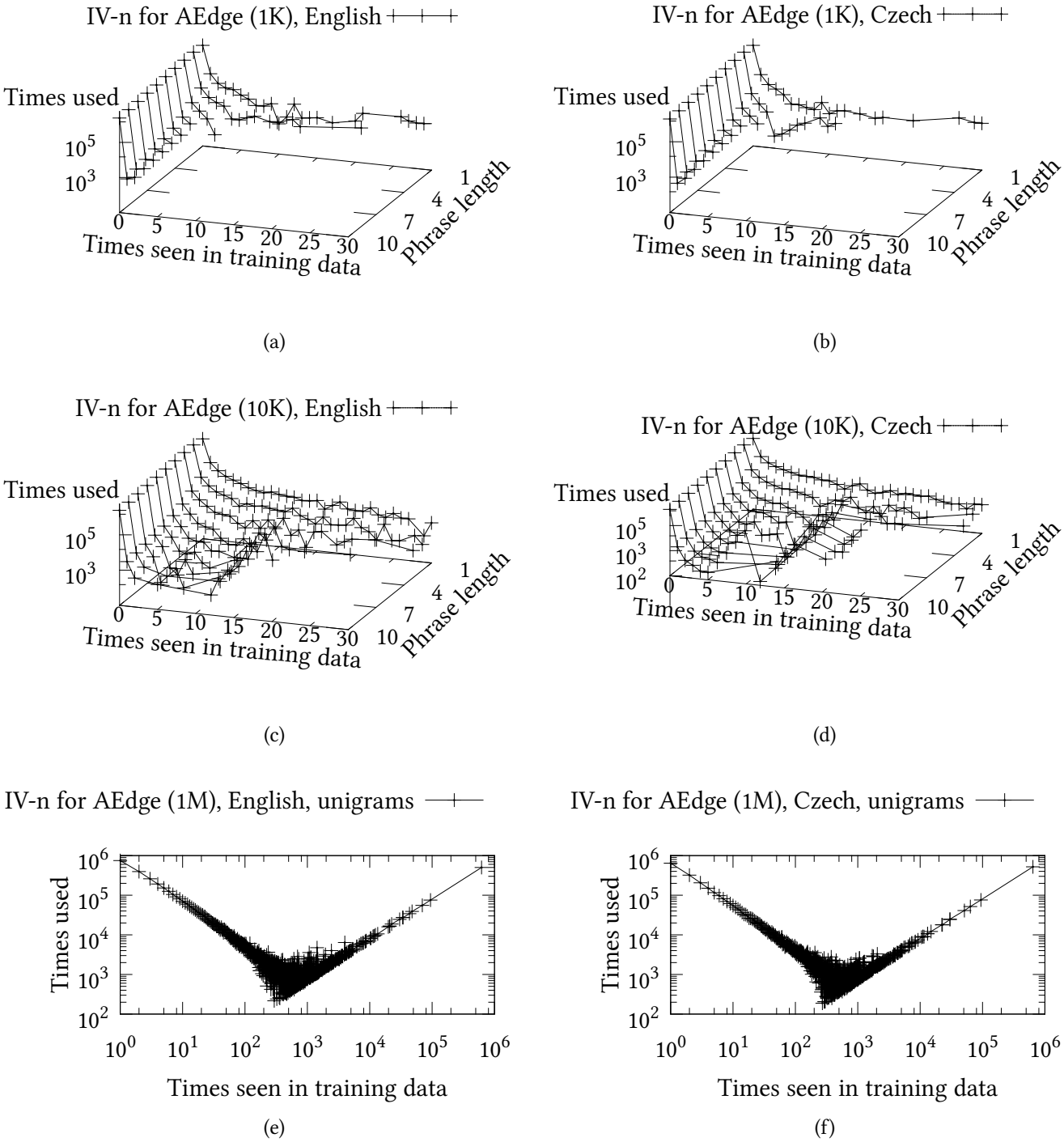
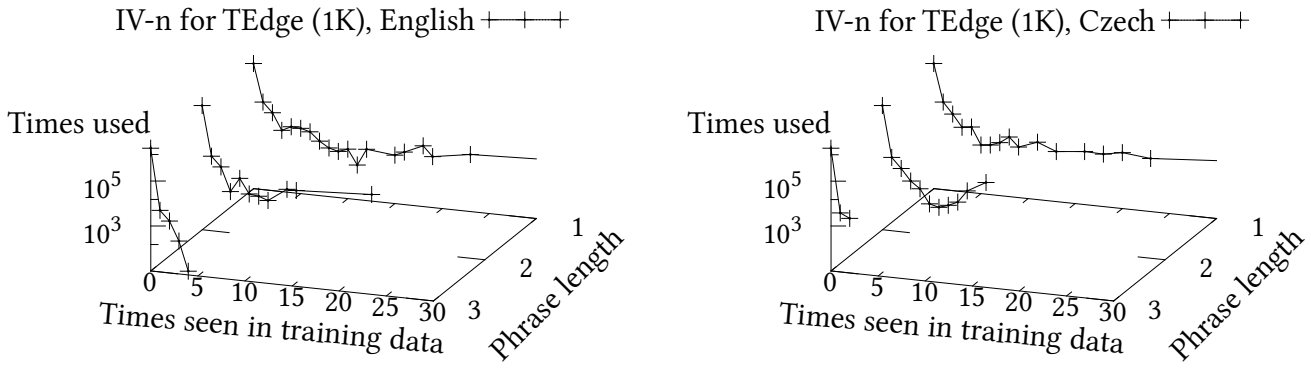


Figure 6.5: IV-n for edges of the morphological parse tree.



(a) (b)

Figure 6.6: IV-n for edges of the tectogrammatical parse tree.

Chapter 7

Conclusion

We have selected and discussed measures for measuring performance of TMs according to three distinct criterions. The cross entropy/TM perplexity was devised for a linguistic problem involving two languages as a novel measure. We have implemented all the measures to evaluate them using both toy and real translation models. We have set their empirical bounds using extremal cases of TMs and also proved their applicability to several real TMs. Further models we originally intended to include in our experiments (like hierarchical TMs) unfortunately exceed the scope of this thesis, mainly due to their greater complexity.

We would like to continue in this work and both extend size of the measured models, and add models of new kinds, like hierarchical TMs. The difference between generating target surface forms from source m-lemmata and morphology, and generating target m-lemmata from source t-lemmata and tectogrammatical tags would be of particular interest for us. We would surely also want to compare a better trained phrase TM with a comparably trained hierarchical model. These topics are left for further research.

Bibliography

[1] Michael Auli, Adam Lopez, Hieu Hoang and Philipp Koehn: A Systematic Analysis of Translation Model Search Spaces, 2009. In: Proceedings Of the 4th EACL Workshop on Statistical Machine Translation, pages 224–232.

[2] Ondřej Bojar and Zdeněk Žabokrtský. 2009. CzEng 0.9: Large Parallel Treebank with Rich Annotation. Prague Bulletin of Mathematical Linguistics, 92.

[3] 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, Evan Herbst, Moses: Open Source Toolkit for Statistical Machine Translation, Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic, June 2007.

[4] Ronald Rosenfeld, A Maximum Entropy Approach to Adaptive Statistical Language Modeling, 1996. Page 2.

[5] Wikipedia, `http://en.wikipedia.org/wiki/Entropy_(information_theory)`

Appendix A

Sizes of data in experiments

The table A.1 lists number of sentences the experiments were actually run with. Although the goal was to measure all the TMs equally, on the whole 10^{th} section of CzEng, the time that would be needed for that was intolerable. Therefore, the calculations were usually stopped after some time. The number in the table expresses number of segments/segment pairs taken from the top of the 10^{th} section of CzEng that were translated in the direction listed in the 2^{nd} column to yield basis for calculations of the OOV, decision entropy and TM perplexity.

Model (max. phr. length)	Direction	Training [sen]	Translated [sen]
Sen	both	ALL	802,392
Pas	both	ALL	802,392
Ltr	cs→en	100	187
Ltr	en→cs	100	118
Phr(1)	en→cs	1,000	138,866
Phr(3)	en→cs	1,000	287,045
Phr(3)	cs→en	1,000	255,891
Phr(10)	en→cs	1,000	762,662
Phr(1)	en→cs	10,000	36,226
Phr(1)	cs→en	10,000	802,392
Phr(3)	en→cs	10,000	24,605
Phr(3)	cs→en	10,000	802,392
Phr(10)	en→cs	10,000	214,414
Afact(1)	en→cs	1,000	202,901
Afact(1)	cs→en	1,000	216,762
Afact(3)	en→cs	1,000	80,549
Afact(3)	cs→en	1,000	456,447
Afact(10)	en→cs	1,000	184,892
Afact(10)	cs→en	1,000	223,951
Afact(1)	en→cs	10,000	209,478
Afact(1)	cs→en	10,000	282,381
Afact(3)	en→cs	10,000	100,000
Afact(3)	cs→en	10,000	1,119
Afact(10)	en→cs	10,000	74,329
Afact(10)	cs→en	10,000	50,353
Afact(1)	en→cs	1,000,000	385,024
Afact(1)	cs→en	1,000,000	15,000
Tfact(1)	en→cs	1,000	1,000
Tfact(1)	cs→en	1,000	2,000

Table A.1: Size of data in experiments. "ALL" means all the experiments with with the specified TM. The 10^{th} section of CzEng contains 802,392 sentences.