Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE

Karel Bílek

Porovnání metod česko-ruského automatického překladu

Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: doc. RNDr. Kuboň Vladislav, Ph.D.

Studijní program: Informatika

Studijní obor: matematická lingvistika

Praha 2014

Those are all the people that I want to thank:

- Vladislav Kuboň – my supervisor, for helping me with the thesis in general, and the related projects
- Natalia Klyueva – a college which I worked the most with
- Ondřej Bojar and Aleš Tamchyna – for helping me with the Moses system and eman evaluation manager
- Martin Popel – for helping me with TectoMT system and gave me lots of ideas in general
- Rudolf Rosa – for giving me ideas for better text structuring
- countless friends that supported me when I just wanted to give up.

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular the fact that the Charles University in Prague has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 paragraph 1 of the Copyright Act.

In …….. date ………… signature of the author

Název práce: Porovnáni metod česko-ruského automatického překladu

Autor: Karel Bílek

Katedra: Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: doc. RNDr. Vladislav Kuboň, Ph.D., Ústav formální a aplikované lingvistiky

Abstrakt: V této práci představuji několik metod česko-ruského automatického překladu, včetně jak více historických, tak více moderních systémů, a včetně jak frázovách, tak pravidlových systémů. Nejdříve stručně popisuji lingvistické základy češtiny a ruštiny a jejich společnou historii a rozdíly. Poté popisuji automatizaci, vytváření a zlepšování některých ze systémů automatického překladu, společně s jejich porovnáním, s použitím jak automatických metrik, tak omezené lidské anotace. Zároveň s tím také popisuji vytvoření několika korpusů česko-ruských paralelních dat a ruských monolingválních dat.

Klíčová slova: Čeština, ruština, strojový překlad

Title: A Comparison of Methods of Czech-to-Russian Machine Translation

Author: Karel Bílek

Department: Institute of Formal and Applied Linguistics

Supervisor: doc. RNDr. Vladislav Kuboň, Ph.D., Institute of Formal and Applied Linguistics

Abstract: In this thesis, I am presenting several methods of Czech-to-Russian machine translation, including both historical approaches and more modern ones, and including both phrase-based and rule-based systems. I am first briefly describing the linguistic background of Czech and Russian, and their common history and differences. Then, I am describing automating, building and improving some of the machine translation systems, together with their comparison, using both an automated metric and a limited human annotation. Meanwhile, I am also describing the creation of a several corpora of Czech-Russian parallel data and Russian monolingual data.

Keywords: Czech, Russian, Machine translation

Contents

1 Brief comparison of Slavic languages 4

1.1 History . . . 4

1.1.1 Common history . . . 4

1.1.2 Division of the languages . . . 5

1.2 Slavic languages overview . . . 7

1.2.1 Morphology . . . 7

1.2.2 Word order . . . 8

1.2.3 Grammatical categories and their changes . . . 9

2 Translation systems 12

2.1 Statistical vs. rule-based – an overview . . . 12

2.2 Rule-based systems . . . 13

2.2.1 RUSLAN . . . 13

2.2.2 Česílko 1.0 . . . 15

2.2.3 Česílko 2.0 . . . 16

2.2.4 PC Translator . . . 17

2.3 Statistical systems . . . 18

2.3.1 Google Translate . . . 18

2.3.2 Bing Translator . . . 18

2.3.3 Yandex Translate . . . 19

2.3.4 Moses . . . 19

2.4 Hybrid systems . . . 24

2.4.1 TectoMT . . . 24

3 Data 28

3.1 Parallel data . . . 28

3.1.1 WMT test sets . . . 28

3.1.2 Intercorp . . . 29

3.1.3 Subtitles . . . 30

3.1.4 UMC corpus . . . 31

3.1.5 Wiki titles . . . 32

3.2 Monolingual Russian data . . . 32

3.2.1 News Crawl . . . 32

3.2.2 Common Crawl . . . 32

3.2.3 Yandex . . . 33

3.3 Statistics 33
3.4 Unused data 34
3.4.1 Lib.ru 34

4 Experiments 35
4.1 Experiments on historical systems 35
4.1.1 RUSLAN 35
4.1.2 Česílko 1.0 36
4.1.3 Česílko 2.0 36
4.2 PC Translator 38
4.3 Web translators 39
4.3.1 Google Translate 39
4.3.2 Bing Translator 39
4.3.3 Yandex Translate 39
4.4 Moses experiments 40
4.4.1 Alternative eman seeds 40
4.4.2 Factored translation experiments 41
4.4.3 Full setup 43
4.5 TectoMT experiments 45
4.6 Future work 47
4.6.1 Better morphology and parser 47
4.6.2 Traslation by letters 47

5 Results 49
5.1 Overview 49
5.2 Automated metrics 49
5.2.1 Discussion 50
5.3 Human evaluation 51
5.4 Some observation about typical mistakes 53

Conclusion 57

Bibliography 58

Attachments 67

A Sample of experiment results 68
A.1 Sample sentences 68
A.1.1 Intercorp 68

A.1.2 WMT 68
A.2 PC Translator 69
A.2.1 Intercorp 69
A.2.2 WMT 70
A.3 Google Translate 71
A.3.1 Intercorp 71
A.3.2 WMT 71
A.4 Bing Translator 72
A.4.1 Intercorp 72
A.4.2 WMT 73
A.5 Yandex Translate 74
A.5.1 Intercorp 74
A.5.2 WMT 74
A.6 Moses 75
A.6.1 Intercorp 75
A.6.2 WMT 76
A.7 TectoMT 77
A.7.1 Intercorp 77
A.7.2 WMT 77

B Data on the attached disk 79
B.1 Corpora 79
B.1.1 Original data 80
B.1.2 Scripts 81
B.2 Systems 83
B.2.1 RUSLAN 83
B.2.2 Česílko 1.0 83
B.2.3 Česílko 2.0 83
B.2.4 Online systems 83
B.2.5 PC Translator 83
B.2.6 Moses 84
B.2.7 TectoMT 84
B.3 Evaluation 84

1. Brief comparison of Slavic languages

This section is an overview of both the history of Slavic languages and of some characteristics of those languages in general.

1.1 History

1.1.1 Common history

Using comparative linguistics, Slavic languages in general can be traced back to Indo-European language family, with Proto-Indo-European (PIE) as its reconstructed ancestral language.

As greatly described in MALLORY; ADAMS, 2006, taking Proto-Indo-Europeans as unified people with unified history can be somehow controversial – we are, as noted there, trying to “put absolute dates on a hypothetical construct”. Still, based on the reconstructed Proto-Indo-European language having, for example, word for wheeled wehicles, we can date Indo-Europeans on Great Eurasian Plain to approximately 4000 BC, as noted by SUSSEX; CUBBERLEY, 2011.

SUSSEX; CUBBERLEY, 2011 further notes that our knowledge of this part of European pre-history is “sketchy and partly conjectural”. He further notes:

> “Although we can only guess how far their territory extended, it is possible that at least the European center of the Indo-European homeland – if not the original homeland itself (on one widely held view) – was in what is now Western Ukraine, and that they spoke a fairly homogeneous language.”

MALLORY; ADAMS, 2006 further describes the reconstruction of this PIE. In here, however, this excerpt from RINGE, 2008 will suffice:

> “Though there continue to be gaps in our knowledge of PIE, an astonishing proportion of its grammar and vocabulary are securely reconstructable by the comparative method. As might be expected from the way the method works, the phonology of the language is relatively certain. Though syntactic reconstruction is in its infancy, PIE syntax is also relatively uncontroversial because the earliest-attested daughter languages agree so well.”

With similar logic, we can reconstruct an ancestral language to all Slavic languages, call it Proto-Slavic and put it on a place in space and time.

SUSSEX; CUBBERLEY, 2011 puts the emergence of Proto-Slavic at around 2000-1500 BC.

SCHENKER, 1993 adds, rather poetically:

> "The Slavs were the last Indo-Europeans to appear in the annals of history. Slavonic texts were not recorded till the middle of the ninth century and the first definite reference to the Slavs' arrival on the frontiers of the civilized world dates from the sixth century AD, when the Slavs struck out upon their conquest of central and south-eastern Europe. Before that time the Slavs dwelled in the obscurity of their ancestral home, out of the eye-reach of ancient historians. Their early fates are veiled by the silence of their neighbours, by their own unrevealing oral tradition and by the ambiguity of such non-verbal sources of information as archaeology, anthropology or palaeobotany."

KORTLANDT, 1982 breaks this period into several sub-periods, like Balto-Slavic, Middle Slavic and Proto-Slavic; while Balto-Slavic, in his approximation, appears at 2000 BC, Late Proto-Slavic disappears and disintegrate in 900-1200 AD. This division is also slightly mentioned in SCHENKER, 1993.

SUSSEX; CUBBERLEY, 2011 also sums up the spacial location of Slavic areas before breaking to individual languages:

> "By the fourth century AD the Slav area stretched from the Oder (Pol *Odra*) River in the west to the Dnieper (Rus *Dnepr*, Ukr *Dnipró*) in the east. In the north they had reached the Masurian Lakes in central Poland, the Baltic Sea and the Pripet (Pol *Prypeć*; also Eng *Pripyat*, from Ukr *Prýp'jať*) Marshes. During this period the Slavs would have spoken a fairly uniform language. Although dialect differences soon began to appear, resulting *inter alia* in the division into Baltic, Slavic or an intermediate Balto-Slavic, the pace of linguistic change was relatively slow."

According to CURTA, 2004, some form of Slavic was still used as a *lingua franca* in Avar qaganate by about 700 AD, but no further than in 800 AD.

1.1.2 Division of the languages

Continuing its narrative, SUSSEX; CUBBERLEY, 2011 notes about breaking of Proto-Slavic:

> “According to general consensus in what is still a controversial area, the real break-up of Proto-Slavic unity began about the fifth century AD. There seems to have been a steady expansion to the north and east by the Eastern Slavs. For the others there is evidence that their migrations were related partly to the disintegration of the Roman and Hun empires and the ensuing vacuum in Central Europe.
>
> One group of Slavs moved westwards, reaching what is now western Poland and the Czech Republic, and the eastern and north-eastern part of modern Germany.
>
> A second wave broke away to the south towards the Balkan Peninsula, where they became the dominant ethnic group in the seventh century, some (in the east) in turn being conquered by the Bulgars, a non-Slavic people of Turkic Avar origin.”

MALLORY; ADAMS, 2006 also puts the point of Slavic break-up at arount 500, while for example KORTLANDT, 1982 puts it at 900-1200 AD, and SCHENKER, 1993 puts it at about 900 AD.

While sources disagree on what exactly are all the Slavic languages in which group (partly because question of a language distinction is a political one), all sources[1] agree on the division to South Slavic languages, East Slavic languages and West Slavic languages; West Slavic being the westwards moving group from the above narrative, South Slavic being the group moving to Balkan and Eastern Slavic being the group moving eastwards.

According to SUSSEX; CUBBERLEY, 2011, the West Slavic language group consists of Czech, Slovak, Polish, Kashubian and Sorbian. As mentioned, some sources divide the languages slightly differently; for example SIEWIERSKA; UHLÍŘOVÁ, 1998 breaks Sorbian into Upper and Lower Sorbian; other sources take Kashubian as only a variant of Polish.

South Slavic language group consists of Serbo-Croatian, Bulgarian, Slovenian and Macedonian. As before, this list is taken from SUSSEX; CUBBERLEY, 2011 – the question of Serbo-Croatian language unity, for example, is a highly politiced one, thanks to recent military conflicts in the region.

The Eastern Slavic language group consists of Russian, Ukrainian and Belarussian, again according to SUSSEX; CUBBERLEY, 2011.

---

[1] SUSSEX; CUBBERLEY, 2011, MALLORY; ADAMS, 2006, KORTLANDT, 1982 and others already mentioned

to wash (someone)	mýt	мыть
to wash (self)	mýt se	мыться

Table 1: Reflectives in Russian vs. Czech

1.2 Slavic languages overview

In this section, I am almost exclusively citing SUSSEX; CUBBERLEY, 2011. While there are books like COMRIE; CORBETT, 2003, they describe the languages one-by-one, whereas SUSSEX; CUBBERLEY, 2011 compares language properties across all Slavic languages in a concise and clear manner that I haven't been able to find anywhere else.

Therefore, even when the book is not quoted and cited *directly*, the general information in this chapter is heavily informed by it.

1.2.1 Morphology

On the morphological typology, Slavic languages belong to synthetic inflectional languages. They are morphologically rich, with a sophisticated affix system and a little analytical approach to verb morphology (for forms like future tense).

Slavic word is composed of roots (which can be one or several) and affixes (prefixes and suffixes). Prefixes usually modify the word's meaning somehow (for example, "ne-" for negative), while suffixes modify the word's class or one of its grammatical categories.

Suffixes are of several types. The ones appearing first are the derivational suffixes, which can determine the word's class "in familiar processes like abstract and agent nominalization, verbalization, adjectivalization", as noted in SUSSEX; CUBBERLEY, 2011. Next type of suffix are endings[2] that mark one of several grammatical categories – like "infinitive, person, number, tense, case and gender", as noted in SUSSEX; CUBBERLEY, 2011.

In some Slavic languages – like Russian – reflexivity is expressed by special suffix, called *post-inflectional suffix* by SUSSEX; CUBBERLEY, 2011; whereas in others – like Czech – a separate word is used. See for example Table 1 (also compare to the voices in the section 1.2.3).

Inflectional categories that exist in Slavic languages are described in the Table 2 (definiteness and deixis as inflectional categories are, however, not relevant to either Czech or Russian, but are used as such in Macedonian or Bulgarian).[3]

Morphological process are quite similar across Slavic languages – however, using

---

[2] Called inflectional suffixes in SUSSEX; CUBBERLEY, 2011

[3] The table is, again, from SUSSEX; CUBBERLEY, 2011.

Verbs	**Nominals**	**Both**
Tense	Case	Person
Aspect	Definiteness	Gender
Mood	Deixis	Number
Voice		

Table 2: Inflectional categories

either different affixes or using the same affixes, but with slightly shifted properties (diferent categories, etc.).

In general, it can be said that languages from the Slavic family are morphologically richer than other languages. The result of this richness is that the number of word types in a given corpus is significantly higher, when compared with more analytical languages (like English).

1.2.2 Word order

Languages like English use word order for marking constituents in a sentence. Slavic languages, on the other hand, use inflective processes, like agreement[4], for the same type of information – subject *agrees* with predicate, and so on.

Because those relations are marked by inflexion, it "allows" the Slavic languages to be of freer word order. The *standard* order is SVO – however, this is possible to change when different emphasis is needed.

In particular, this reffers to the so-called Functional Sentence Perspective. Very simply said, it describes the sentence as consisting of two parts – Topic and Comment, appearing in this order and being separated by a verb. Topic is the part of sentence, *about which we say some information* – while Comment is the *new information*. Most importantly, Topic doesn't have to be a grammatical subject of the sentence – for example, the Czech sentence "Ve městě bydlí strašidla" (*Ghosts live in the city*, literally *In city live ghosts*).

Slavic languages usually (with some exceptions like Bulgarian and Macedonian) don't have particles or any other means of marking definitiveness – the only mean is the definite information being in the topic of the sentence. In other view, the functional sentence perspective can be viewed as "replacing" the definitive particles, and is usually translated as such in translation to English.

In respect to the translation between English and Slavic languages, word order can be seen as something that's hard to translate correctly. With respect to translation between Slavic languages, it could possibly help us, since we don't have to, in theory, change the word order too much.

---

[4] SUSSEX; CUBBERLEY, 2011 lists concord, agreement and government

genitive	invocant nomen **Domini** nostri Iesu Christi	who call on the name **of** our **Lord** Jesus Christ	všem, kdo na jakémkoli místě vzývají jméno našeho společného **Pána** Ježíše Krista
ablative	gratia vobis et pax a Deo Patre nostro et **Domino** Iesu Christo	Grace and peace to you **from** God our Father and **the Lord** Jesus Christ	Milost vám a pokoj od Boha, našeho Otce, a od **Pána** Ježíše Krista

Table 3: Demonstrating ablative and genitive conflation on 1 Corinthians

singular	one cow	ena krava	jedna kráva
dual (in Slovenian)	two cows	dve kravi	dvě krávy
plural	three cows	tri krave	tři krávy

Table 4: Demonstrating duals on Slovenian

1.2.3 Grammatical categories and their changes

In this section, I will show several grammatical categories and their change from PIE to Slavic languages.

Cases

PIE had at least eight cases – nominative, accusative, genitive, dative, instrumental, locative, ablative and vocative. SUSSEX; CUBBERLEY, 2011 lists these eight cases; RINGE, 2008 also adds allative (which, however, survived only in old Hittite).

In Slavic languages, ablative and genitive were conflated into just genitive. We can demonstrate this conflation on of two phrases from New Testament.

Old Latin retained both ablative and genitive. Latin genitive of *dominus* (lord, master) is *dominī*, latin ablative of the same word is *dominō*. Both these forms were used in the very beginning of 1 Corinthians in Latina Vulgata (latin version of The Bible).

In the Table 3, you can see the translation to both Czech and English. Both cases are inflated in Czech as genitive "(našeho) pána".[5]

Numbers

PIE had three numbers – singular, plural and dual.[6]

In most of the Slavic languages (including Czech and Russian), dual disappeared, leaving only traces in the grammar. One of the languages where dual remained in full is Slovenian. To illustrate dual in Slovenian, I have added a comparison of "one cow", "two cows" and "three cows" in Slovenian and Czech in Table 4.

---

[5] Bible sources: WIKISOURCE, 2013, BIBLICA, 1973, 1978, 1984, 2011, FLEK et al., 2012. Note that English and Czech translations are not actually translations from Latin, but from more primary sources, but it will suffice for this simple comparison.

[6] According to both SUSSEX; CUBBERLEY, 2011 and RINGE, 2008.

active	νιπτω	I wash (someone)	myji (někoho)
medium	νίπτομαι	I wash (myself)	myji se
passive	νίπτομαι	I am washed (by somebody)	jsem myt

Table 5: Voices in Classic Greek vs. Czech

In Czech, dual was retained in declensions of several words, like “hands”; in Russian, the dual “рукама” survives in some dialects, but is generally incorrect.[7]

Genders

PIE had three genders, masculine, feminine and neutral.[8] Slavic languages retained these genders, refining them with added features Personal and Animate.

Tenses

According to SUSSEX; CUBBERLEY, 2011, late PIE had six tenses: present, future, aorist, imperfect, perfect and pluperfect.

The tenses were somehow retained in Slavic languages; however, they are used more analytically and with the help of auxiliary verbs – for example, “budu zpívat” (I will sing) in Czech, or “буду петь” in Russian.

Moods

PIE had four moods: indicative, subjunctive, optative and imperative.[9]

In Slavic languages, imperative was replaced by the optatives, and subjunctive mood became conditional.

Voices

PIE had an active and a mediopassive voice.[10]

In Slavic languages, this was refined as active and a passive voice, while reflexives, in a way, took the function of a mediopassive voice.

Since mediopassive voice will probably be unknown in general to the reader, I have added an example of Classic Greek that still retained it, in Table 5,[11] with both English and Czech translation.

---

[7] See OFFORD, 1996, page 18.

[8] As noted in both SUSSEX; CUBBERLEY, 2011 and RINGE, 2008.

[9] According to both SUSSEX; CUBBERLEY, 2011 and RINGE, 2008

[10] According to both SUSSEX; CUBBERLEY, 2011 (where the mediopassive voice is called “middle voice”) and RINGE, 2008

[11] See for example PARKER, 2009, ARCHIBALD, 2008

imperfective determinate	нести́	nést
imperfective indeterminate	носи́ть	nosit
perfective	понести	ponést

Table 6: Aspects in Czech and Russian

Aspects

PIE had distinction between two aspects – eventive and stative; eventive aspect being further divided into perfective and imperfective aspect.[12]

In Slavic, the stative aspect is degrammatized[13]; however, the perfective / imperfective distinction became more important than in other Indo-European languages. SUSSEX; CUBBERLEY, 2011 calls the growing distinction “the most important development in Proto-Slavic”.

Imperfective motion verbs were also added determinate/non-determinate distinction.

This determinate / non-determinate and perfective / imperfective distinction is present in both Czech and Russian. For the demonstration on the two languages, see Table 6 and note, how hard would be to correctly translate the distinction into English.

---

[12] See RINGE, 2008, page 24

[13] ANDERSEN, 2013

2. Translation systems

Machine translation (MT) is a task that's as old as the computer. When the very first computers were created for the task of encryption and decryption, one of the other areas of interest was translation of natural languages.[1]

Machine translation between Czech and Russian has, too, some history.

In this chapter, I am describing both historical and more recent approaches for machine translation between those two languages. In the chapter 4, I am then describing our experiments with those systems.

2.1 Statistical vs. rule-based – an overview

MT systems has historically used many different approaches. One way of classifying them is on the axis of rule-based vs. statistical.

In general, we can re-use the definition, used in BOJAR, 2012, which is as follows[2]:

- rule-based MT systems:
    - use analysis, transfer and synthesis steps
    - use formal grammars
    - use hand-made dictionaries
    - have linguistic information hard-coded and therefore aren't language-agnostic
- statistical MT systems
    - use more variants of outputs, rank them with some score, and choose the best one
    - train internal dictionaries from big parallel data
    - have more compact translation core, their inner working are less obvious
    - use statistics instead of linguistic rules and therefore are more language-agnostic

---

[1] As noted in the introduction in KOEHN, 2010 – "The history of machine translation goes back over 60 years, almost immediately after the first computers had been used to break encryption codes in the war, which seemed an apt metaphor for translation: what is a foreign language but encrypted English?"

[2] The following list is a rough translation from the mentioned book

However, with actual, real-life systems, the distinction is usually not as clear-cut. For example, statistical MT systems like Moses (2.3.4) can get significantly better results with added linguistic information; on the other hand, systems like TectoMT (2.4.1), which can for some intents and purposes be classified as more rule-based, have individual parts in some way based on statistics.

2.2 Rule-based systems

2.2.1 RUSLAN

RUSLAN is a machine-translation system, developed between 1984 and 1988 at several departments of Charles University, Prague. RUSLAN firmly belongs to the *rule-based* category, since at that timeframe, statistical machine translation wasn't even invented yet.

Description of the system can be found in OLIVA, 1989 or HAJIČ, 1987 – however, the reader has to bear in mind that both the systems *and* their manuals and descriptions are severly dated. (At least for me personally, especially the book OLIVA, 1989 was hard to read and navigate in.) Contemporary (but not as detailed) description of the system can be found in BOJAR; HOMOLA; KUBOŇ, 2005.

The whole RUSLAN system has several components:

- preprocessing, written in Pascal
- morphological analysis, using dictionary, written in Q-Systems (described further) and interpreted in Fortran IV
- syntactico-semantical analysis, using morphology, also written in Q-Systems; this component uses FGD as its theoretical starting point
- generation, also using Q-Systems
- morphological synthesis, using Pascal

Q-Systems

Q-Systems (sometimes also Systems Q) – Q stands for "Quebec" – are a tool for machine translation, developed at Montreal University by Alain Colmerauer, also the creator of Prolog.[3]

Q-Systems are similarly declarative as Prolog. This means the author can focus more on the *result* than the *order* of analysis. If there is any ambiguity, all the possibilities are explored *in parallel* (this is how Q-Systems differ from, for example, Prolog).

---

[3] See COLMERAUER, 1970.

In theory, this could make writing lexical rules easier and resulting in simplier rules; in reality, the resulting rules are quite unreadable, as will be seen later.

Q-Systems are not very widely used or widely worked with. One of the reasons might be the fact that all documentation is in French.

Dictionary

RUSLAN uses a Czech-to-Russian dictionary, written by hand in afforementioned Q-Systems. Dictionary item looks like this:

```
DLOUH==M(RS(+(*INT)),MI2289,DLINNYJ).
DLOUH==M(RS(-(*INT)),MI2276,DOLGIJ).
```

This describes two possibilites of the translation of the word “dlouhý” to Russian: the first is “длинный” and the second is “долгий”. They differ by the semantic feature INT they require or forbid from the word they depend on.

More complex dictionary item looks like this:

```
C3ES3TIN
  ==Z(@(*A), MIO109, $(JAZYK),
     2(POS, #($), &, $(MI28), $(C2ES2KIJ),
       1(=,@($), #($),$($))),
     1(=,@($),#($),$($))).
```

This item translates the word “čeština” to Russian words “чешский язык” and also describes their relationship.

Maybe because memory was more expensive than today, all dictionary items are used without any comments, leaving only very difficult-to-decypher rules.

Analysis

The rules for analysis are even less readable. Random example of two such rules are as follows:

```
1(B*, X*1, /, X*2, F*1(C*), X*3, /, X*4, @(V*), X*5, %(X*),
 I(*), 1(X*6, $($)), X*7)


1Z(A*9), (Z*2)
  == 1(D*, /, X*2, F*1/X*),/,@(V*),X*5,%(X*),1*,1(X*6,$($)),X*7,
       A*B,
    5(U*1, @(U*2), U*3, $(U*), 3(E*(Y*1), B*(C*), W*1, W*3, %(X*),
  $(W*)),
+1Z(A*9, Z*2)
 / -NON- (, + -DANS- X*9 -ET- +(V*) -HORS- X*9, +(VZT)
  -ET- -(V*) -HORS- X*9, *
  -ET- C* = S
  -ET- X*3,* -HORS- /,N(S), S(S), D(S), A(S), L(S), I(S)
  -ET- / -HORS- X*2, 2
  -ET- (, Y%1 = -NUL-
      -OU- E*(Y*1) -HORS- U*2,*
      -OU- E*(+(V*, *)) -HORS- U*2, *
      -OU- -NON- E*(-(V*)) -HORS- U*2, * .)
  -ET- (. E*(Y*1) *N
      -OU- H(B*(C*)) -DANS- U*1 .) .
```

Those are all left with next to no comments. For example, the only comment for the group of more than 20 rules, including the two rules above, is `RELATIVE CLAUSES ADJOINED TO THEIR HEADS`.

2.2.2 Česílko 1.0

"Česílko" is a name for two entirely different machine translation systems with slightly different goals and, more importantly, slightly different structure. Both were originally intended for Czech-to-Slovak translation.

Česílko 1.0 was a system, developed in 2000, and was aimed for direct translation between Czech and Slovak and intended to assist a translation memory[4]. The translation works lemma-by-lemma in a following fashion:

- morphological analysis of source (Czech) language
- disambiguation
- direct translation, lemma-by-lemma
- morphological synthesis

The system is written in a mixture of C, C++ and Flex (fast lexical analyser generator). The code itself is not really well documented and modular, but that can be attributed to the age of some of the components – despite the whole system being developed in 2000, some files seem to be as old as 1991.

---

[4] See HAJIČ; HRIC; KUBOŇ, 2000.

2.2.3 Česílko 2.0

Česílko 2.0 is a different project with similar goals, but using different frameworks and adding more transfer rules[5].

The system works in a following fashion:

- **non-deterministic** morphological analysis of source Czech language
- translation of lemmas
- applying transfer rules by changing syntactic tree
- morphological synthesis
- ranking of all the generated sentences

Unlike Česílko 1.0, Česílko 2.0 uses a non-deterministic parser and explores all the possibilities in parallel.

Česílko 2.0 uses more advanced and more clearly defined tranfer rules. This advanced transfer would, in an ideal world, make the system more modular and extensionable for our purposes.

Česílko 2.0 is written in the language Objective-C. Because Objective-C might not be known to the reader, I will describe it a slightly more detailed manner.

Objective-C

Objective-C is a very simple and elegant extension of C language, developed by Brad Cox in 1980s by adding Smalltalk features to C[6].

Objective-C is, in my opinion, very easy to learn and understand, at least compared to C++, its more popular counterpart.

Objective-C is not a proprietary language and is possible to compile with either gcc or Clang/LLVM compilers. However, what is proprietary is its most used standard library, Cocoa. I will describe it here, since it will be important in further sections.

Cocoa

When Steve Jobs left Apple, he made a smaller company called NeXT. Among other things, they produced a proprietary operating system called NeXTSTEP, based on Unix.[7]

---

[5] See HOMOLA; KUBOŇ; VIČIČ, 2009

[6] See HILLEGASS; PREBLE, 2011

[7] For a more detailed history, see `https://developer.apple.com/legacy/library/documentation/Cocoa/Conceptual/CocoaFundamentals/WhatIsCocoa/WhatIsCocoa.html#//apple_ref/doc/uid/TP40002974-CH3-SW12`.

This operating system used Objective-C as its standard language, and proprietary libraries, called OpenStep.[8]

Several years later, Apple (now merged with NeXT) made its new version of Mac OS, called Mac OS X; this operating system was partially based on NeXTSTEP and also used some of its proprietary libraries, now renamed Cocoa.[9]

Cocoa is not the only library for Objective-C, but because Apple is the main investor in Objective-C-based systems, it's a de-facto standard library. Cocoa is nowadays found in every Mac PC, iPhone and iPad and maybe other Apple's products.

2.2.4 PC Translator

PC Translator is a commercial translation system from a Czech company LangSoft (`http://www.langsoft.cz/translator.htm`). PC Translator works with several language pairs, all with Czech on either source or target side.

Authors of PC Translator don't publish any papers or other literature about the system – what can we tell about its functionality is gathered only from its promotional website and from the experiments with the software itself.

PC translator seems to be purely rule-based. The system seems to work in following steps:

- some (probably rule-based) morphological analysis of the source language
- translation of the lemmas from source language to target language by searching in a large dictionary
- some synthesis of morphological information and the translated lemma

The system doesn't seem to do any kind of reordering. It also doesn't seem to do any analysis on a deeper level, like sentence constituents. Some of the phrases in the dictionary are longer than one word, but most of them seem to be one-word only.

One of the advantages of PC Translator is its large dictionary – however, the dictionary is sometimes choosing very odd and inprobable choices when disambiguating between more possible translations. For example, the English sentence "I like dogs" is translated as "Mám rád kleště", because the term "dog" can be also translated as "kleště"[10]. This can be seen as a proof that PC Translator is a purely rule-based system.

---

[8] Despite the name, OpenStep is not open source – the Open allude to the fact that its API specification was open.

[9] The kernel of Mac OS X is open source, as is its "underlying" operating system called Darwin – however, this system does not contain Cocoa libraries.

[10] from Collins' Dictionary: "dog – 5. a mechanical device for gripping or holding, esp one of the axial slots by which gear wheels or shafts are engaged to transmit torque"

According to its marketing materials, PC Translator v14 uses a Czech-Russian dictionary with above 650.000 words.

2.3 Statistical systems

2.3.1 Google Translate

Google Translate is a popular free online translation service by Google, an American web search giant (`http://translate.google.com`). Although Google is producing many academic papers on machine translation, the whole system is still proprietary and we cannot fully inspect it, as in the case of PC Translator, and we can only state our conjectures.

According to Google's own papers[11], Google Translate uses mostly statistical approach to machine translation.

However, because of its purely statistical approach, it either needs huge amounts of data for every language pair, or it needs to use so-called "pivot languages"[12] – in the case of Google Translate, it's usually English; specific English word order and English idioms are then re-translated into the target language and sometimes introduce downright wrong translations.

API

Google Translate, apart from being a website, has a paid translation API[13]. The API is a REST-based API which returns the translation in standard JSON; however, it also needs fairly complicated OAuth authentication.

Some unofficial libraries remove this complexity and abstract it away from the user. One of them is called prosaically "google-api-translate-java" (`https://code.google.com/p/google-api-translate-java/`) and is, not very unexpectedtedly, Java-based.

2.3.2 Bing Translator

Another available online translation service is Microsoft Translator / Bing Translator. (In Microsoft's own materials, the system is usually called Bing Translator when referring to the website and Microsoft Translator when referring to the API, however it's not very consistent. I will call the whole system Bing Translator, even when referring to the API that's called just "Microsoft Translator" in the documentation.)

---

[11] for example OCH, 2005 – F. J. Och is a head of Google Translate group in Google

[12] See for example KOEHN, 2010

[13] `https://developers.google.com/translate/?hl=en`

Bing Translator is very similar to Google Translate – it is an online website with an easy GUI and an additional paid API. Again, the team occasionally publishes some scientific papers, but the system is again proprietary as a whole.

In separate experiments between English and German, I found out that for some language pairs, Bing Translator does more rule-based-looking post-editation. However the system as a whole seems statistical, similarly to Google Translate.

API

Again, Microsoft offers paid Bing Translator API (confusingly marketed as a "dataset" inside Windows Azure platform).

The API is slightly more complex than Google's API because of the auto-expiring token, but Microsoft itself offers some abstracting code as an example in its documentation[14] in C# and PHP.

2.3.3 Yandex Translate

Yandex (`http://www.yandex.ru`) is a Russian search portal that, according to its website[15], generates 61 percent of web search traffic in Russia.

Apart from being a search engine, Yandex offers a variety of other services. One of them is Yandex Translate (`http://translate.yandex.com`)[16] – again, a simple website for automatized translations, similar to aforementioned Google Translate or Bing Translator.

API

Yandex Translate also has a translation API. The API itself is absolutely free, unlike the other two translation systems, and is probably the easiest of the three online services to implement; however, it has strange and vaguely defined usage limits with no way of checking the actual usage.

2.3.4 Moses

Moses is an open-source machine translation toolkit with GPL licence, developed as a successor to a closed-source Pharaoh system. See for example KOEHN; HOANG; BIRCH, et al., 2007 or MOSES MT, 2013.

---

[14] `http://msdn.microsoft.com/en-us/library/hh454950.aspx`

[15] `http://company.yandex.com/`

[16] Or `http://translate.yandex.ru` for Russian version

The system is very modular and very customizable, which makes it a bit harder to describe. What makes it also harder is that the term “Moses” is usually applied for both the “core” Moses decoder and the phrase extractor, and the whole toolkit that’s bundled with it. I will try to describe it from the point of view that’s relevant to the task of this thesis and write only about the modules that I have actually used, and about my Moses use-case in general.

Pipeline overview

In a very broad view on Moses pipeline, we have some corpus of texts, either parallel or monolingual, and we want to somehow learn a *model* for the translation task. We can then use this model for translating any other sentences in the source language.

This is still a fairly broad definition. For our purposes, let’s assume we have a bilingual corpora of a given language pair and a different, usually bigger, monolingual corpora of the target language. We can then learn *translation model* from the bilignual corpora, which is responsible for the “precision” of the translation; and then *language model*, responsible for “fluency”. The actual translation is then “combining” those two factors.

The translation model is called *phrase-based*, because it contains whole phrases, and it contains probabilities of their possible translation, inferred from the corpus. Similarly, language model contains probabilities of various word n-grams.

Now we can look a little closer to what is actually hapenning and what are the actual needed steps.

The bilingual corpora have to be first prepared by aligning the sentences, so every sentence has exactly one translation. (Almost every corpus, available online, is already sentence-aligned.)

The sentences are then word-aligned, which means pairing words to their translations. I am using MGIZA++ (GAO; VOGEL, 2008). From this word alignment, Moses learns a so-called *phrase-based translation model.* From the monolingual corpora, a statistical *language model* is learned – using, for example, SRILM language model (STOLCKE; ZHENG; WANG; ABRASH, 2011).

Moses is then used for so-called *decoding* of the information from both the language model and the translation model, which choses the best possible translation, using algorithms like beam-search.

However, for the best translation, we need to tune Moses parameters for optimal results. This is done using so-called *minimum rate error rating* – or MERT for short, which is tuning the parameters on a small separate development set.

After MERT tuning, we finally have working language model, translation model and Moses parameters, which is our complete translation system.

To reiterate, I am using following Moses pipeline:

- getting sentence-aligned parallel corpus, plus bigger monolingual corporus
- word-alignment on parallel corpus
- creating phrase-based translation model
- creating language model
- tuning the parameters for Moses decoder

Managing experiments

The crucial part of Moses is its decoder and phrase extractor. However, we also need some overaching system for managing all the described steps (model training, etc.) – steps variously fail, don't compile, don't fit in memory, etc. We would also like to reuse partial results in more experiments.

Moses itself has built-in perl-based experiment managment system, called prosaically Experiment Management System (EMS). However, this system is not very widely used on UFAL and I decided to not use it.

Instead of EMS, on ÚFAL, another perl-based tool called eman (experiment manager) is used. Eman is described well in BOJAR; TAMCHYNA, 2013 or at its website, `http://ufal.mff.cuni.cz/eman`.

Eman breaks down experiment into so-called "steps". Step encapsulates an atomic part of an experiment and can be in one of a few various states. More importantly, step can be dependent on various other states; if a step fails, all steps dependent on it automatically fail. The whole experiment is then just another step, dependent on all the necessary substeps.

Step is represented by a directory in a playground directory. Step is created by copying a script, called "seed", from a library of seeds, to a new directory.

Word alignment

For word alignment, I am using MGIZA++[17], which is a GPL toolkit based on GIZA++[18], which is itself based on models, sometimes called IBM Model 1 to IBM Model 5[19], which are themselves based on expectation–maximization algorithm (EM).

IBM Models and the underlying EM algorithms are explained perfectly in Chapter 4 of KOEHN, 2010 or in those slides by the same author – `http://www.inf.ed.`

---

[17] See GAO; VOGEL, 2008

[18] See OCH; NEY, 2003

[19] See BROWN; PIETRA; PIETRA; MERCER, 1993

`ac.uk/teaching/courses/mt/lectures/ibm-model1.pdf`.

GIZA++ is an implementation of those models. MGIZA++ is just its multi-threaded variant, which makes the word alignment slightly faster.

Phrase-extraction

In this step, Moses takes the word alignment from the previous step and learns a so-called "phrase table". Unlike word alignment, phrase extraction spans multiple words on every side in so-called "phrases".

Phrase table consists of list of phrases, their probabilities in both ways of translation, and their lexical weighting – lexical weighting is the probability of the translated phrase counted by individual word pairs. The exact meaning of the numbers is well explained in KOEHN; OCH; MARCU, 2003.

The phrase-table defines a so-called "translation model".

Language model

Language model is a part of the system, that tries to model the probability of a target language sentence alone. It's trained on a monolingual corpus.

I am using SRILM, which is an open source language modeling toolkit. (Although it's open-source, it uses its own license, that allows free use only for non-commercial and educational purposes.) Current status of SRILM is described in STOLCKE; ZHENG; WANG; ABRASH, 2011, original design is described in STOLCKE, 2002.

SRILM uses several models, one of them is n-gram word model, described well in KOEHN, 2010[20]. I use n-grams model to the order 3 with words and order 5 with tags (see section 4.4.2). I smooth the models with Kreser-Ney smoothing with Chen and Goodman's modification[21].

Language model interpolation

If we have more than one monolingual corpora (as I do, as described in 3.2), but we are not sure how helpful each of them are, we can use so-called interpolation (also called mixing).

Linear interpolation in general is described for example in GUTKIN, 2000. On a separate heldout data, set of *lambdas* are trained – the resulting probabilities are then just the individual probabilities, multiplied by the lambdas and summed.

---

[20] chapter 7

[21] See CHEN; GOODMAN, 1996 and `http://www.speech.sri.com/projects/srilm/manpages/ngram-discount.7.html`

Linear interpolation is supported by Moses by undocumented script in the codebase, called `interpolate-lm.perl`, which in turn uses SRILM's undocumented AWK script `compute-best-mix.gawk` and SRILM's `ngram` with `-mix-lm` option[22]. Eman manager then uses these scripts in the `mixlm` seed.

Factored translation

The pipeline, described in the previous sections, translates phrases from the source language to the target language "as is". Only the exact phrases, found on the source side, can be translated to the exact phrases on the target side; and as they are decoded by Moses, only the phrases themselves are taken into account.

However, with morphologically rich languages such as Russian or Czech, this can result in worse translations because of the number of word forms and resulting data sparsity. With so-called factored translation, we can add some morphological information while still keeping the main ideas of phrase-based translation. Factored translation was introduced in KOEHN; HOANG, 2007.

With factored translation, phrased-based approach is extended with morphological (or other) information[23]. We can add additional information (for example, lemma or morphological tag) to either side of the translation, on a word level – this is called a *factor*. Then, instead of training language models and/or translation models on the words alone, we train them on some combination of these factors and then, with the help of Moses that supports factored translation models, combine them together.

Recasing

If the language and translation models are all trained on lowercased corpora (like ours are), we need to train a recaser that will convert the translated text from lower case back to upper case.

We could make a rule-based recaser, such as the ones that are included in Moses; however, we can also train a statistical recaser. The recaser is basically a complete Moses model, trained as a translation from lower-cased corpus to a cased corpus, where any (case-sensitive) monolingual corpus can serve as a source for the language model – where source language is the lowercased corpus and the target language is the original corpus.

---

[22] See `http://www.speech.sri.com/projects/srilm/manpages/ngram.1.html`

[23] Paraphrased from KOEHN; HOANG, 2007. The exact quote is "Therefore, we extended the phrase-based approach to statistical translation to tightly integrate additional information."

2.4 Hybrid systems

2.4.1 TectoMT

TectoMT is a translation system, developed almost exclusively at ÚFAL (`http://ufal.mff.cuni.cz/tectomt`; ŽABOKRTSKÝ; PTÁČEK; PAJAS, 2008). While it's based on linguistically motivated theory (unlike "pure" statistical systems), some of its individual parts are based on statistical approaches (unlike "pure" rule-based systems) – therefore, I think it's appropriate to put it somewhere in the middle on the axis from the section 2.1. Similarly to Moses in section 2.3.4, I will try to describe the general structure of the system, but only as relevant to our experiments.

TectoMT is available for a download from UFAL's public SVN repository, with the instructions on UFAL's public wiki (`https://wiki.ufal.ms.mff.cuni.cz/external:tectomt:tutorial`). Even more than Moses, TectoMT is an experimental software for academic usage with constant changes from many contributors – and for that reason it takes a while to learn to use it.

Treex

TectoMT is built on the Treex platform, which used to be developed together with TectoMT under the same name, but later branched out as its own project and is nowadays used for other applications (Depfix, HamleDT). (`http://ufal.mff.cuni.cz/treex`). Still, because of the long coupled development, Treex source code and its inner structures are based on the needs of TectoMT, and even today it's sometimes difficult to say where exactly the framework ends and application begins. For example, while Treex is downloadable from CPAN perl repository, the version on CPAN is outdated and doesn't work with TectoMT; all TectoMT blocks exist under Treex package; the only way to get newest Treex sources is to install the whole TectoMT framework.

Treex and TectoMT are free software. Treex is dual-licensed under Artistic License 1.0 and GPLv2, as most CPAN packages are. TectoMT is licensed under GPLv2 outright. However, there are modules in TectoMT with more restrictive licencing – some of them can be used only non-commercially – and some models are trained from non-free sources and probably couldn't be used outside of academia.

Trees and layers

Ultimatively, TectoMT is based on a linguist theory that predates machine translation by decades.

The Functional Generative Description theory comes from Prague's Linguist Circle (and its older theories), and was described for example in SGALL, 1967 (in Czech) or SGALL, 1969 (in English). It describes a system of various layers of description and the system of their representation and composition, where the layers are (from the "deepest" level) tectogrammatical, phenogrammatical, morphemic, morphophonemic and phonetical[24]. The concept of tectogrammar was first introduced in CURRY, 1961.

Some of the layers would use dependency trees, which are inspired by Czech *sentence analysis* (described for example in ŠMILAUER, 1958).

While Functional Generative Description is a theory, Prague Dependency Treebank project[25] is an application of this theory on actual data. Its data format and software tools are used directly in TectoMT.

PDT uses several layers, with an inspiration from FGD theory. However, instead of the many FGD layers, PDT uses the following ones:

- w-layer (word layer) for segmented words
- m-layer (morphological layer), where every word has been transformed into a combination of lemma and tag; there is still no relation between words and the structure is still "flat"
- a-layer (analytical layer), where the sentence is tranformed into a dependency tree. The edges represent constituent dependency (or some other relation) and are marked with one of 28 analytical functions (also called *afun*)[26]
- t-layer (tectogrammatical layer), that tries to express semantic structure of a sentence, again with a dependency tree. Nodes on this layer sometimes correspond to nodes on a-layer, but sometimes some artificial nodes are added and, on the other hand, auxiliary words are removed. In addition to *t-lemma* (corresponding with morphological lemma), each node has a *functor*, that tries to somehow convey a semantic function of a relation to node's head (for example, `AIM` as adjunct expressing purpose). Semantic morphological categories are represented by *grammatemes* (for example, `number=sg` for singular).

TectoMT uses the described PDT layer logic. The idea of TectoMT is to first convert source sentences through all the layers to t-layer (*analysis*), translating the t-layer to the target language (*transfer*) and converting it back to full sentences (*synthesis*).

---

[24] SGALL, 1969, page 26.

[25] BEJČEK; HAJIČOVÁ; HAJIČ, et al., 2013, the latest description in BEJČEK; PANEVOVÁ; POPELKA, et al., 2012 and more detailed in HAJIČ; PANEVOVÁ; BURÁŇOVÁ; UREŠOVÁ; BÉMOVÁ, 1999, ZEMAN; HANA; HANOVÁ, et al., 2005 and MIKULOVÁ; BÉMOVÁ; HAJIČ, et al., 2005

[26] Technically, the edge is not marked with the function, only the dependent node.

However, TectoMT modifies the PDT model with the addition of *formemes* (described for example in ŽABOKRTSKÝ, 2010). Formemes are added to nodes on t-layer, and represent "in which morphosyntactic form the t-node was (in the case of analysis) or will be (in the case of synthesis) expressed in the surface sentence shape"[27].

Theoretically, they should be seen as something "between" the t-layer and the layers above. Example formeme is `n:since+X` for English expression of time, translatable as `n:od+2` for Czech (2 for genitive)[28].

Formemes are technically not "correct" according to FGD description and they shouldn't be needed for analysis or synthesis, and in the tranfer phase, we should be able to just transfer the functors.

However, the motivation for formemes (at least my understanding of it) is, that *we are not that far*, "pure" semantic translation is not possible with our tools, and it's better to transfer t-lemmas, formemes and grammatemes, and generate the more surface layers from that.

Getting back to Russian language, with respect to PDT layers – while some preliminar research has been done into representing Russian according to the described model (MAREČEK; KLJUEVA, 2009), the application of PDT theory into Russian is far from complete.

Blocks

Every TectoMT task can be decomposed into so-called *blocks*. To quote ŽABOKRTSKÝ, 2010: "The basic processing units are blocks. They serve for some very limited, well defined, and often linguistically interpretable tasks (e.g., tokenization, tagging, parsing)." In other words, block is given a tree as an input and the block then does some well-defined task on it.

More blocks in a row are called a *scenario*.

Blocks can be language specific or language agnostic; simple blocks like copying a tree are usually language agnostic, as well as other parametrizable blocks – however, most blocks are usually language specific.

Because of language specificity of most of the blocks, scenarios for various language pairs are language specific too. For example – while it's not up-to-date, English-to-Czech scenario is thoroughly described in ŽABOKRTSKÝ, 2010.

---

[27] ŽABOKRTSKÝ, 2010

[28] Example from ŽABOKRTSKÝ, 2010

Makefiles

TectoMT package itself also contains various scripts and tools – one of them are Make scripts for easier running and evaluation of the scenarios[29]. The goals of those scripts are different from Moses evaluation managers, described in 2.3.4, as well as it means, but we can imagine it as being slightly similar. (Those Make scripts don't, unfortunately, have any nice name to refer to them as.)

MaxEnt, HMTM

Interesting techniques of using Hidden Markov Tree Models and Maximum Entropy models for TectoMT translation are described in ŽABOKRTSKỲ; POPEL, 2009 and MAREČEK; POPEL; ŽABOKRTSKÝ, 2010. Unfortunately however, I haven't been able to train any of those for Czech-to-Russian translation.

Current Czech-to-Russian scenario

Unlike Moses where I had to had to train whole models from zero, TectoMT already had some rudimentary scenario for Czech-to-Russian machine translation.

From what I heard from colleagues, this scenario was put together in a very short timeframe (about 24 hours), and it's not described in any paper or any other documentation. The scenario consists of 66 blocks, where Czech analysis is 33 blocks, transfer is 5 blocks and Russian analysis is 28 blocks.

While I have to admit I don't have enough experience to judge it, Czech analysis seems to be well done and probably copied from some other working scenario.

However, the transfer and Russian synthesis doesn't seem to be very advanced. Transfer of "t-lemmas" actually only involves look-up in a very small dictionary, made by taking the proprietary PC-Translator dictionary (section 2.2.4) and taking a subset of its lexical lemmas that also appear in UMC (section 3.1.4). Unlike more advanced models, in this model, any given lemma is always translated the same way, no matter the context. Similarly, transfer of formemes is just a copy and a few hand-written rules.

With regards to Russian synthesis – about half of its blocks seem to be Czech synthesis blocks. This can be somehow justified by the similarity of the two languages, but it also doesn't exactly inspire confidence.

In general, it's obvious the scenario was done very quickly and, to me, it's quite surprising it's even translating something.[30]

---

[29] What needs to be said is that's it's tailored mostly for UFAL's cluster infrastructure instead of for general usage.

[30] Although, as will be seen in the chapter 5, the results are not that good.

3. Data

In this chapter, I am describing the datasets that I used for my experiments.

3.1 Parallel data

3.1.1 WMT test sets

Two of my sets are WMT test data – WMT 2012 and WMT 2013.

WMT (short for Workshop on Statistical Machine Translation) is, as the name suggests, an annual workshop about statistical machine translation. One of the recurring activities is *shared translation task*, where various teams compete on translation of a shared test data, with a given set of languages. (See for example BOJAR; BUCK; CALLISON-BURCH, et al., 2013, or the rich history on `http://www.statmt.org`.)

In 2012 and 2013, for all the available languages, one multi-lingual parallel testset was created. As noted in BOJAR; BUCK; CALLISON-BURCH, et al., 2013, in 2013, Russian was added as one of the languages.

In the year 2012, Russian was not one of the languages. However, in the year 2013, WMT released data called `news-test2012` which *does* retroactively include Russian, additionally to other languages from the year 2012, so I decided to use that, too.

The sentences in the training set are manually translated; for the year 2013, the set is described in BOJAR; BUCK; CALLISON-BURCH, et al., 2013 and available on `http://www.statmt.org/wmt13/translation-task.html`. For each of the languages, a fixed number of (different) sentences is taken and then translated to all the other languages.[1]

Therefore, most Czech and Russian sentences in this set are not a direct translation of each other, but they are different translations of the same source sentences from various languages – except for sentences that are originally from either Czech or Russian sources.

It can be argued, that because the Czech and Russian side are translated separatedly from different languages, the advantage of similarity of the two languages is lost – different idoms and different word order will be used. However, if I used only the directly translated sentences, the data would be significantly smaller.

To reiterate – I extracted the Czech and Russian sentences from WMT 2013 test set and from WMT 2012 test set.

---

[1] In the year 2014, the testset was created slightly differently and I could not extract Czech-Russian pair from it.

3.1.2 Intercorp

Intercorp[2] is a parallel corpus for many language pairs, each including Czech. The history and other information is thoroughly described in ČERMÁK; ROSEN, 2012. One of the language pairs in Intercorp is Czech-Russían.

I am using two separate Intercorp corpora for technical reasons.

Mixed Intercorp

The first Intercorp corpus was used for some of the Moses models (section 4.4.3) and was created by my colleague Natalia Klyueva. The source data are both from direct Russian-Czech translation and translations from third language, and the sources are not marked clearly.

In this thesis I will call the corpus "Mixed Intercorp", because all the various sources are mixed together.

Original Intercorp

Because I wanted some additional data for testing purposes, I decided to ask for more data from Intercorp. I was given access to "raw" Intercorp data (by Institute of the Czech National Corpus) for non-commercial, academic purposes.

The data itself is organized by source, and each data source is given an information of the original language; even in the Czech-Russian part of Intercorp, there are texts with an English source (for example, Czech and Russian translation of Harry Potter novels).

The data are in a strange, XML-like format, that's apparently used by a Manatee corpus management system.[3]

Filtered Intercorp

I was able to extract just the data, that are either direct translations from Russian to Czech or vice versa, thanks to the metadata in the corpus.

To have a separate set, I removed the data already present in the "mixed" Intercorp.

The resulting data is purely from fictional novels, except for Jiří Levý's Art of Translation, which is (as the name suggests) a translation theory book.

---

[2] Stylized as InterCorp in some materials.

[3] `https://www.sketchengine.co.uk/documentation/wiki/SkE/PreparingCorpusOverview`

Author	**English name**	**Year**	**Sent.**
Nikolai Ostrovsky	How the Steel Was Tempered	1936	9844
Ilya Ilf, Yevgeni Petrov	The Twelve Chairs	1928	8525
Mikhail Bulgakov	The Master and Margarita	1967	7124
Nikolai Nikolaevich Nosov	The Adventures of Neznaika and His Friends	1953-1954	3523
Jiří Levý	The Art of Translation	1957	3149
Aleksandr Solzhenitsyn	One Day in the Life of Ivan Denisovich	1962	3090
Alexander Pushkin	The Captain's Daughter	1863	2984
Aleksandr Solzhenitsyn	An Incident at Krechetovka Station	1963	1467
Aleksandr Solzhenitsyn	Matryona's Place	1963	880

Table 7: Filtered Intercorp data

All of the data are translations from Russian to Czech, except, again, Jiří Levý's Art of Translation.

All the used novels are in the Table 7, sorted by the sentenced count. (English transcriptions, English title translations and years of publication are taken from English Wikipedia.)

3.1.3 Subtitles

Another set of data that I used were subtitles from OpenSubtitles database.

FilmTit

In a separate FilmTit project (BÍLEK; ČECH; DAIBER, et al., 2012), me and my colleagues tried to make a project for subtitle translation from English to Czech, working simultaneously as a translation memory and a machine translation system.

OpenSubtitles

For that project, we were given access to the set of subtitles from the server OpenSubtitles (`http://opensubtitles.org`).

This dataset was, however, not a pair of aligned sentences. It was not even a pair of aligned *files*; we were given just a set of SRT files, and a table which paired each of those files with a movie (identified by IMDB number) – each movie usually has more subtitle files, and there are usually more errors in the data.

Subtitle files have the sentences paired with timestamps. (We described the format more thoroughly in BÍLEK; ČECH; DAIBER, et al., 2012.)

From a set of SRT files paired with a movie, we selected just one Czech and just one English SRT file which we found most similar, based on the timestamps. From the pair of the files, we then extracted the sentences that have the most similar timestamps.

Tolerance

These two pairings – pairings of subtitle files and pairing of the actual lines – are non-trivial tasks, and require a "tolerance" – how different can the time marks of a sentence be to be still paired together.

Higher tolerance produces bigger corpus with more errors, while lower tolerance produces smaller, but more correct corpus.

When experimenting on the FilmTit project (as, again, described more thoroughly in BÍLEK; ČECH; DAIBER, et al., 2012), we found out, that the best results (tested both on another movie subtitles and on a different corpus) are – without exception – with *bigger corpus* and *higher tolerance*. Even when that introduced a lot of incorrect sentence pairs, the overall results were still better with the biggest possible corpus.

Czech-Russian subtitles

I asked OpenSubtitle maintainers, again, for another set of data, this time with Czech and Russian. Because it contains only movies, that have both Czech and Russian subtitles, the set was much smaller than with English and Czech. I was still able to use the same algorithms from FilmTit to build a parallel corpora, since the raw files had essentially the same format.

I again used the highest possible tolerance, and therefore surely introduced a lot of errors. Unfortunately, for a lack of time, I was't able to replicate the experiments for the ideal tolerance here. However, I hope that the results would be similar than in English-to-Czech translation – that is, the biggest possible corpus will result in the best translation.

3.1.4 UMC corpus

UMC (ÚFAL Multilingual Corpus) is a Czech-English-Russian corpus (see KLYUEVA; BOJAR, 2008[4]). The data were given to UMC creators by Project Syndicate, Prague-based non-profit news organization, translating news and opinions from around the world.

UMC has two versions – UMC 0.3 and UMC 0.1. UMC 0.3 is then strangely divided into *all*, *test* and *devel* – however, the parts are strangely mixed together and (only in some files) lowercased.

I decided to use UMC 0.1 corpus as "umc-train", and from 0.3 I use the test part as "umc-test" and the devel part as "umc-devel"[5].

---

[4] This paper talks about UMC 0.1. I haven't been able to find any paper about 0.3, but there is some information on the website – `http://ufal.mff.cuni.cz/legacy/umc/cer/`

[5] However, the texts in the folders test and devel was actually lowercased, so I had to take the data

3.1.5 Wiki titles

I also extracted all of the titles from the Czech and Russian Wikipedia, that correspond to each other.[6] Wikimedia Foundation (parent organization of Wikipedia) produces complete dumps of Wikipedia in XML; I used one of those dumps and derived pairs of Czech and Russian titles that are translations of each other.[7]

Those are usually only noun phrases and the main word is usually in nominative singular, so the morphology isn't that rich – however, I hoped that the model will learn some phrases needed for the translation of the named entities.

3.2 Monolingual Russian data

3.2.1 News Crawl

The largest part of my monolingual data is corpus from WMT workshops that they call "News Crawl".

According to CALLISON-BURCH; KOEHN; MONZ; SCHROEDER, 2009, WMT workshop has been continously crawling web articles since 2007 for making test sets. This allowed them to make a big, randomized corpus from all these sources.

The corpus is categorized by year, and I treat each year as its own corpus for the interpolation (as described in 2.3.4).

3.2.2 Common Crawl

Common Crawl is a publicly available web crawl[8] – `http://commoncrawl.org/`.

As described in SMITH; SAINT-AMAND; PLAMADA, et al., 2013, group of researchers tried to extract parallel data from this web crawl. One of the language pairs was English-Russian and the result is publicly available on WMT site. I used the Russian side of the corpus.

However, the quality of this corpus is very discutable. Because it contains data downloaded from the "raw web", it often has sentences in different languages, sentences in machine-translated Russian, random UTF-8 symbols, random HTML data, some code, and so on.

---

from the all folder.

[6] There is a corpus called *Wiki Headlines* on WMT2013 website, for English and Russian. I am not sure how that got created, but it has nothing to do with my corpus.

[7] Unfortunately, at some point in 2013, Wikipedia changed the way interlanguage links work, so my old script no longer works; however, the new way of saving interlanguage links should be even easier to exploit.

[8] From Wikipedia – "A Web crawler is an Internet bot that systematically browses the World Wide Web, typically for the purpose of Web indexing. "; web crawl is then a result of such a web crawler

Corpus	Lines	Tokens	*per line*	Types
UMC dev	765	11,870	*15.52*	5,764
UMC test	2,000	30,884	*15.44*	11,575
WMT 2013	3,000	48,268	*16.09*	15,255
WMT 2012	3,003	54,569	*18.17*	17,258
Wikinames	114,742	244,539	*2.13*	91,766
InterCorp filtered	37,586	379,432	*10.1*	62,479
InterCorp mixed	148,847	1,595,531	*10.72*	149,052
UMC train	93,395	1,741,892	*18.65*	111,107
Subtitles	2,324,373	11,971,542	*5.15*	333,166

Table 8: Statistics of Czech side of parallel corpora

Corpus	Lines	Tokens	*per line*	Types
UMC dev	765	11,936	*15.6*	5,622
UMC test	2,000	31,884	*15.94*	11,296
WMT 2013	3,000	48,080	*16.03*	15,691
WMT 2012	3,003	53,499	*17.82*	16,473
Wikinames	114,742	253,128	*2.21*	93,168
InterCorp filtered	37,586	367,838	*9.79*	67,666
InterCorp mixed	148,847	1,508,591	*10.14*	144,884
UMC train	93,395	1,750,475	*18.74*	107,756
Subtitles	2,324,373	11,897,564	*5.12*	327,510

Table 9: Statistics of Russian side of parallel corpora

This was one of the reasons why I decided to use linear interpolation as discussed in 2.3.4 – hoping, that the tuning algorithm will automatically “find the right balance” between the language models.

3.2.3 Yandex

Yandex was already described in 2.3.3. Apart from providing free translation API, Yandex also provides an English-Russian parallel corpus (`https://translate.yandex.ru/corpus?lang=en`). I used the Russian part of this corpus as a monolingual corpus.

The version of Yandex corpus that I used was originally lowercased. Since then, Yandex already made a new version with the correct cases; I did not use the new version for any experiments for time constrains.

3.3 Statistics

In the tables 8, 9 and 10, I am presenting some basic statistics about my corpora, sorted by token count.[9]

---

[9] The tokenization for the task of counting tokens and types is very rudimentary and just breaks words on every punctuation mark – in my opinion, it doesn’t matter, since the table is only for orientation anyway. Words were converted to lower-case before type counting.

Corpus	**Lines**	**Tokens**	***per line***	**Types**
News Crawl 2008	38,195	580,308	*15.19*	63,003
News Crawl 2010	47,818	643,363	*13.45*	70,430
News Crawl 2009	91,119	1,315,794	*14.44*	98,901
Common Crawl	878,386	16,837,812	*19.17*	665,385
Yandex	997,000	19,942,195	*20*	694,787
News Crawl 2011	9,945,918	140,041,123	*14.08*	1,569,963
News Crawl 2012	9,789,861	140,914,399	*14.39*	1,450,003

Table 10: Statistics of Russian side of monolingual corpora

3.4 Unused data

3.4.1 Lib.ru

My colleague Natalia Klyueva downloaded in the year 2012 large amount of fiction books from Russian online library `http://lib.ru`.

Unfortunately, I neglected this source and I forgot to include it in any models; I noticed it only at a very late stage and too late for further inclusion in the models described in the section 4.4.3.

4. Experiments

In this chapter, I am describing my experiments with the systems, described in the chapter 2.

In general, my goals were to

- try to run the historical systems on the data,
- automate the black-box systems, so I could at least reliably run them and compare their outputs,
- explore the more open frameworks and try to use them for Czech-to-Russian machine translation and eventually identify possible future work,
- and finally, compare all the systems on the same set of data.

The last goal will be explored in further details in the next chapter.

4.1 Experiments on historical systems

The experiments on historical systems were mostly unsuccessful.

4.1.1 RUSLAN

Dictionary coverage of WebColl

?? RUSLAN dictionary contains about 8,000 lexical items. The domain of the translation and, therefore, the domain of the dictionary itself, was manuals for old computers from 1980's. With my coleague Natalia Klyueva, we decided to try, how much is this dictionary applicable to a current corpus.

In a set of experiments (BÍLEK; KLYUEVA; KUBOŇ, 2013), we tried to measure how many nouns from the RUSLAN dictionary appear at all in a modern text.[1]

For that, we used a monolingual Czech corpus WebColl (SPOUSTOVÁ; SPOUSTA; PECINA, 2010), consisting of roughly 7 million sentences (114 million tokens).

The results are infavourable: from 21,785,622 noun tokens in the corpus, only 7,289,278 are represented in the dictionary.

This means that about two third of nouns would never be translated.

---

[1] The goal of the experiment described in the paper was actually even broader – we tried to work on machine learning and assignment of semantic categories. However, the dictionary coverage was a sub-experiment.

Experiments

Despite the unfavourable results of WebColl coverage (last section) and despite the general un-maintainability of the RUSLAN code (section 2.2.1), I intended to run the system and try it on some test data.

Unfortunately, I have not been able to obtain any version that would even run, let alone translate the thousands of test sentences. Maybe because of the Pascal pre-processing, maybe because of the FORTRAN implementation of Q-Systems interpreter.

Because the coverage is so poor and the domain of the dictionary so outdated, I have decided to not dedicate further time to fixing RUSLAN and investigating the errors.

4.1.2 Česílko 1.0

As described in the section 2.2.2, Česílko 1.0 is not very extendable for Russian as the target language. Extending the system for Russian would mean significant addition to the code, which is not very mantainable by today's standards.

Even then, the code in general assumes that the languages are directly translatable word-for-word. As will be seen in the PC Translator results (chapter 5), word-for-word translation from Czech to Russian doesn't give very good results anyway.

For those reasons, I decided to not extend Česílko 1.0 with Russian.

4.1.3 Česílko 2.0

When Petr Homola was writing Česílko 2.0, he decided to use Cocoa and Objective-C for development. In the section 2.2.3, I tried to describe those two.

However, I wanted to use Česílko 2.0 on Linux environment (for a better replicability). Cocoa is not available on other systems than iOS and Max OS X. For running Česílko 2.0 on Linux, we need another library, called GNUstep – and that creates unpredictable problems.

GNUstep

GNUstep is a free re-implementation of OpenStep/Cocoa. (See `http://www.gnustep.org/`).

Its development started in the NeXTSTEP days; however, it still hasn't met feature parity with Cocoa's OS X.

Aaron Hillegass in 2nd edition of his popular book *Cocoa Programming on Mac OS X* discouraged people from using GNUstep. He redacted this note in later versions of the book, perhaps because of protests from GNUstep developers[2], but in my opinion, his notes are still valid.

GNUstep implementations are very often buggy, not feature-complete with Cocoa and unpredictable. Unfortunately, those bugs are hitting Česílko 2.0.

GNUstep and Česílko

On Mac OS X, Česílko seems to run fine. However, on Linux, where I wanted to run the MT systems (and where only GNUstep is available), GNUstep bugs create unpredictable results.

In my experiments with Czech-to-Slovak translations, I noticed that on Mac OS X, there are about 5-times more sentences generated, than on Linux – while the program was compiled from the same sources.[3]

After thorough inspection, I found out the error was in GNUstep implementation of NSDictionary – Cocoa's implementation of associative array[4]. In some unpredictable cases, NSDictionary returns two different values for two equal NSString keys. It might have to do something with Unicode; however, NSStrings are supposed to be UTF-8 by default.

As a result of this bug, one of the Česílko modules returned *completely wrong* inflection patterns for a number of words; the morphological analyzer then returned only a fraction of the results.

After a "hacky", but working workaround for this issue, the system returned the same correct results on both OS X and Linux. The "hack" involved concatenating a space to the NSStrings – that somehow fixed the issue.[5]

However, I am not at all confident there aren't more similar issues in GNUstep to further develop the system for Russian. I believe findind and fixing the issues of a framework, that's basically copying API of another closed-source library, that's very rarely used in MT projects in the first place, is way beyond the scope of this thesis.

Reading the paper BOJAR; GALUŠČÁKOVÁ; TÝNOVSKÝ, 2011, that presents Česílko 2.0 with a very low BLEU, I think the same issue plagued the authors of that paper – it's unprobable the BLEU of the correctly working system would be that low, especially when compared with HOMOLA; KUBOŇ; VIČIČ, 2009, where the results of Česílko 2.0 were slightly better than of Česílko 1.0.

On a personal note, I must add that I still *do* like Objective-C as a language and

---

[2] `http://www.gnustep.org/resources/BookClarifications.html`

[3] Of course linking to Cocoa instead of GNUstep on Mac OS X.

[4] `https://developer.apple.com/library/mac/documentation/Cocoa/Reference/Foundation/Classes/NSDictionary_Class/Reference/Reference.html`

[5] I want to note, again, that this issue did not appear on OS X/Cocoa, only Linux/GNUstep.

I find its syntax very elegant and, to some degree, self-documenting. However, the poor compatibility of the de-facto standard library with non-Apple systems is making it not very practical.

4.2 PC Translator

I found out it's not easy to automate translating with PC Translator, especially when our goal is to be able to run it from a Linux command-line.

Its GUI is suited for translating by hand, sentence-by-sentence, but not for automated translation of thousands of sentences. Also, by definition, Windows GUI is harder to automate on Linux machine from a script.

However, I was able to work around that, with the help of VMWare Player virtualization software (`http://www.vmware.com/cz/products/player`) and AutoHotkey GUI scripting software, that allows us to emulate screen clicking (`http://www.autohotkey.com/`). My workflow is:

- on Linux machine, encode the source from UTF-8 to windows-friendly encoding
- encode the source as HTML code
- start a virtual machine with PC Translator pre-installed
- on the start of the virtual machine, run AutoHotkey script from an outer-machine folder (thanks to VMWare shared folders and Windows Startup scripts)
- via this AutoHotkey script, run PC Translator and click on "translate file" feature
- translate the HTML file (also shared in the VMWare shared folder)
- turn off the virtual machine
- turn the file back from HTML and Windows encodings back to UTF-8

The HTML part is needed because PC Translator had some problems with translating ordinary text files, plus we can pair the translated sentences better thanks to `id` parameters in `div` tags.

I use the newest version of PC Translator available at the time of writing this, which is PC Translator v14.

4.3 Web translators

All the tests from this section were done around 3rd May, 2014. (I think it's important to note the date of the tests, because the quality of online services might change over time.)

4.3.1 Google Translate

To automate Google Translate, I don't want use the website itself, simply because pasting tens of thousands of lines into a browser window usually crashes the browser and is probably against Google Translate's Terms of Use.

There are some workarounds around this, such as "faking" browser environment using some automation tools and/or libraries and/or using some browser extensions, but I use more stable option, which is the paid Google Translate API, as described in the section 2.3.1.

I figured out using the paid API is not too expensive for testing purposes, so I ended up paying for it, and using it with the library described in 2.3.1.

The cost is measured per character on the source side. I used about 3 million characters and paid about 60 dollars. This is rather high for any repeated experiments, but not that high for a one-time translation.

4.3.2 Bing Translator

With similar reasoning as described in 4.3.1, I decided to use Bing Translator paid API, with the PHP script described in 2.3.2.

The pricing is slightly different in Microsoft Translator than in Google Translate, but in general is slightly cheaper. First 2 million letters are for free, next 2 million are for about 40 US dollars.

4.3.3 Yandex Translate

Yandex Translate API is at the same time easiest and hardest to use from the three web services.

On one hand, its API is trivial and it's trully free to access, with no charges whatsoever.

On the other hand, the API limits are very vague and majorly slow the experiments down. In my experiments, the API simply stopped returning sentences after approximately 1 million characters per 24 hours. After 24 hour waiting period, the API became usable again.

This actually means the experiments have to be regularly stopped for 24 hour "cool-offs", which is very impractical.

4.4 Moses experiments

I trained the whole Moses model from scratch, using data, described in chapter 3. I will try to describe those experiments in this section.

4.4.1 Alternative eman seeds

In my opinion, while eman itself is well written, I found the seeds themselves hard to read, too repetitive, and with large amount of code copied and pasted over.

For that reason, I tried to rewrite the seeds as perl modules instead of bash scripts for more clarity and reusability. I am, however, not personally sure if my effort in this regard was successful. I decided to use the module `MooseX::Declare`[6], which seemed to me at that time like a modern way to write modules in perl.

Unfortunately, that module is using very difficult-to-understand perl concepts and source code transformations through `Devel::Declare`, and as a result, it takes long to run and, perhaps worse, returns very confusing and undecypherable errors. So as a result of my rewrite, I have seeds with code that's probably easier to read and refractor, but on the other hand, it's slow and produces very opaque errors.

Author of `MooseX::Declare` is now recommending `Moops` module instead for declarative syntax; this module is, however, requiring perl version 14 and above, while on UFAL's network, only perl 10 is installed.

My "new" seeds – now residing in `ufal-smt-playground` repository, in the `pm-seeds` directory – are basically copying the functionality of the normal seeds, with some additions (better solution to binarization of the models, support for backoff models). They are compatible with "old" seeds by using small "helper" seeds that just run the perl modules.

However, based on the git repository activity[7], it seems like my new seeds haven't really took traction between other colleagues on ÚFAL; I would guess because of the slow speed and the general complexity of the modules themselves.

---

[6] `http://search.cpan.org/~ether/MooseX-Declare-0.38/lib/MooseX/Declare.pm`
[7] `https://redmine.ms.mff.cuni.cz/projects/ufal-smt-playground/repository`

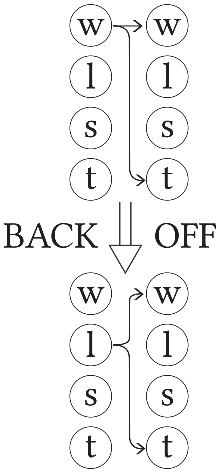

Figure 1: Lemma backoff model

4.4.2 Factored translation experiments

Experiments in this section were done together with my colleague, Natalia Klyueva. I was in charge of the Moses setup, while Natalia Klyueva was recommending me the source of the data, reccomending me the TreeTagger software and the general goal (reducing of OOV rate).

Our first Moses model was essentially as described in the section 2.3.4, without factors.

When we used the model without factors, we realized our Moses results have a high OOV rate[8] – this is easily recognizable by Latin script appearing in Czech-to-Russian translation (or Cyrillics in the opposite direction).

I then tried to compare several set-ups for factored tranlation to get – at the same time – lower OOV rate and higher BLEU scores. I am describing our factored translation experiments in this section, together with the results.

Data source

We used dataset UMC-train for training both language and translation model, UMC-dev for MERT tuning and UMC-test for BLEU testing.

Lemma backoff model – overview

I used a modified version of a set-up described in KOEHN; HADDOW, 2012 as `lemma backoff`. The set-up is illustrated on Figure 1 – where:

- w is for word form
- l is for lemma

---

[8] Out Of Vocabulary; how many words were untranslated due to not being found in the phrase table

- s is for stem (see further)
- t is for tag

Lemma backoff model – details

The primary translation model is from full word on source side to the full word and morphological tag on target side.

The backoff translation model is from lemma on source side to the full word and morphological tag on target side. I do *not* generate words from lemma+tag.

I am then using two language models, one for tags and one for words (both separately interpolated, as described in 2.3.4).

I was not using *interpolated* backoff, simply because regular backoff is easier to use with Moses – for regular backoff, all that is needed is to add `[decoding-graph-backoff]` section in the `moses.ini` configuration file.

Russian tagging

For tagging Russian, I used TreeTagger software (`http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`, also see SCHMID, 1994 and SCHMID, 1995) with a Russian parameter file[9]. TreeTagger is a closed-source software with a restrictive license, but for free for research purposes.

Russian analysis

Unfortunately, we do not have Russian morphological analysis ready. (I am touching on this subject in the section 4.6.1.)

For that reason, I could not generate new forms from lemma+tag – that is why I was translating directly to form+tag.

This has the unfortunate disadvantage that no new forms are created, and all possible forms are taken only from the parallel corpus. Working Russian analysis would probably improve the translation (as also noted in 4.6.1).

Czech lemmatizing

For Czech, I used morphological analyzer Morče (`http://ufal.mff.cuni.cz/morce/references.php`).

---

[9] trained on a corpus created by Serge Sharoff, see `http://corpus.leeds.ac.uk/mocky/`

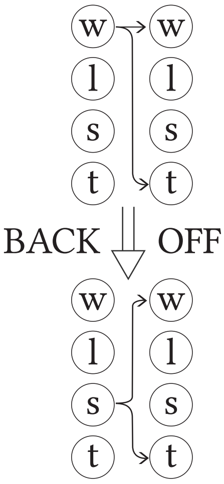

Figure 2: Stem backoff model

Stem backoff model

As another experiment, I tried to take "stems" instead of lemmas. However, instead of lexically motivated stems, I tried *very crude* stems – just using the first n letters of a word. (Separate experiments for n between 3 and 6.)

Surprisingly, this got better results, than linguistically motivates lemmas.

The model is illustrated on Figure 2.

Using stems instead of lemmas is suggested for example in POPOVIĆ; NEY, 2004. However, their stems are more linguistically motivated, while I just very crudely took first few letters.[10]

Results

The results of the described experiments are seen on Figure 3 – *baseline* is original moses with no factors, *1-lemma* and *1-stem* are only the "backoff" models without the main model, and *2-stem* and *2-lemma* are the whole models with backoff.

We can see that stem with length 6 gets the best results. So, I used stem with the length 6 in further Czech-Russian experiments, such as the WMT submission BÍLEK; ZEMAN, 2013 or the full setup described in the next section.

As I already noted – the fact that cutting words instead of using lemmas works better is *surprising*, but it *works*, so that is why I am using it further.

4.4.3 Full setup

My final Moses system uses the setup, described in this section (and section 2.3.4).

---

[10] It's actually debatable if my "stems" can be called stems at all.

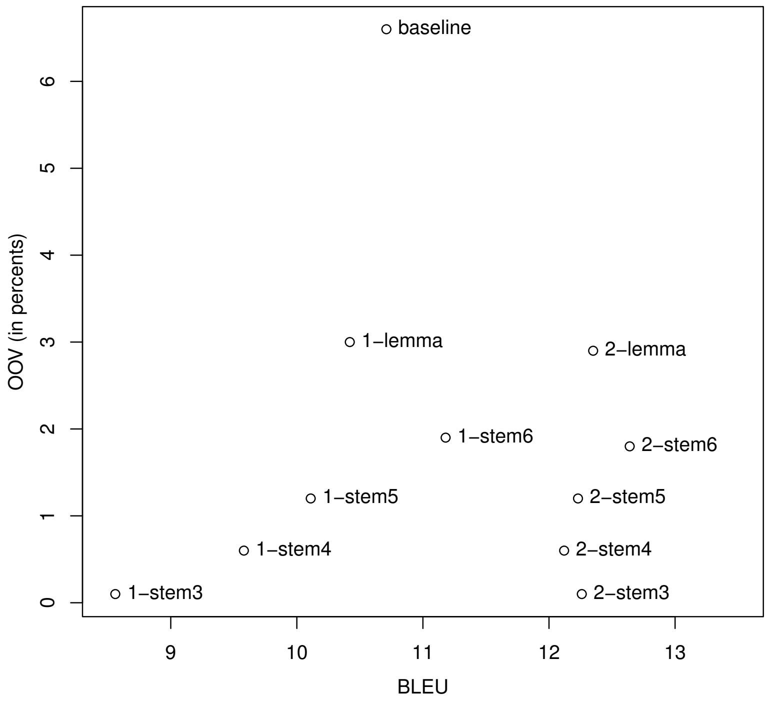

Figure 3: Comparison of various set-ups

Translation model

I use the following parallel corpora, concatenated into one big "eman" corpus, for training a translation model:

- Mixed Intercorp
- Subtitles
- UMC train
- Wikinames

I do not use Filtered Intercorp because I intend it only for testing, as will be described in the chapter 5.

Language model

I use all the monolingual corpora from the section 3.2, plus target sides of the parallel corpora from translation model.

I use linear interpolation, with UMC-dev as a development corpus. I use a linear interpolation instead of log-linear interpolation simply because I didn't notice the option for log-linear interpolation until the model was already built.

Factors, recaser

I use factors as described in the section 4.4.2, with the “crude” stemming. I use recaser, trained as a language model on all the monolingual corpora, as described in the section 2.3.4.

4.5 TectoMT experiments

As I already mentioned in 2.4.1, TectoMT already had a Czech-to-Russian scenario – however, the results were not very good.

I have managed to make the results a bit better by editing TectoMT modules and adding some models. I will describe my changes to TectoMT.

I measured the BLEU changes after each change on “development” set WMT-2012.[11] The original scenario has BLEU 0.0561.

Better morphology

TectoMT’s module for Russian morphology (generation) works as a lookup in a simple, tab-separated file, with the list of lemmas, forms, tags and frequencies. When the lookup in this file fails and the word is not found, the word is used as-is, without any flexis.

The original file seemed to be extracted from Syntagrus corpus, with morphological annotation converted to “PDT-style” tags.

However, this file is still rather small and the coverage could be bigger; from the 53.499 words in the WMT2012 corpus, 7.220 tokens (about 13 percent) wouldn’t be found at all.

I decided to made a larger morphology table by tagging monolingual corpora (3.2) with TreeTagger (4.4.2) and converting the tags from Multext-EAST, that is being used by TreeTagger, to PDT tags, using the tool Interset (ZEMAN, 2010).

In this bigger morphological table, only 1.118 tokens from WMT 2012 couldn’t be found (about 2 percent).

BLEU was increased to 0.0613; this means 0.52 BLEU points increase.

Reflexives

One of the common errors I noticed was reocurring of wrong reflexive `_ся` in the results, usually with wrongly translated verb. (Czech and Russian reflexives are slightly different, as demonstrated in 1.2.1.)

---

[11] I did not want to test the incremental BLEU improvements on the same data as the more “final” testing, described in the chapter 5.

The problem was as follows:

- Czech verb with a reflexive is converted to a tectogrammatical lemma, that has the auxiliary `_se` merged into it
- the static model for lemma transfer (see also the next section) contains only lexical lemmas, not t-lemmas, so it does not have the word with `_se` / `_ся`
- because the static model cannot find the lemma, the transliteration is used
- the transliteration is wrongly translated and still contains the `_ся`

The best, more "long-term" solution would be to train a better Russian transfer model. However, with the help of Martin Popel, we implemented a short-term fix that looks up the verb without the reflexive on Czech side and adds it on Russian side.

This fix increases BLEU to 0.0632; this means 0.19 BLEU points increase from the previous version.

Better transfer

As mentioned in section 2.4.1, the transfer model was originally made in a strange way. The original authors took PC Translator software, then extracted from its internal dictionary a list of lemmas (or, rather, a subset of it) and made an *interset* of this dictionary and all the words in UMC. This produces only a very small dictionary of approximately 13.833 lemmas.

I decided to create a bigger dictionary. Unfortunately, we *do not have a working Russian analysis* (such as parser) and therefore, I couldn't convert Russian corpora to t-lemmas and align them.[12]

Instead, I decided to align "only" the lexical lemmas. Therefore, I aligned lexical lemmas using eman and MGiza++ (2.3.4), TreeTagger for tagging and MorČe for Czech tagging.

From the word-alignment on the lemmas, I took only the interset and exported the lemmas. I also added the whole dictionary, exported from PC Translator.[13]

The result is a dictionary that is 296.447 lemmas big.

This new dictionary (added to TectoMT with the help of Martin Popel) increases the BLEU to 0.0704; this means 0.72 BLEU points increase from the previous version.

---

[12] We do have a SynTagRus; however, as mentioned in MAREČEK; KLJUEVA, 2009, this corpus doesn't have a layer, that would correspond to t-layer.

[13] Public distribution of a system with "directly" copied PC Translator dictionary can be of course problematic, but my understanding of Czech copyright law is that for academical purposes, distribution of such a system should be fine.

Final scenario

Final scenario is the same as the original scenario, with the described additions.

On the development set WMT-2012, the described fixes increased BLEU from 0.561 to 0.704; that is 1.43 BLEU points.

4.6 Future work

4.6.1 Better morphology and parser

In both Moses and TectoMT experiments, I soon run into the same problem: we (on ÚFAL) do not have some any advanced working Russian morphology (both tagging and generation) or Russian parser.

It's true we can take TreeTagger and use it as a black box, as I did in the experiments above. However, some newer and more open system could help us to tune the tagger better.

Working generation would allow us to try better factors (with phrase-based translation), the synthesis in TectoMT would probably return better results.

Working Russian parsing would allow us to train Hidden Tree Markov Models/MaxEnt models and then make better transfer models in TectoMT. It would also allow us to try some automatic post-editing, such as with Depfix system[14].

Martin Popel recommended me to use MorhpoDiTa and either MaltParser or MATE parser. For any future experiments, this should probably be the first step.

4.6.2 Traslation by letters

I decided to try an experiment with Moses phrase translation. Instead of taking words as the primary tokens, I tried to split the sentences on an individual letters and try to learn the models from that.

(I have heard from my colleagues that there is some existing research on this, however, I haven't been able to find it.)

This could, for example, somehow capture the transliteration, and in the case of similar languages (Czech and Slovak, maybe even Czech and Russian) make some reasonable guesses on translations.

Unfortunately, for some reasons, translation models trained on my data got unbearingly big and took more resources – both RAM and disk space – than I could afford on ÚFAL's network.

---

[14] `https://ufal.mff.cuni.cz/depfix`

I stopped with those experiments for a lack of time. However, I still think it would be interesting to investigate those further – at least for really close languages (Czech and Slovak), but even for Czech and Russian.

5. Results

5.1 Overview

I am testing all the following systems:

- PC Translator *(rule-based, 2.2.4, 4.2)*
- Google Translate *(statistical, 2.3.1, 4.3.1)*
- Bing Translator *(statistical, 2.3.2, 4.3.2)*
- Yandex Translate *(statistical, 2.3.3, 4.3.3)*
- Moses *(mostly statistical, 2.3.4, 4.4)*
- TectoMT *(hybrid, 2.4.1, 4.5)*

I am testing on two separate test sets: WMT 2013 and Filtered Intercorp, as described in the section 3.1.

I have randomly selected 10 sentences, 5 from each set, to allow readers to compare the system for themselves; the results are in the attachment A.

Unfortunately, I cannot rule out the possibility that Google, Yandex or Bing Translator already have WMT 2013 sentences, or at least some of them, in their training data, as they have been public for about a year when I run the tests. It's less likely that they trained on Intercorp data – however, as they are black-box systems, we can never tell for sure.

5.2 Automated metrics

I compared the systems using several automated metrics, all of them implemented in Moses internal evaluator, and all of them described well in MACHÁČEK, 2012 (Czech). Specifically, they are BLEU, WER, TER, CDER and PER.

	BLEU	WER	TER	CDER	PER
Yandex	11.65	24.47	25.77	31.95	38.63
Moses	11.62	26.73	28.26	32.98	43.65
Google	8.79	21.22	22.35	27.36	37.39
Bing	7.22	19.93	21.11	25.82	36.51
TectoMT	5.81	16.46	17.87	25.76	31.20
PC Translator	5.02	17.99	18.85	25.06	31.54
Baseline	0.77	8.36	8.61	13.23	19.30

Table 11: Automated metrics on Filtered Intercorp

	BLEU	WER	TER	CDER	PER
Yandex	19.55	34.55	36.63	40.99	49.02
Google	17.96	33.58	35.58	38.64	48.35
Moses	17.44	33.20	35.18	38.81	49.14
Bing	15.49	30.80	32.93	36.22	46.71
TectoMT	8.80	25.00	26.68	30.45	41.47
PC Translator	6.74	21.41	22.60	26.75	35.48
Baseline	0.83	10.93	11.26	14.32	20.14

Table 12: Automated metrics on WMT 2013

For a comparable BLEU metric, I re-tokenize both reference and the tested system by Moses' built-in tokenizer skript. I also normalize punctuation, using script from WMT pages. I decided to use case-sensitive BLEU – that means that words *Кристиан* and *кристиан* are two different words.

As a baseline, I use a standard transliteration GOST 7.79 RUS (ГОВЕРДОВСКАЯ, 2002).

The scores are in tables 11 and 12, sorted by BLEU.

5.2.1 Discussion

The first notable thing is that the various metrics on a single corpus rougly agree on the order, except for small differences – most notably, on Intercorp, Yandex and our Moses would switch places, depending on the metric. Similarly, with WMT data, Moses, Yandex and Google would switch places, depending on the metric.

What was slightly surprising for me was the results of Yandex Translate. I have originally added Yandex to the list of systems only for "completeness", but it actually outperformed Google Translate and my Moses setup. In retrospect, this makes sense – Yandex is a Russian company and, as such, probably has better statistical models of Russian and a better morphology.

We can also note that the only purely rule-based system – PC Translator – always ended up last, hybrid system came out slightly better and more statistical systems came out the best (altough we *do* use morphological tags in Moses in the language model, as described in 4.4.2). It might be seen as the proof that statistical systems have better results; however it can also be seen as the proof that the metrics are better suited to statistical systems.

Difference between test corpora

What is also interesting is how much the results on Intercorp and WMT test data differ from each other, as seen in the Table 13, ordered by the quotient.

Every system had better results on WMT than on Intercorp, even our baseline.

	Intercorp	WMT	÷
Bing	7.22	15.49	2.15
Google	8.79	17.96	2.04
Yandex	11.65	19.55	1.68
TectoMT	5.81	8.80	1.51
Moses	11.62	17.44	1.50
PC Translator	5.02	6.74	1.34
Baseline	0.77	0.83	1.08

Table 13: Differences between BLEU on two test corpora

However, Bing and Google had almost twice as good BLEU on WMT than on Intercorp, while our systems were more “stable”.

This is probably caused by the fact, that Google and Bing train their models on more publicly accessible news data, while I added some prose to Moses parallel data along with the news. I still have mostly news data in the language model; if I used the Lib.ru data mentioned in 3.4.1, I could maybe get even better results on Intercorp.

This all, however, begs a more theoretical question. Is it right that we, as MT researchers, mostly test our systems on news data, as for example in all WMT translation tasks? Shouldn’t we broad the domain a bit, to include fiction, and maybe other literature – and possibly even more kind of data?

It’s possible that with heavy accent on parallel news data, we are skewing the translation systems so that they translate news articles well, but are significantly worse on other type of data. Should we strive for more general translation systems, or for translation systems, that do one type of text well?

This is an open question, and I don’t claim to have an answer. I don’t have the type of data Google probably has, so I don’t know what exactly are users translation and in what amounts.

It’s also mostly a rhetorical question. So far, every WMT translation task has been using news test data, and it doesn’t seem this will change in the near future.

5.3 Human evaluation

Appraise, TrueSkill

Originally, I planned to use Appraise system, used for human evaluation at WMT (FEDERMANN, 2012), and afterwards feeds its output to the TrueSkill algorithm (SAKAGUCHI; POST; VAN DURME, 2014), that was used in WMT 2014 as the best method for ordering systems based on human annotation (BOJAR; BUCK; FEDERMANN, et al., 2014).

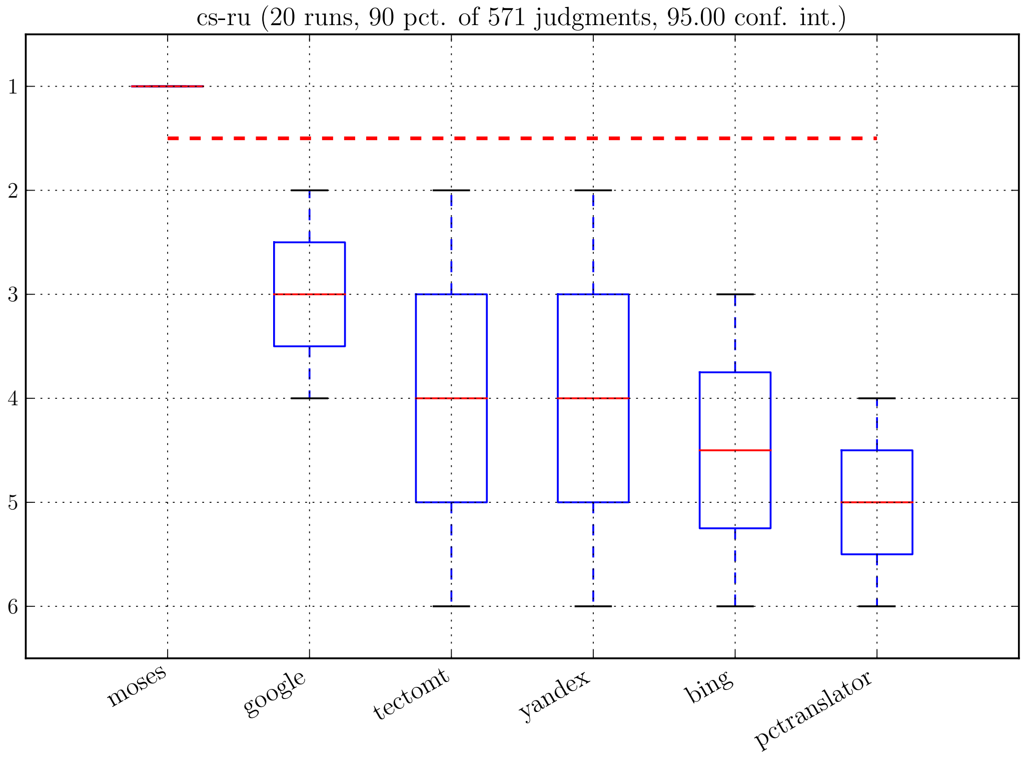

Figure 4: TrueSkill results

My PHP system

Appraise ended up being too hard to install and set-up[1], so I ended up writing my own simple PHP application, heavily inspired by Appraise.

Similar to Appraise, the application presents source sentence and a reference sentence to users, and asks them to rate them from best to worst. Unlike Appraise, that randomly chooses 5 systems every time, I used the fact that we have 6 systems and present all 6 to them to users. Sentences are took randomly from the two test sets.[2]

Results

I put the systems online and invited several people and tried to advertise it. However, with little funds and little time to experiment with the incentives, I decided to not pay anything for the annotation and let the user decide, how much he wants to annotate.

This led to only 38 annotated sequences in total.[3] This is not very much; however, one of the described features of TrueSkill algorithm is that it doesn't need that many judgements, so I decided to run it anyway.

---

[1] Appraise is a Django application, and I have almost no experience with either Django or Python

[2] Because of the difference in size of the sets, the set is selected first with 0.5 probability, and only then the sentence itself.

[3] For some reason, TrueSkill percieves this as 571 judgements; I am not sure how was this number created.

The results are on Figure 4. Moses is put as the best system into its own cluster, while the rest is put into another cluster. In the second cluster, PC Translator is probably the worst, but other than that we can't really say much.

I am not sure how much does this result tell us, given that it's only 38 sentences in total. Looking back, I should have probably made the task easier instead of harder (by giving less choices, not more choices), and I should have offered some financial incentives for annotators.

Some comments from the annotators

While I did not have that many annotators, they gave me some some comments about the test in general. I thought they would be interesting to note here.

- Some of the sentences are too long, which makes the judgements too hard.
- The translations seem either all correct or all wrong.
- The reference translation is either completely wrong (wrong alignment) or not a literal translation, which confuses the annotator.

The first comment is probably caused by the fact that I had used prose along with news articles. It would be possible to limit the annotation to shorter sentences, but then the annotation wouldn't be entirely accurate, since the shorter sentences are usually better translated in general.

The second comment is probably caused by the nature of MT tasks in general.

The last comment is the unfortunate side-effect of either misalignment or just less literal translations. It might be possible to *not* show the reference translation at all; I am not sure if that would get the judgements easier or harder.

5.4 Some observation about typical mistakes

I haven't done any systematic classification of the mistakes. I am not a native speaker of Russian, and I found out that spotting translation errors is not an easy task for a person with only passing knowledge of a language, even when the correct, reference translations are available.[4]

I have originally planned to use human annotators for error classification. However, with the low annotator activity in an arguably easier task of ranking the translation quality (5.3), I gave up on that idea too.

---

[4] In retrospect, my assumtion that spotting errors without really knowing the language will be doable has been very naive.

As far as I know, systematic evaluation of machine translation mistakes in this concrete language pair is currently done by my colleague Natalia Klyueva, who is a native Russian speaker and, at the time of writing this thesis, PhD student on ÚFAL.

I have, however, done some unsystematic observations about the various systems and their mistakes. All the examples in this section are from the test set, but most of them are not in the attachment A.

As in the attachment A, the transliteration is a standard transliteration GOST 7.79 RUS (ГОВЕРДОВСКАЯ, 2002).

PC Translator

PC Translator is, even to a person with only a passive knowledge of the language, obviously the worst of the six systems. The sentences are translated literally word for word, with no respect for differing grammar of the two languages.

For example, take the following PC Translator translation (not a complete sentence):

- **Opatrnost je ovšem na místě například**
  Осторожность есть конечно на месте например
  Ostorozhnost‘ est‘ konechno na meste naprimer

PC Translator ignores that the word "je" (to be) is not usually directly translated to Russian, and it just keeps the word in the sentence.

Both Yandex and Moses translates this phrase correctly:

- **Opatrnost je ovšem na místě například**
  Осторожность, однако, на месте, например
  Ostorozhnost‘, odnako, na meste, naprimer

Those mistakes seem like a direct consequence of the system design and I am not sure we can use it in any major way, except maybe for directly copying the dictionary (as I did in TectoMT, see 4.5).

Strange TectoMT mistakes

TectoMT returns sentences with "strange" mistakes, that I cannot fully grasp or understand.

As a first mistake, it has strange problems with punctiation. This can seem like a small problem, but each punctuation mark is an extra word in all the metrics.

For example, see this translation:

- **Může říct: ”Změníme ten zákon a povolíme to”, ale nemůže říct, že budeme dělat, jako že nic.**
  Он сказать , : » , мы изменим это закон и , , мы разрешить это » но , он не сказать , что мы делаем как .
  On skazat‘ , : » , my’ izmenim e‘to zakon i , , my’ razreshit‘ e‘to » no , on ne skazat‘ , chto my’ delaem kak .

You can see that the resulting punctuation is very chaotic.

Also, auxiliary words are often inserted at wrong places, as “и” (and) in the following translation:

- **Chvíli váhal a pak se přece jen vydal zpátky a zazvonil .**
  Минуту он колебался и , , потом же просто он отправитьсяся вернуться и , он позвонил .
  Minutu on kolebalsya i , , potom zhe prosto on otpravit‘syasya vernut‘sya i , on pozvonil .

Some of the mistakes of TectoMT are caused by the fact that the transfer is trained (as mentioned in 4.5) on lexical lemmas, while TectoMT uses t-lemmas. For example, in the following translation:

- **Existují mezi USA a mnoha evropskými národy názorové rozdíly?**
  Существуют между Преследуешь и между многими европейскими народами мнения разница ?
  Sushhestvuyut mezhdu Presleduesh‘ i mezhdu mnogimi evropejskimi narodami mneniya raznicza ?

I personally have no idea how “USA” got translated as “Преследуешь” (stalking); however, it’s translated like that in every sentence where it appears.

In my opinion, most of TectoMT mistakes are caused mainly by relative unstability of the scenario, and the fact that only the most simple models are used for transfer and generation. I believe further refinements of the models could make the translation better by far.

Untranslated words in online systems

None of the online statistical systems seem to solve the OOV problem by transliteration. Surprisingly even Yandex, that had the best resulting BLEU, usually keeps more words in Czech than Moses.

This might be the reason why in human anotation, Moses had better results, than Yandex; however, this is only a conjecture.

For example, take this Yandex translation:

- **Vítejte , sousede !**
  Добро пожаловать , sousede !
  Dobro pozhalovat‘ , sousede !

and compare with Moses translation:

- **Vítejte , sousede !**
  Добро пожаловать , сосед !
  Dobro pozhalovat‘ , sosed !

Lost negative

“Classic” mistake of statistical machine translation is the lost (or reversed) negativity. We can see it in our test set at places, for example with Google’s translation (not a complete sentence):

- **Postavy v dramatech nemluví slangem**
  Символы в драмах говорить сленг
  Simvoly’ v dramax govorit‘ sleng

However, this mistake is acually much less common than I anticipated. For example, Moses translated the phrase more correctly[5]:

- **Postavy v dramatech nemluví slangem**
  В драме не говорит - сленг
  V drame ne govorit - sleng

Language pivoting

The previous demonstration is also a proof of another classical mistake, caused by using English as a pivot language.

- **Postavy v dramatech nemluví slangem**
  Символы в драмах говорить сленг
  Simvoly’ v dramax govorit‘ sleng

The word “Postavy” is first translated to English as “Characters” and only then back to Russian as “Символы”. This mistake only appears in Google Translate and Bing Translator; it doesn’t appear in our systems (because we don’t use pivot languages), and it doesn’t appear in Yandex Translate.

---

[5] Only concerning the negative.

Conclusion

I have automated, built, improved, demonstrated and compared (both by human annotators and by automated metrics) several translation systems, both phrase-based and rule-based, between Czech and Russian.

From the systems I have tried, phrase-based translation systems are simply easier to build and give better results.

TectoMT as a more hybrid system shows promise, but with this language pair, the work is only starting; however, it is telling, that it's probably easier to build a new system based on Moses that reaches about the same translation quality as "state-of-the-art" systems, than it would be with TectoMT – and impossible with purely rule-based systems.

Future work

The first future work, as already mentioned in 4.6.1, should probably be a better Russian parser and a better Russian morphology. This would allow us to experiment more with post-editing; we could also use it in factored translation models in Moses; and of course it would allow us to build better models with TectoMT.

Bibliography

ANDERSEN, Henning. 2013. On the Origin of the Slavic Aspects: Aorist and Imperfect. *Journal of Slavic Linguistics*. 2013, vol. 21, no. 1, pp. 17–43. Available also from WWW: ⟨`http://www.questia.com/library/journal/1G1-332655163/on-the-origin-of-the-slavic-aspects-aorist-and-imperfect`⟩.

ARCHIBALD, Eric. 2008. *New Testament Greek Course* [online]. Swindon : Hayes Press, 2008 [visited on 2014-04-18]. Available from WWW: ⟨`http://www.hayespress.org/biblehelps-GREEK-30`⟩.

BÍLEK, Karel; ČECH, Josef; DAIBER, Joachim, et al. 2012. *FilmTit*. 2012. Project documentation. Available also from WWW: ⟨`https://github.com/runn1ng/FilmTit/blob/master/src/doc/result/documentation.pdf?raw=true`⟩.

BÍLEK, Karel; KLYUEVA, Natalia; KUBOŇ, Vladislav. 2013. Exploiting Maching Learning for Automatic Semantic Feature Assignment. In BOONTHUM-DENECKE, Chutima; YOUNGBLOOD, G. Michael (ed.). *FLAIRS Conference*. 2013. Available also from WWW: ⟨`http://www.aaai.org/ocs/index.php/FLAIRS/FLAIRS13/paper/view/5922`⟩.

BEJČEK, Eduard; HAJIČOVÁ, Eva; HAJIČ, Jan, et al. 2013. *Prague Dependency Treebank 3.0*. 2013. Available also from WWW: ⟨`http://hdl.handle.net/11858/00-097C-0000-0023-1AAF-3`⟩.

BEJČEK, Eduard; PANEVOVÁ, Jarmila; POPELKA, Jan, et al. 2012. Prague Dependency Treebank 2.5 – a revisited version of PDT 2.0. In KAY, Martin; BOITET, Christian (ed.). *Proceedings of the 24th International Conference on Computational Linguistics (Coling 2012)*. Mumbai, India : Coling 2012 Organizing Committee, 2012, pp. 231–246.

BIBLICA. 1973, 1978, 1984, 2011. *The Holy Bible, New International Version, NIV* [online]. Colorado Springs : Biblica, 1973, 1978, 1984, 2011 [visited on 2014-04-18]. Available from WWW: ⟨`http://www.biblegateway.com/passage/?search=1+Corinthians+1`⟩.

BÍLEK, Karel; ZEMAN, Daniel. 2013. CUni Multilingual Matrix in the WMT 2013 Shared Task. In. *Proceedings of the Eighth Workshop on Statistical Machine Translation*. Sofia, Bulgaria : Association for Computational Linguistics, 2013, pp. 85–91. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W13-2207`⟩.

BOJAR, Ondřej; GALUŠČÁKOVÁ, Petra; TÝNOVSKÝ, Miroslav. 2011. Evaluating Quality of Machine Translation from Czech to Slovak. In LOPATKOVÁ, Markéta (ed.). *Information Technologies – Applications and Theory*. 2011, pp. 3–9. ISBN 978-80-89557-01-1.

BOJAR, Ondřej; HOMOLA, Petr; KUBOŇ, Vladislav. 2005. An MT System Recycled. In. *Proceedings of MT Summit X*. 2005, pp. 380–387. Available also from WWW: ⟨`http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.103.8336\&rep=rep1\&type=pdf`⟩. ISBN 974-7431-26-2.

BOJAR, Ondřej; TAMCHYNA, Aleš. 2013. The Design of Eman, an Experiment Manager. *The Prague Bulletin of Mathematical Linguistics*. 2013, vol. 99, pp. 39–58. ISSN 0032-6585.

BOJAR, Ondřej. 2012. *Čeština a strojový překlad (Czech Language and Machine Translation)*. Praha, Czech Republic : ÚFAL, 2012. Studies in Computational and Theoretical Linguistics. ISBN 978-80-904571-4-0.

BOJAR, Ondrej; BUCK, Christian; FEDERMANN, Christian, et al. 2014. Findings of the 2014 Workshop on Statistical Machine Translation. In. *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA : Association for Computational Linguistics, 2014, pp. 12–58. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W/W14/W14-3302`⟩.

BOJAR, Ondřej; BUCK, Christian; CALLISON-BURCH, Chris, et al. 2013. Findings of the 2013 Workshop on Statistical Machine Translation. In. *Proceedings of the Eighth Workshop on Statistical Machine Translation*. Sofia, Bulgaria : Association for Computational Linguistics, 2013, pp. 1–44. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W13-2201`⟩.

BROWN, Peter F.; PIETRA, Vincent J. Della; PIETRA, Stephen A. Della; MERCER, Robert L. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Comput. Linguist.* 1993, vol. 19, no. 2, pp. 263–311. Available also from WWW: ⟨`http://acl.ldc.upenn.edu/J/J93/J93-2003.pdf`⟩. ISSN 0891-2017.

CALLISON-BURCH, Chris; KOEHN, Philipp; MONZ, Christof; SCHROEDER, Josh. 2009. Findings of the 2009 Workshop on Statistical Machine Translation. In. *Proceedings of the Fourth Workshop on Statistical Machine Translation*. Athens, Greece : Association for Computational Linguistics, 2009, pp. 1–28. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W/W09/W09-0401`⟩.

ČERMÁK, František; ROSEN, Alexandr. 2012. The Case of InterCorp, a multilingual parallel corpus. *International Journal of Corpus Linguistics*. 2012, vol. 13, no. 3, pp. 411–427. Available also from WWW: ⟨`http://utkl.ff.cuni.cz/~rosen/public/2012_intercorp_ijcl.pdf`⟩. ISSN 1384-6655.

CHEN, Stanley F; GOODMAN, Joshua. 1996. An empirical study of smoothing techniques for language modeling. In. *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Santa Cruz, California : Association for Computational Linguistics, 1996, pp. 310–318. ACL '96. Available also from WWW: ⟨`http://www.speech.sri.com/projects/srilm/manpages/pdfs/chen-goodman-tr-10-98.pdf`⟩.

COLMERAUER, Alain. 1970. *Les systèmes-Q; ou un formalisme pour analyser et synthétiser des phrases sur ordinateur*. 1970.

COMRIE, Bernard; CORBETT, Greville G. 2003. *The Slavonic Languages*. 2003. Routledge Language Family Series. ISBN 9780203213209.

CURRY, Haskell B. 1961. Some Logical Aspects of Grammatical Structure. In JAKOBSON, Roman O. (ed.). *Structure of Language and its Mathematical Aspects, volume 12 of Symposia on Applied Mathematics*. Providence : American Mathematical Society, 1961, pp. 56–68.

CURTA, Florin. 2004. The Slavic lingua franca (Linguistic notes of an archaeologist turned historian). *East Central Europe*. 2004, vol. 31, no. 1, pp. 125–148.

FEDERMANN, Christian. 2012. Appraise: An Open-Source Toolkit for Manual Evaluation of Machine Translation Output. *The Prague Bulletin of Mathematical Linguistics*. 2012, vol. 98, pp. 25–35. Available also from WWW: ⟨`https://ufal.mff.cuni.cz/pbml/98/art-federmann.pdf`⟩.

FLEK, Alexandr et al. 2012. *Bible, překlad 21. století* [online]. 2012 [visited on 2014-04-18]. Available from WWW: ⟨`http://onlineb21.bible21.cz/bible.php?kniha=1korintskym`⟩.

GAO, Qin; VOGEL, Stephan. 2008. Parallel Implementations of Word Alignment Tool. In. *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*. Columbus, Ohio : Association for Computational Linguistics, 2008, pp. 49–57. SETQA-NLP '08. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W08-0509.pdf`⟩. ISBN 978-1-932432-10-7.

ГОВЕРДОВСКАЯ, Р.Г.. 2002. *Правила транслитерации кирилловского письма латинским алфавитом (Rules of transliteration of Cyrillic script by Latin alphabet).* Moscow : ИПК Издательство стандартов, 2002. Available also from WWW: ⟨`http://nauka.kz/upload/files/45._GOST_7.79-2000.pdf`⟩.

GUTKIN, Alexander. 2000. *Log-linear interpolation of language models.* 2000. Available also from WWW: ⟨`http://www.cstr.ed.ac.uk/downloads/publications/2000/gutkin_mphil.pdf`⟩.

HAJIČ, Jan. 1987. RUSLAN: An MT System Between Closely Related Languages. In. *Proceedings of the Third Conference on European Chapter of the Association for Computational Linguistics.* Stroudsburg, PA, USA : Association for Computational Linguistics, 1987, pp. 113–117. EACL '87. Available also from WWW: ⟨`http://dx.doi.org/10.3115/976858.976879`⟩.

HAJIČ, Jan; HRIC, Jan; KUBOŇ, Vladislav. 2000. Machine Translation of Very Close Languages. In. *Proceedings of the Sixth Conference on Applied Natural Language Processing.* Seattle, Washington : Association for Computational Linguistics, 2000, pp. 7–12. ANLC '00. Available also from WWW: ⟨`http://www.aclweb.org/anthology/A00-1002`⟩.

HAJIČ, Jan; PANEVOVÁ, Jarmila; BURÁŇOVÁ, Eva; UREŠOVÁ, Zdeňka; BÉMOVÁ, Alla. 1999. *A Manual for Analytic Layer Annotation of the Prague Dependency Treebank.* 1999. Available also from WWW: ⟨`http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/a-layer/html/index.html`⟩.

HILLEGASS, Aaron; PREBLE, Adam. 2011. *Cocoa Programming for Mac OS X.* Fourth. 2011. Available also from WWW: ⟨`https://www.bignerdranch.com/book/cocoa_programming_for_mac_os_x_th_edition_`⟩. ISBN 9780132902205.

HOMOLA, Petr; KUBOŇ, Vladislav; VIČIČ, Jernej. 2009. Shallow Transfer Between Slavic Languages. In. *Proceedings of Balto-Slavonic Natural Language Processing.* Kraków, Poland : Polska Akademia Nauk, 2009, pp. 219–232. ISBN 978-83-60434-59-8.

KLYUEVA, Natalia; BOJAR, Ondřej. 2008. UMC 0.1: Czech-Russian-English Multilingual Corpus. In. *Proceedings of International Conference Corpus Linguistics.* 2008, pp. 188–195. Available also from WWW: ⟨`http://ufal.mff.cuni.cz/legacy/umc/cer/download.php?f=umc-0.1-paper-2008.pdf`⟩.

KOEHN, Philipp. 2010. *Statistical Machine Translation.* Cambridge : Cambridge University Press, 2010. Statistical Machine Translation. Available also from WWW: ⟨`http://www.statmt.org/book/`⟩. ISBN 9780521874151.

KOEHN, Philipp; HADDOW, Barry. 2012. Interpolated backoff for factored translation models. In. *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas (AMTA)*. 2012.

KOEHN, Philipp; HOANG, Hieu. 2007. Factored Translation Models. In. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic : Association for Computational Linguistics, 2007, pp. 868–876. Available also from WWW: ⟨`http://homepages.inf.ed.ac.uk/pkoehn/publications/emnlp2007-factored.pdf`⟩.

KOEHN, Philipp; HOANG, Hieu; BIRCH, Alexandra, et al. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In. *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. Prague, Czech Republic : Association for Computational Linguistics, 2007, pp. 177–180. ACL '07. Available also from WWW: ⟨`http://acl.ldc.upenn.edu/P/P07/P07-2045.pdf`⟩.

KOEHN, Philipp; OCH, Franz Josef; MARCU, Daniel. 2003. Statistical Phrase-based Translation. In. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*. Edmonton, Canada : Association for Computational Linguistics, 2003, pp. 48–54. NAACL '03. Available also from WWW: ⟨`http://acl.ldc.upenn.edu/N/N03/N03-1017.pdf`⟩.

KORTLANDT, Frederik. 1982. Early dialectal diversity in South Slavic I. *Studies in Slavic and General Linguistics*. 1982, pp. 177–192. Available also from WWW: ⟨`http://kortlandt.nl/publications/art058e.pdf`⟩.

MACHÁČEK, Matouš. 2012. *Metriky pro optimalizaci modelů strojového překladu*. 2012. Bachelor thesis. Available also from WWW: ⟨`http://www1.cuni.cz/~obo/vyuka/projekty/machacek-metriky.pdf`⟩.

MALLORY, J. P.; ADAMS, Douglas Q. 2006. *The Oxford Introduction to Proto-Indo-European and The Proto-Indo-European World*. New York : Oxford University Press, 2006. Available also from WWW: ⟨`http://ukcatalogue.oup.com/product/9780199296682.do`⟩. ISBN 9780199287918.

MAREČEK, David; KLJUEVA, Natalia. 2009. Converting Russian Treebank SynTagRus into Praguian PDT Style. In. *Proceedings of the Workshop on Multilingual Resources, Technologies and Evaluation for Central and Eastern European Languages*. Borovets, Bulgaria : Association for Computational Linguistics, 2009, pp. 26–31. MRTECEEL

’09. Available also from WWW: ⟨`http://ufal.mff.cuni.cz/~marecek/papers/2009_ranlp.pdf`⟩.

MAREČEK, David; POPEL, Martin; ŽABOKRTSKÝ, Zdeněk. 2010. Maximum entropy translation model in dependency-based MT framework. In. *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*. 2010, pp. 201–206. ISBN 978-1-932432-71-8.

MIKULOVÁ, Marie; BÉMOVÁ, Alevtina; HAJIČ, Jan, et al. 2005. *A Manual for Tectogrammatical Layer Annotation of the Prague Dependency Treebank*. 2005. Available also from WWW: ⟨`http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/t-layer/html/index.html`⟩.

MOSES MT. 2013. *Moses/Overview* [online]. 2013 [visited on 2014-05-10]. Available from WWW: ⟨`http://www.statmt.org/moses/?n=Moses.Overview`⟩.

OCH, Franz Josef. 2005. *MT Summit X*. Phuket : Google, 2005.

OCH, Franz Josef; NEY, Hermann. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Comput. Linguist.* 2003, vol. 29, no. 1, pp. 19–51. Available also from WWW: ⟨`http://acl.ldc.upenn.edu/J/J03/J03-1002.pdf?q=modles`⟩. ISSN 0891-2017.

OFFORD, Derek. 1996. *Using Russian: A Guide to Contemporary Usage*. Cambridge : Cambridge University Press, 1996. Available also from WWW: ⟨`http://books.google.cz/books?id=iWy0clZRkQwC`⟩. ISBN 9780521457606.

OLIVA, Karel. 1989. *A Parser for Czech Implemented in Systems Q*. Prague : Matematicko-fyzikální fakulta UK, 1989. Explizite Beschreibung der Sprache und automatische Textbearbeitung.

PARKER, Micheal W. 2009. *Hellenistic Greek* [online]. 2009 [visited on 2014-04-18]. Available from WWW: ⟨`http://greek-language.com/grammar/22.html`⟩.

POPOVIĆ, Maja; NEY, Hermann. 2004. Towards the use of Word Stems and Suffixes for Statistical Machine Translation. In. *In Proceedings of The International Conference on Language Resources and Evaluation*. 2004. Available also from WWW: ⟨`http://www.ecess.eu/ECESS/Public_documents/Popovic_Stem\%2BSuffixforSMT_LREC04.pdf`⟩.

RINGE, Don. 2008. *From Proto-Indo-European to Proto-Germanic*. Oxford : Oxford University Press, 2008. A Linguistic History of English. Available also from WWW: ⟨`http://ukcatalogue.oup.com/product/9780199552290.do`⟩. ISBN 9780199552290.

SAKAGUCHI, Keisuke; POST, Matt; VAN DURME, Benjamin. 2014. Efficient Elicitation of Annotations for Human Evaluation of Machine Translation. In. *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA : Association for Computational Linguistics, 2014, pp. 1–11. Available also from WWW: ⟨`http://www.aclweb.org/anthology/W14-3301`⟩.

SCHENKER, Alexander M. 1993. Proto-Slavonic. *The Slavonic Languages*. 1993, pp. 60–121.

SCHMID, Helmut. 1994. Probabilistic part-of-speech tagging using decision trees. In. *Proceedings of international conference on new methods in language processing*. 1994, pp. 44–49. Available also from WWW: ⟨`http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/tree-tagger1.pdf`⟩.

SCHMID, Helmut. 1995. Improvements In Part-of-Speech Tagging With an Application To German. In. *In Proceedings of the ACL SIGDAT-Workshop*. 1995, pp. 47–50. Available also from WWW: ⟨`http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/tree-tagger2.pdf`⟩.

SGALL, P. 1969. *A Functional approach to syntax in generative description of language*. New York : American Elsevier Pub. Co., 1969. Mathematical linguistics and automatic language processing. ISBN 9780444000453.

SGALL, Petr. 1967. *Generativní popis jazyka a česká deklinace*. Prague : Academia, 1967.

SIEWIERSKA, Anna; UHLÍŘOVÁ, Ludmila. 1998. An overview of word order in Slavic languages. *Constituent Order in the Languages of Europe*. 1998, vol. 20, no. 1, pp. 105. Available also from WWW: ⟨`http://www.degruyter.com/viewbooktoc/product/4280`⟩.

ŠMILAUER, Vladimír. 1958. *Učebnice větného rozboru*. 1958. Učební texty vysokých škol.

SMITH, Jason R.; SAINT-AMAND, Herve; PLAMADA, Magdalena, et al. 2013. Dirt Cheap Web-Scale Parallel Text from the Common Crawl. In. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria : Association for Computational Linguistics, 2013, pp. 1374–1383. Available also from WWW: ⟨`http://www.aclweb.org/anthology/P13-1135`⟩.

SPOUSTOVÁ, Drahomíra; SPOUSTA, Miroslav; PECINA, Pavel. 2010. Building a Web Corpus of Czech. In. *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010).* Valletta, Malta : European Language Resources Association, 2010, pp. 998–1001. Available also from WWW: ⟨`http://www.lrec-conf.org/proceedings/lrec2010/pdf/810_Paper.pdf`⟩. ISBN 2-9517408-6-7.

STOLCKE, Andreas. 2002. SRILM-an extensible language modeling toolkit. In. *Proceedings International Conference on Spoken Language Processing.* 2002, pp. 257–286. Available also from WWW: ⟨`http://www.speech.sri.com/cgi-bin/run-distill?papers/icslp2002-srilm.ps.gz`⟩.

STOLCKE, Andreas; ZHENG, Jing; WANG, Wen; ABRASH, Victor. 2011. SRILM at sixteen: Update and outlook. In. *Proceedings IEEE Automatic Speech Recognition and Understanding Workshop.* 2011. Available also from WWW: ⟨`http://t3-1.yum2.net/index/www.speech.sri.com/papers/asru2011-srilm.pdf`⟩.

SUSSEX, Roland; CUBBERLEY, Paul. 2011. *The Slavic Languages.* Cambridge : Cambridge University Press, 2011. Cambridge Language Surveys. Available also from WWW: ⟨`http://www.cambridge.org/us/academic/subjects/languages-linguistics/european-language-and-linguistics/slavic-languages`⟩. ISBN 9780521294485.

WIKISOURCE. 2013. *Biblia Sacra Vulgata (Stuttgartensia)/ad Corinthios I — Wikisource,* [online]. 2013 [visited on 2014-04-18]. Available from WWW: ⟨`http://la.wikisource.org/w/index.php?title=Biblia_Sacra_Vulgata_(Stuttgartensia)/ad_Corinthios_I\&oldid=62334`⟩.

ŽABOKRTSKÝ, Zdeněk. 2010. *From Treebanking to Machine Translation.* 2010. Habiliation thesis. Available also from WWW: ⟨`https://ufal.mff.cuni.cz/~zabokrtsky/publications/theses/hab-zz.pdf`⟩.

ŽABOKRTSKÝ, Zdeněk; PTÁČEK, Jan; PAJAS, Petr. 2008. TectoMT: Highly Modular MT System with Tectogrammatics Used as Transfer Layer. In. *ACL 2008 WMT: Proceedings of the Third Workshop on Statistical Machine Translation.* Columbus, OH, USA : Association for Computational Linguistics, 2008, pp. 167–170. ISBN 978-1-932432-09-1.

ŽABOKRTSKÝ, Zdeněk; POPEL, Martin. 2009. Hidden Markov tree model in dependency-based machine translation. In. *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers.* 2009, pp. 145–148. Available also from WWW: ⟨`http://www.aclweb.org/anthology/P09-2037`⟩.

ZEMAN, Dan; HANA, Jiří; HANOVÁ, Hana, et al. 2005. *ÚFAL Technical Report*. 2005. Available also from WWW: ⟨`http://ufal.mff.cuni.cz/pdt2.0/doc/manuals/en/m-layer/html/index.html`⟩.

ZEMAN, Daniel. 2010. Hard Problems of Tagset Conversion. In. *Proceedings of the Second International Conference on Global Interoperability for Language Resources*. 2010, pp. 181–185. Available also from WWW: ⟨`http://ufal.ms.mff.cuni.cz/~zeman/publikace/2010-01/submitted2-camera-ready.pdf`⟩.

Attachments

A. Sample of experiment results

In this attachment, I am demonstrating the results of the six MT systems on ten randomly selected sentences.

I am presenting all Russian sentences with both original Cyrillic and GOST 7.79 RUS transliteration (ГОВЕРДОВСКАЯ, 2002), for the convenience of the reader.

A.1 Sample sentences

I have randomly selected 5 sentences from Intercorp and 5 from WMT-2013.

A.1.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Мне всего двадцать четыре года , и я не могу доживать свой век с книжечкой инвалида труда , скитаться по лечебницам , зная , что это ни к чему .
  Mne vsego dvadczat‘ chety’re goda , i ya ne mogu dozhivat‘ svoj vek s knizhechkoj invalida truda , skitat‘sya po lechebniczam , znaya , chto e‘to ni k chemu .
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генерал ходил взад и вперед по комнате , куря свою пенковую трубку .
  General xodil vzad i vpered po komnate , kurya svoyu penkovuyu trubku .
- **” Možná že žádné brilianty neexistují ? ”**
  - Может быть , никаких брильянтов нет ?
  - Mozhet by’t‘ , nikakix bril‘yantov net ?
- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  Эта работа не допускала описки - так же как прицел орудия .
  E‘ta rabota ne dopuskala opiski - tak zhe kak pricel orudiya .
- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  Он с гордостью вспомнил , как легко покорил когда-то сердце прекрасной Елены Боур .
  On s gordost‘yu vspomnil , kak legko pokoril kogda-to serdce prekrasnoj Eleny’ Bour .

A.1.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Тьяго Силва, который является одним из лучших защитников в мире, также мотивирует всех двигаться вперед.

T'yago Silva, kotory'j yavlyaetsya odnim iz luchshix zashhitnikov v mire, takzhe motiviruet vsex dvigat'sya vpered.

- **"Dávali mi pět let života a už je to sedm," říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**
  "Мне давали пять лет, я прожил семь", - говорит он, между жизнью и смертью, лежа в кровати в приюте паллиативного ухода Виктор-Гадбуа в Белёй, куда прибыл накануне.
  "Mne davali pyat' let, ya prozhil sem'", - govorit on, mezhdu zhizn'yu i smert'yu, lezha v krovati v priyute palliativnogo uxoda Viktor-Gadbua v Belyoj, kuda priby'l nakanune.
- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Однако внимательность нужна, например, на мостах, где поверхность может быть намерзшая и скользкая.
  Odnako vnimatel'nost' nuzhna, naprimer, na mostax, gde poverxnost' mozhet by't' namerzshaya i skol'zkaya.
- **Prostě je ignoruji.**
  Я просто не обращаю внимания.
  Ya prosto ne obrashhayu vnimaniya.
- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**
  По словам доктора Кристиан Мартель, система здравоохранения Квебека недостаточно эффективна, чтобы обеспечить право на паллиативный уход высокого качества до того, как будет разрешен переход к эвтаназии.
  Po slovam doktora Kristian Martel', sistema zdravooxraneniya Kvebeka nedostatochno e'ffektivna, chtoby' obespechit' pravo na palliativny'j uxod vy'sokogo kachestva do togo, kak budet razreshen perexod k e'vtanazii.

A.2 PC Translator

A.2.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Есть мне только двадцать четыре рейс и не могу пережив весь свой жизнь с удостоверение личности инвалидами труд и болтаться по больницах , когда знаю , что это бесполезные .
  Est' mne tol'ko dvadczat' chety're rejs i ne mogu perezhiv ves' svoj zhizn' s udostoverenie lichnosti invalidami trud i boltat'sya po bol'niczax , kogda znayu , chto e'to bespolezny'e .
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генерал ходил по комнате туда - сюда , дыма свою пенковая трубка .
  General xodil po komnate tuda - syuda , dy'ma svoyu penkovaya trubka .

- **” Možná že žádné brilianty neexistují ? ”**
  ” возможно никакие бриллианты отсутствовать ?
  ” vozmozhno nikakie brillianty’ otsutstvovat‘ ?

- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  Обмен причём ошибиться , требовало то такой же аккуратность , как когда специализируется пушка .
  Obmen prichyom oshibit‘sya , trebovalo to takoj zhe akkuratnost‘ , kak kogda specializiruetsya pushka .

- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  С гордостью вспомнил , как ходко захватил когда - то сердце красивое Елена Баурове .
  S gordost‘yu vspomnil , kak xodko zaxvatil kogda - to serdce krasivoe Elena Baurove .

A.2.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Thиаго Силва, какой принадлежать к лучшим защитник в мире, тоже даёт возможность остальным плечом к плечу рост.
  Thiago Silva, kakoj prinadlezhat‘ k luchshim zashhitnik v mire, tozhe dayot vozmozhnost‘ ostal‘ny’m plechom k plechu rost.

- **”Dávali mi pět let života a už je to sedm,” říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**
  ”давали мне пять лет жизни и уже это семь,” говорит минус эмоций в своем гнезде в доме для башка опеку Victor-Гадбоис зажечься Бєлоєил, куда доехал предшествующий день.
  ”davali mne pyat‘ let zhizni i uzhe e‘to sem‘,” govorit minus e‘mocij v svoem gnezde v dome dlya bashka opeku Victor-Gadbois zazhech‘sya Bєloєil, kuda doexal predshestvuyushhij den‘.

- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Осторожность есть конечно на месте например на некоторых перекрытие, где может быть покрытие замёрзший и сальный.
  Ostorozhnost‘ est‘ konechno na meste naprimer na nekotory’x perekry’tie, gde mozhet by’t‘ pokry’tie zamyorzshij i sal‘ny’j.

- **Prostě je ignoruji.**
  Запросто есть игнорирую.
  Zaprosto est‘ ignoriruyu.

- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**
  По врачи Christiane Мартєлове нет qуєбєцкэ медицинский комплекс до-

статочно производительный, чтобы обеспечил подход всех» личностей к качественный башка опеку, нежели можно будет согласиться с проведением єутаназиє.
Po vrachi Christiane Martєlove net quєвєczke‘ medicinskij kompleks dostatochno proizvoditel‘ny’j, chtoby’ obespechil podxod vsex» lichnostej k kachestvenny’j bashka opeku, nezheli mozhno budet soglasit‘sya s provedeniem єutanaziє.

A.3 Google Translate

A.3.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Я всего двадцать четыре года, и я живу свою жизнь с купоном недействительным работы и болтаться в больнице, зная, что это бесполезно.
  Ya vsego dvadczat‘ chety’re goda, i ya zhivu svoyu zhizn‘ s kuponom nedejstvitel‘ny’m raboty’ i boltat‘sya v bol‘nice, znaya, chto e‘to bespolezno.
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генеральный ходил по комнате взад и вперед, его дым пены.
  General‘ny’j xodil po komnate vzad i vpered, ego dy’m peny’.
- **” Možná že žádné brilianty neexistují ? ”**
  ”Возможно, не алмазы там?”
  ”Vozmozhno, ne almazy’ tam?”
- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  Он не мог ошибиться в этом, это требуется такой же точности, как при фокусировке пушки.
  On ne mog oshibit‘sya v e‘tom, e‘to trebuetsya takoj zhe tochnosti, kak pri fokusirovke pushki.
- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  Мы с гордостью вспомнил, как легко покорил когда-то сердце прекрасной Елены Баур.
  My’ s gordost‘yu vspomnil, kak legko pokoril kogda-to serdce prekrasnoj Eleny’ Baur.

A.3.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Тьяго Силва, один из лучших защитников в мире, также позволяет другой параллельный рост.
  T‘yago Silva, odin iz luchshix zashhitnikov v mire, takzhe pozvolyaet drugoj parallel‘ny’j rost.
- **”Dávali mi pět let života a už je to sedm,” říká bez emocí na svém lůžku**

v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.
”Они дали мне пять лет жизни, и это семь”, говорит он без эмоций на его постели у себя дома для паллиативной помощи Виктор-Гадбуа в Beloeil, куда они прибыли в предыдущий день.
”Oni dali mne pyat‘ let zhizni, i e‘to sem‘”, govorit on bez e‘mocij na ego posteli u sebya doma dlya palliativnoj pomoshhi Viktor-Gadbua v Beloeil, kuda oni priby’li v predy’dushhij den‘.

- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Внимание, однако, находится в месте, например, некоторые перемычки, где поверхность может быть ледяной и скользкий.
  Vnimanie, odnako, naxoditsya v meste, naprimer, nekotory’e peremy’chki, gde poverxnost‘ mozhet by’t‘ ledyanoj i skol‘zkij.
- **Prostě je ignoruji.**
  Просто игнорируйте их.
  Prosto ignorirujte ix.
- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**
  По словам доктора Кристиана Martel Квебеке система здравоохранения не является достаточно мощным, чтобы обеспечить доступ для всех людей на высококачественной паллиативной помощи, прежде чем он может согласиться проводить эвтаназию.
  Po slovam doktora Kristiana Martel Kvebeke sistema zdravooxraneniya ne yavlyaetsya dostatochno moshhny’m, chtoby’ obespechit‘ dostup dlya vsex lyudej na vy’sokokachestvennoj palliativnoj pomoshhi, prezhde chem on mozhet soglasit‘sya provodit‘ e‘vtanaziyu.

A.4 Bing Translator

A.4.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Мне только двадцать четыре года, и я не могу prožit всю мою жизнь с доказательствами инвалидов на работу и общаться с больницы, когда я знаю, что это бесполезно.
  Mne tol‘ko dvadczat‘ chety’re goda, i ya ne mogu prožit vsyu moyu zhizn‘ s dokazatel‘stvami invalidov na rabotu i obshhat‘sya s bol‘nicy’, kogda ya znayu, chto e‘to bespolezno.
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генерал ходил вокруг комнаты здесь и там, дым ИТ.
  General xodil vokrug komnaty’ zdes‘ i tam, dy’m IT.

- **” Možná že žádné brilianty neexistují ? ”**
  «Может есть не алмазы?»
  «Mozhet est‘ ne almazy’?»

- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  Ему не было разрешено сделать ошибку, он требует такой же точности, как цели пушки.
  Emu ne by’lo razresheno sdelat‘ oshibku, on trebuet takoj zhe tochnosti, kak celi pushki.

- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  С гордостью, что он вспомнил, как легко завоевал однажды сердце красивой Елены Baurové.
  S gordost‘yu, chto on vspomnil, kak legko zavoeval odnazhdy’ serdce krasivoj Eleny’ Baurové.

A.4.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Тиаго Силва, который является одним из лучших защитников в мире, тоже, позволяя другим расти рядом друг с другом.
  Tiago Silva, kotory’j yavlyaetsya odnim iz luchshix zashhitnikov v mire, tozhe, pozvolyaya drugim rasti ryadom drug s drugom.

- **”Dávali mi pět let života a už je to sedm,” říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**
  «Они дали мне пять лет жизни, и это семь,» говорит он без эмоций на его кровати в доме для паллиативной помощи, Виктор-Gadbois в Белэиле, где он имел прибыл накануне.
  «Oni dali mne pyat‘ let zhizni, i e‘to sem‘,» govorit on bez e‘mocij na ego krovati v dome dlya palliativnoj pomoshhi, Viktor-Gadbois v Bele‘ile, gde on imel priby’l nakanune.

- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Предупреждение является, однако, на месте, например, на некоторых из přemostěních, где она может быть конечно не и скользкой поверхности.
  Preduprezhdenie yavlyaetsya, odnako, na meste, naprimer, na nekotory’x iz přemostěních, gde ona mozhet by’t‘ konechno ne i skol‘zkoj poverxnosti.

- **Prostě je ignoruji.**
  Просто игнорирует.
  Prosto ignoriruet.

- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**
  По словам д-р Кристиан Martelové не является достаточно мощным, чтобы

обеспечить доступ всех лиц к паллиативной помощи высокого качества, чем это будет возможным согласиться выполнять эвтаназии система здравоохранения Квебека.
Po slovam d-r Kristian Martelové ne yavlyaetsya dostatochno moshhny'm, chtoby' obespechit' dostup vsex licz k palliativnoj pomoshhi vy'sokogo kachestva, chem e'to budet vozmozhny'm soglasit'sya vy'polnyat' e'vtanazii sistema zdravooxraneniya Kvebeka.

A.5 Yandex Translate

A.5.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Мне только двадцать четыре лет, и я не могу prožit всю свою жизнь с legitimací инвалидов труда и сутулиться вокруг после больницы , когда я знаю , что это бесполезно .
  Mne tol'ko dvadczat' chety're let, i ya ne mogu prožit vsyu svoyu zhizn' s legitimací invalidov truda i sutulit'sya vokrug posle bol'nicy' , kogda ya znayu , chto e'to bespolezno .
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генерал ходил по комнате взад и вперед , дыма свою pěnovku .
  General xodil po komnate vzad i vpered , dy'ma svoyu pěnovku .
- **" Možná že žádné brilianty neexistují ? "**
  " Возможно, что никаких бриллиантов нет ? "
  " Vozmozhno, chto nikakix brilliantov net ? "
- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  В дальнейшем при этом сделать ошибку , требуется такой же точности , как когда направлена пушка .
  V dal'nejshem pri e'tom sdelat' oshibku , trebuetsya takoj zhe tochnosti , kak kogda napravlena pushka .
- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  С гордостью вспомнил , как легко когда-то завоевал сердце прекрасной Елены Baurové .
  S gordost'yu vspomnil , kak legko kogda-to zavoeval serdce prekrasnoj Eleny' Baurové .

A.5.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Тьяго Силва, который относится к лучшим усталых защитников в мире, тоже позволяет другим рядом друг с другом рост.

T‘yago Silva, kotory’j otnositsya k luchshim ustaly’x zashhitnikov v mire, tozhe pozvolyaet drugim ryadom drug s drugom rost.

- **”Dávali mi pět let života a už je to sedm,” říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**

  ”Дали мне пять лет жизни, и уже это семь”, говорит без эмоций на своей койке в доме для паллиативной помощи Виктор-Gadbois в Beloeil, куда прибыл последний день.

  ”Dali mne pyat‘ let zhizni, i uzhe e‘to sem‘”, govorit bez e‘mocij na svoej kojke v dome dlya palliativnoj pomoshhi Viktor-Gadbois v Beloeil, kuda priby’l poslednij den‘.

- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**

  Осторожность, однако, на месте, например, на некоторых přemostěních, где может быть поверхность namrzlý и скользкий.

  Ostorozhnost‘, odnako, na meste, naprimer, na nekotory’x přemostěních, gde mozhet by’t‘ poverxnost‘ namrzlý i skol‘zkij.

- **Prostě je ignoruji.**

  Просто игнорирую.

  Prosto ignoriruyu.

- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**

  По doktorky Кристиан Martelové не quebecký системы медицинского достаточно мощный, чтобы обеспечить доступ всех людей к качественной паллиативной помощи, чем можно будет согласиться с проведением эвтаназии.

  Po doktorky Kristian Martelové ne quebecký sistemy’ medicinskogo dostatochno moshhny’j, chtoby’ obespechit‘ dostup vsex lyudej k kachestvennoj palliativnoj pomoshhi, chem mozhno budet soglasit‘sya s provedeniem e‘vtanazii.

A.6 Moses

A.6.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**

  Мне только 24 лет , и я не могу метаниях всю свою жизнь с легитимностью инвалидов работы и слоняться по госпиталям , когда я знаю , что это бесполезно .

  Mne tol‘ko 24 let , i ya ne mogu metaniyax vsyu svoyu zhizn‘ s legitimnost‘yu invalidov raboty’ i slonyat‘sya po gospitalyam , kogda ya znayu , chto e‘to bespolezno .

- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**

  Генерал ходил по комнате , свою pěnovku дыма .

General xodil po komnate , svoyu pěnovku dy’ma .

- **” Možná že žádné brilianty neexistují ? ”**
  - Возможно , что не существует никаких бриллиантами ? ”
  - Vozmozhno , chto ne sushhestvuet nikakix brilliantami ? ”
- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  При этом нельзя было сделать ошибку , это же точности , как , например , когда происходило уничтожение .
  Pri e‘tom nel‘zya by’lo sdelat‘ oshibku , e‘to zhe tochnosti , kak , naprimer , kogda proisxodilo unichtozhenie .
- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  С гордостью вспомнил , как легко он когда-то сердце прекрасной Елены baurové .
  S gordost‘yu vspomnil , kak legko on kogda-to serdce prekrasnoj Eleny’ baurové .

A.6.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Thiago Сильва , который является лучшим защитником в мире , и позволяет другим рядом экономический рост .
  Thiago Sil‘va , kotory’j yavlyaetsya luchshim zashhitnikom v mire , i pozvolyaet drugim ryadom e‘konomicheskij rost .
- **”Dávali mi pět let života a už je to sedm,” říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**
  ” Мне давали пять лет , и это уже семь , - говорит без эмоций в своей постели в доме для обслуживания полумерой Виктор beloeil , куда он приехал в предыдущий день .
  ” Mne davali pyat‘ let , i e‘to uzhe sem‘ , - govorit bez e‘mocij v svoej posteli v dome dlya obsluzhivaniya polumeroj Viktor beloeil , kuda on priexal v predy’dushhij den‘ .
- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Осторожность , однако , на месте , например , на некоторых обводку , где может быть поверхность полосе и скользким .
  Ostorozhnost‘ , odnako , na meste , naprimer , na nekotory’x obvodku , gde mozhet by’t‘ poverxnost‘ polose i skol‘zkim .
- **Prostě je ignoruji.**
  просто игнорируют .
  prosto ignoriruyut .
- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní palia-**

tivní péči, než bude možno souhlasit s provedením eutanazie.
По словам доктора Кристиан Мартелл не Квебек служба здравоохранения достаточно полезным , чтобы обеспечить доступ всех людей полумерой качества обслуживания , чем можно будет согласиться с сделали эвтаназия .
Po slovam doktora Kristian Martell ne Kvebek sluzhba zdravooxraneniya dostatochno polezny'm , chtoby' obespechit' dostup vsex lyudej polumeroj kachestva obsluzhivaniya , chem mozhno budet soglasit'sya s sdelali e'vtanaziya .

A.7 TectoMT

A.7.1 Intercorp

- **Je mi teprve čtyřiadvacet let a nemohu prožit celý svůj život s legitimací invalidy práce a potloukat se po nemocnicích , když vím , že je to marné .**
  Он мне только сутки годы и , я не могу пережить всю его жизнь с документом инвалида работы и слонялись его по больницах , когда я знаю , что это бесполезно .
  On mne tol'ko sutki gody' i , ya ne mogu perezhit' vsyu ego zhizn' s dokumentom invalida raboty' i slonyalis' ego po bol'niczax , kogda ya znayu , chto e'to bespolezno .
- **Generál chodil po pokoji sem a tam , kouře svou pěnovku .**
  Генерал ходил по комната сюда , и там , курят его шумовка .
  General xodil po komnata syuda , i tam , kuryat ego shumovka .
- **” Možná že žádné brilianty neexistují ? ”**
  Что » может никакие бриллианты не существуют » ?
  Chto » mozhet nikakie brillianty' ne sushhestvuyut » ?
- **Nesměl při tom udělat chybu , vyžadovalo to stejnou přesnost , jako když se zaměřuje dělo .**
  При этом он не сделал ошибку , требовало это же точность , когда как сосредоточиться пушка .
  Pri e'tom on ne sdelal oshibku , trebovalo e'to zhe tochnost' , kogda kak sosredotochit'sya pushka .
- **S hrdostí vzpomněl , jak snadno dobyl kdysi srdce krásné Heleny Baurové .**
  С гордостью он вспомнил , как легко он завоевал когда-то сердце красивой Елены Баурова .
  S gordost'yu on vspomnil , kak legko on zavoeval kogda-to serdce krasivoj Eleny' Baurova .

A.7.2 WMT

- **Thiago Silva, který patří k nejlepším obráncům na světě, taky umožňuje ostatním vedle sebe růst.**
  Тгиаго Силва , который принадлежит к хорошим защитникам на мире ,

также он позволяет другим возле него роста .
Tgiago Silva , kotory'j prinadlezhit k xoroshim zashhitnikam na mire , takzhe on pozvolyaet drugim vozle nego rosta .

- **"Dávali mi pět let života a už je to sedm," říká bez emocí na svém lůžku v domě pro paliativní péči Victor-Gadbois v Beloeil, kam přijel předešlý den.**
  » они давали мне пять годы жизни и , уже это семь » , он говорит без эмоций на его постели в доме для Паллиативное ухода вицтор-гадбоис в Бэлоэил , куда он приехал предыдущий день .
  » oni davali mne pyat' gody' zhizni i , uzhe e'to sem' » , on govorit bez e'mocij na ego posteli v dome dlya Palliativnoe uxoda vicztor-gadbois v Be'loe'il , kuda on priexal predy'dushhij den' .
- **Opatrnost je ovšem na místě například na některých přemostěních, kde může být povrch namrzlý a kluzký.**
  Осторожность но на месте например на некоторых мост , где поверхность псих и скользкая .
  Ostorozhnost' no na meste naprimer na nekotory'x most , gde poverxnost' psix i skol'zkaya .
- **Prostě je ignoruji.**
  Просто их я игнорирую .
  Prosto ix ya ignoriruyu .
- **Podle doktorky Christiane Martelové není quebecký zdravotnický systém dostatečně výkonný, aby zajistil přístup všech osob ke kvalitní paliativní péči, než bude možno souhlasit s provedením eutanazie.**
  По доктору Христиан Мартел не қуэбэцкый здравоохранение система достаточно исполнительные , чтобы он обеспечил доступ всех людей к качественный Паллиативное уходу , než он будет возможно согласен с выполнением эвтаназии .
  Po doktoru Xristian Martel ne que'be'czky'j zdravooxranenie sistema dostatochno ispolnitel'ny'e , chtoby' on obespechil dostup vsex lyudej k kachestvenny'j Palliativnoe uxodu , než on budet vozmozhno soglasen s vy'polneniem e'vtanazii .

B. Data on the attached disk

I am attaching a hard drive to this thesis.[1] For a better compatibility, the drive is formatted with NTFS.[2]

The disk has several subfolders:

- `corpora` for the corpora
- `systems` for the systems and experiments
- `evaluation` for the some evaluation scripts (that include the results of the translation)
- `thesis` for XƎLaTeXsource code of this thesis plus its PDF version

All the included scripts are mostly experimental and, as most of used systems and frameworks themselves (Moses, TectoMT, GNUstep...), they are not easy to run. I have *not* tried to run any of the experiments anywhere else than on the ÚFAL network. [3]

For reference: ÚFAL network is made of 64-bit Ubuntu 10.04 LTS installations, with perl 5.10 and Sun Grid Engine installed.

Some of the corpora and systems have special licenses that *don't allow them to be shared*; for example, in the Intercorp license agreement, I had to sign that:

> "The User agrees not to re-distribute or otherwise make publicly available
> the SCD, or any derivative work based on it"

I also include a VMWare virtual machine with pre-installed Microsoft Windows (that I don't have legal permission to share) and PC Translator (that I don't have legal permission to share).

My understanding of Czech copyright law is that it's legal to share such data in academic, non-commercial purposes, such as attaching them to a thesis on a hard drive.

B.1 Corpora

The folder `corpora` has several subfolders:

---

[1] Because of technical difficulties, I have only one copy instead of three.

[2] For some reason, Ubuntu's Nautilus refuses to display some of the folders, while `ls` seems to work fine. I do not have resources to investigate this further; it might be connected with the fact some of the folders were created on Mac OS X.

[3] Except for the VMWare/PC Translator setup, that I have run only on my personal computer.

- `original_data` for the raw, original data, as downloaded
- `scripts` for some of the extraction scripts
- `cleaned_data` for already filtered corpora
- `unused` for the unused data

B.1.1 Original data

WMT

Both WMT test sets are in the folder `wmt`. The files were downloaded from `http://www.statmt.org/wmt13/translation-task.html`.

`wmt/test_2013.tgz` has several SGML[4] files for every language in the competition. With every document, information about original language is included.

`wmt/test_2012.tgz` includes more languages and even previous years.

The previously mentioned webpage is also saved in the `wmt/wmt.html` file.

Intercorp

`intercorp/mixed.gz` is a gzipped text file with all the data from the mixed corpus. Each line has both Czech and Russian text, divided by a tabulator.

`intercorp/filtered/data.tgz` is all the Intercorp data.[5]

The tarred and gzipped file includes `intercorp_shuff_cs` and `intercorp_shuff_ru`, that include the book data (both sentences and metadata) in a strange, XML-like format. The sentences in the books are shuffled.

The file `intercorp_shuff_ru2cs` includes the linking of the sentences.

UMC

UMC corpus is in the files `umc/umc-0.1-corpus.zip` and `umc/umc003-cs-en-ru-triparallel-testset.zip` zipped, as downloaded from the UMC website, that's also saved in the folder `umc/doc`.

In UMC 0.1, all that matters to us is the file `Czech-Russian.1-1.txt` with the sentences that are linked to one another.

In UMC 003, the sentences are strangely mixed (and the `README` file is not entirely accurate) and strangely lowercased. The only non-lowercased text is in the file `all/ps2009.tok.csenru.gz`.

---

[4] As far as I know, SGML is a superset of XML; however the files seem like well-formed XML; not valid, because the DTDs are not present

[5] This data source is under a license agreement, that's in the `License_Agreement.odt` file.

Wikipedia titles

As I already mentioned in 3.1.5, wikipedia now use a different format of inter-language linking somewhere in 2013, where my old script no longer works. I do not have the original dump; I, however, have an older 2012 dump on which my script works.

The dump is in the file `wiki/cswiki-20121112-pages-articles.xml.bz2`

News Crawl

All News Crawl corpora are in the folder `newscrawl`. The files are exactly as downloaded from the page already mentioned in the section WMT.

The files are tarred and gzipped in the `training-monolingual-news-2008.tgz` (and similar for other years). There are more language files in each of them.

Common Crawl

Common Crawl is in the folder `commoncrawl`. The file is exactly as downloaded from the page already mentioned in the section WMT.

The files are tarred and gzipped in the file `training-parallel-commoncrawl.tgz`.

We use only `commoncrawl.ru-en.ru` with the Russian text, but in the file `commoncrawl.ru-en.annotation`, there are links to sources of all the data.

Yandex

Yandex data are in the `yandex` directory, tarred and gzipped in the `corpus.en_ru.1m.tgz` file. The file contains just two text files – one for English (that we don't use) and one for Russian.

Subtitles

Subtitles are as given to us, in the folder `subtitles`.

B.1.2 Scripts

Most of the data require only some very easy one-liners to prepare; I have included only the more complicated scripts.

As I mentioned before, those scripts were intended for one-time use on specific computers and I am not guaranteeing their reusability; however I am including them for completeness.

Intercorp

Scripts for extracting Filtered Intercorp data (3.1.2) from the original data are in the `intercorp` directory.

Following must be done before running any of the scripts:

- the `intercorp_shuff_ru` has to be corrected to be well formed XML by enclosing with a big `<all>` tag; also some other minor corrections (like replacing `&` with `&amp;` and so on), and saved as `intercorp_shuff_ru.corrected`, similar with the Czech data
- the `intercorp_shuff_ru2cs` data has to be split into separate XML files (for example by using UNIX `split`) and saved as `xx01` to `xx86`

The scripts then do the following:

- `make_info.pl` parses the corrected XML and prints metadata about the books in YAML into `info.yaml`
- `extract_text.pl` extracts the text from one of the `xx` books and prints it in `splitbooks` directory; first argument to the script is the number of the book.
- `are_used.pl` takes the data in `info.yaml`, the directory `splitbooks` and a sorted corpus (its address is the first argument) and detects which books were and which weren't used in the corpus
- `direct.pl` takes the info from the last script and determines, which books were not used and were direct translations of each other, instead of translation from third language, and prints this information

Subtitles

The "script" is actually the whole FilmTit project (**??**) with a lot of unrelated modules, but also with a subtitle alignment module. For running it, we need correctly set-up Maven.[6]

`align_files.sh` runs the correct module; the input and output directories are defined as variables at the top of the script.[7]

---

[6] Scala is probably not needed as Maven installs it correctly based on the pom.xml file. The project is tested only with Maven 2, see next footnote.

[7] This script hasn't been run in two years, so it is highly untested.

Wikipedia

`extract.pl` is a perl script, that extracts the title pairs from a Czech wikipedia dump.[8]

B.2 Systems

In the folder `systems`, there are several systems available.

B.2.1 RUSLAN

I did not find a working RUSLAN copy, so I am not providing that; I am, however, including a RUSLAN dictionary plus the experiments, described in **??**.

B.2.2 Česílko 1.0

Česílko 1.0 is in the folder `cesilko10`. I haven't done any experiments with the code, as noted in 4.1.2.

B.2.3 Česílko 2.0

My slightly fixed version of Česílko 2.0 is in the folder `cesilko20`; the difference between the original code and my slightly fixed code is in the file `code_diff`.

B.2.4 Online systems

Scripts for online systems are in the `online_systems` directory; the java library for Google Translate, the PHP script for Bing Translator and a simple shell script for Yandex Translate.

B.2.5 PC Translator

PC Translator virtual machine and helper script are included in the directory `pc_translator`.

- `vmware` is the VMWare machine itself, with working Windows XP and PC Translator v14.
- `sharedfolder` is a folder that has to be set up in VMWare Tools on the Windows XP machine correctly as `sharedfolder`

---

[8] As noted elsewhere, the dump has to be in an old, pre-2013 format, like the file that's on the disk.

- `run/do_csru.pl` is the script, that tries to start to convert STDIN into PC Translator-friendly format, then moves the AutoHotkey script into the shared folder, then starts the machine that – if set up correctly – will start PC Translator, translate the file and turn itself off.

I have not tested this script outside of my own computer, so I am not sure how well it works.

B.2.6 Moses

I am providing a standard `eman` playground (together with `eman` version I have been using). The experiments relevant to this thesis are written in the `relevant_experiments` file.

B.2.7 TectoMT

I am also providing the latest TectoMT version, together with the `share` folder that actually includes my models.

Numbers of revisions in TectoMT SVN, that are my improvements, are written in the `my_revisions` file.

B.3 Evaluation

In the folder `evaluation`, I include

- the actual results of the translation, plus my small perl script for BLEU evaluation (and generating the examples for section A)
- the PHP script for human evaluation (plus SQL dump of the server)
- slightly modified TrueSkill