Computing and Informatics, Vol. 35, 2016, 441–469

AUTOMATED IMPLEMENTATION PROCESS OF MACHINE TRANSLATION SYSTEM FOR RELATED LANGUAGES

Jernej Vičič, Petr Homola, Vladislav Kuboň

Primorska Institute for Natural Sciences and Technology
Muzejski trg 2, 6000 Koper, Slovenia
&
Charles University Prague, Institute of Formal and Applied Linguistics
Malostranské náměstí 25, 118 00 Praha 1, Czech Republic
e-mail: jernej.vicic@upr.si, phomola@gmail.com, vk@ufal.mff.cuni.cz

Abstract. The paper presents an attempt to automate all data creation processes of a rule-based shallow-transfer machine translation system. The presented methods were tested on four fully functional translation systems covering language pairs: Slovenian paired with Serbian, Czech, English and Estonian language. An extensive range of evaluation tests was performed to assess the applicability of the methods.

Keywords: Machine translation, related languages, shallow transfer, automatic data creation

Mathematics Subject Classification 2010: 68T50

1 INTRODUCTION

The field of automatic translation of natural languages witnessed a major breakthrough in the last decade. The paradigm of a Statistical Machine Translation – SMT [8, 36] and especially its most successful method, a direct translation of small pieces of text (phrases) with the help of the evidence found in huge volumes of parallel bilingual data, became a mainstream research direction. The successful systems represented by Google Translate [20] even managed to persuade the general

public that translating natural languages may be relatively successfully performed automatically. This is indeed a major achievement, although the quality of the translation results still lags behind a quality achieved by a skilled human translator. The paradigm is further described in Section 2.3.

On the other hand, the success of the phrase-based statistical approach to machine translation also negatively affected the research still relying on more traditional approaches. Although there are specific research areas where the traditional approaches (relying mostly on handcrafted resources – dictionaries, grammars, translation rules – we are going to use the term rule-based methods in the sequel) might still compete with the statistical paradigm, many researchers think that the effort invested into building a rule-based system must be much bigger than the effort invested into gathering parallel data and using them in a statistical system based for example upon the Moses [27] platform.

In this article we would like to show that this obstacle in the slow process of development of rule-based machine translation systems, namely the amount of human labour which is necessary for creating grammar rules and dictionary items, can be overcome. We are going to demonstrate it by means of a system aiming at translation between related languages, because even this special category of systems which may exploit a similarity of related languages, faces the issue. The systems for machine translation between related languages typically use simplified architecture and exploit the similarity of languages by means of the application of shallow grammar and transfer rules, but even those rules require substantial effort. This article presents an attempt to automate all data creation processes of a rule-based shallow-transfer machine translation system and its background.

In our experiments, we have intentionally avoided using language pairs which are too closely related (and therefore too similar), such as Czech and Slovak, or Serbian and Croatian. Very close similarity might actually cause some bias in our experiments – we are aiming at methods generally applicable regardless of a degree of relatedness. As it has already been shown in several papers (see, e.g. [24] for the translation from Czech to Slovak, Polish and Lithuanian, and [25] for Czech to Slovak and Russian), the results for Czech to Slovak translation are much better than the results for the translation from Czech to other Slavic languages. On top of that, the modules used in the Czech to Slovak system are much simpler, they rely on morphological and syntactic similarity much more than the modules used in other language pairs.

Several methods that automate some parts of the shallow transfer Rule Based Machine Translation (RBMT) system construction have been presented and are even used as part of the construction toolkits like Apertium [12], which is a widely used open source toolkit for creating machine translation systems between related languages. Parts of the creation process have been addressed by several authors. Let us mention for example automated monolingual dictionary extraction [19]; support for agglutinative languages [5]; Part Of Speech – POS, defined in Section 2.2, tagger training [42, 23, 7]; automatic induction of shallow-transfer rules [41]; automatic extraction of bilingual dictionaries [10]. Some of these technologies have been used

in our experiments along with newly developed methods. All methods and materials discussed in this article were tested on a fully functional machine translation system based on Apertium.

It may be argued that building an SMT system would be a natural choice given that we aim at an automatisation of the process of creation an MT system for a new language pair. The unsupervised stochastic approach has, however, a couple of drawbacks that cannot be ignored; the SMT systems, to be successful, require huge amount of parallel texts [35] that is available only for very few widely used languages like English, Spanish, French, Arabic, Chinese, etc. The performance is worsened when the target language is a language with the free word order or with rich flection, and a set of properties what is typical for Slavic languages, the language group we are primarily targeting.

Our preference for rule-based methods is natural. Such approach provides a number of advantages, such as precise traceability of the translation procedures and easy updating [18] and debugging. Unfortunately, systems based on RBMT methods are inglorious for the high cost of language data production [3].

The article is organised as follows: The state of the art is presented in Section 2, the presentation of methods used in our experiments can be found in Section 3.1. The description of the methods used is described in Section 4, the evaluation methodology with results is presented in Section 5, and the paper is concluded with a discussion in Section 6.

2 STATE OF THE ART

The research presented in this paper is within the scope of Fully Automatic Machine Translation (FAMT), which comprises every automatic translation of natural languages with no user intervention [15]. More specifically, the research focuses on the translation of related languages, one of the most suitable paradigms for this domain is the Shallow Transfer Rule-Based Machine Translation. It has a long tradition and it has been successfully used in a number of MT systems, some of which are listed in Section 2.1.

One of the methods, which guarantees relatively good results for the translation of closely related languages is the method of a rule-based shallow-transfer approach. It has a long tradition and it has been successfully used in a number of MT systems, some of which are listed in Shallow-transfer systems usually use a relatively linear and straightforward architecture where the analysis of a source language is usually limited to the morphemic level.

Figure 2 shows the architecture of the most popular translation system for related languages Apertium [12] and its predecessor, Česílko [22], designed primarily for the translation between Slavic languages.

To the authors' knowledge, there were no experiments that have tried to automatically construct all linguistic data for a fully functional shallow transfer RBMT system, other than the already presented attempts in [48].

Parts of the creation process have been addressed by several authors such as Automated lexical extraction [19]; Support for agglutinative languages [5]; POS tagger training [42, 23, 7]; automatic induction of shallow-transfer rules [41]; automatic extraction of bilingual dictionaries [10]. Some of these technologies are used in this paper along with newly developed methods. They are implemented by means of an open source system Apertium, [12], which presents an optimal platform for further linguistic exploitation and improvement of the data and algorithms as all the data are organised in a transparent manner using XML format. The advantages of an XML are standardisation, readability and editability by humans. The fact that Apertium is available as open source enables easy inclusion of new methods into the existing tools and its modular design supports adding new modules into the existing translation pipeline and changing the overall design.

2.1 Existing MT Systems for Related Languages

A number of experiments in the domain of machine translation for related languages have led to the construction of more or less functional translation systems. The systems are ordered alphabetically:

- Altinas [2] for Turkic languages.
- Apertium [12] for Romance languages.
- Dyvikl, Bick and Ahrenberg [14, 6, 1] for Scandinavian languages.
- Česílko [21], for Slavic languages with rich inflectional morphology, mostly language pairs with Czech language as a source.
- Ruslan [37] full-fledged transfer based RBMT system from Czech to Russian.
- Scannell [43] for Gaelic languages; between Irish (Gaeilge) and Scottish Gaelic (G'aidhlig).
- Tyers [46] for the North Sámi to Lule Sámi language pair.
- Guat [48] for Slavic languages with rich inflectional morphology, mostly language pairs with Slovenian language.

The experiments presented in this paper are based on technologies presented by [12] and [21].

2.2 Part of Speech and Morphosyntactic Tagging

Part of Speech (POS) tagging is the process of marking up the word forms in a text as belonging to a particular class defined as part of speech, based on both its definition and the particular context (usually in a sentence). Morphologically rich languages (e.g. Slavic, Romance and other languages) use an extended set of tags usually called Morphosyntactic tags (MSD) where other descriptors are added to the basic Category of the word like Type, Gender, Number, Case. An example of such tagset

is presented in [16]. The tagging process can be automated by tools such as [7] and [42]. An example of a tagged sentence is presented in Figure 11, the *ana* tag is the MSD descriptor of the adjacent word.

2.3 Statistical Machine Translation – SMT

Statistical machine translation is based on parametric statistical models, which are constructed on bilingual aligned corpora (training data). The methods focus on looking for general patterns that arise in the use of language instead of analyzing sentences according to grammatical rules. The main tool for finding such patterns is counting a variety of objects – statistics. The main idea of the paradigm is to model the probability that parts of a sentence from the source language translate into suitable parts of sentence in the target language.

2.4 The Finite-State Rules

Definition 1. The rules consist of pairs: $< pattern, action >; pattern \equiv Lc_i \circ b \circ Lc_{i+1} \circ b \circ \ldots; action \equiv actions(pattern)$.

A pattern is denoted by a sequence of a variable length of lexical categories (defined in Definition 13) of the source language, separated by blanks (b – blank). The action denotes actions that must be applied to the source pattern and the output pattern of the lexical categories of the target language. The actions defined in the main part of the rule are applied after detection of the source pattern.

The structural transfer module of Apertium uses finite-state pattern matching to detect fixed-length patterns of lexical forms (chunks or phrases) needing special processing due to grammatical divergences between two languages (gender and number of changes to ensure agreement in the target language, word reordering, lexical changes such as changes in prepositions, etc.) and performs the corresponding transformations. In the experiment we used the same type of rules, but only looking for the agreement in all possible lexical forms of words in a local context.

An example of a rule is presented in Figure 1. A rule consists of two parts: *pattern* and *action*. Patterns are usually expressed in terms of lexical categories, for instance, "article-noun" or "article-noun-adjective". The *action* part determines what action should be executed on particular pattern. The *out* part deals with actual output generation.

2.5 Statistical Language Model

A statistical language model assigns a probability to a sequence of words by means of a probability distribution. Language modeling is used in many natural language processing applications such as speech recognition, machine translation, part-of-speech tagging, parsing and information retrieval. We use a simple language model based on tri-grams (trained on word forms without any morphological annotation)

```
<rule>
  <pattern>
    <pattern-item n="CAT__adj_5"/>
    <pattern-item n="CAT__n_4"/>
  </pattern>
  <action>
    <out>
      <lu>
        <clip pos="1" side="tl" part="lemh"/>
        <clip pos="1" side="tl" part="adj_5_0"/>
        <clip pos="2" side="tl" part="n_4_1"/>
        <clip pos="2" side="tl" part="n_4_2"/>
        <clip pos="2" side="tl" part="n_4_3"/>
        <clip pos="1" side="tl" part="adj_5_4"/>
      </lu>
      <b pos="1"/>
      <lu>
        <clip pos="2" side="tl" part="lemh"/>
        <clip pos="2" side="tl" part="n_4_0"/>
        <clip pos="2" side="tl" part="n_4_1"/>
        <clip pos="2" side="tl" part="n_4_2"/>
        <clip pos="2" side="tl" part="n_4_3"/>
      </lu>
    </out>
  </action>
</rule>
```

Figure 1. An example of a local agreement rule based on the Apertium structural transfer rule scheme. The pattern matches all adjectives with 5 lexical categories adjacent to noun with 4 categories. The action part outputs two lexical units (<lu>), first the adjective which has 3 lexical categories identical to the noun (n_4_1, n_4_2, n_4_3). The words agree in three categories (1 = gender, 2 = number, 3 = case).

which is intended to sort out "wrong" target sentences (these include grammatically ill-formed sentences as well as inappropriate lexical mapping).

3 ARCHITECTURE OF THE TRANSLATION SYSTEM

In order to guarantee a uniform and transparent implementation environment we have decided to use the Apertium [38] shallow-transfer machine translation toolbox for our experiments although most of the methods could be applied to other systems as well. Because the results described in [24] clearly indicate that the architecture omitting the POS tagger at the beginning and using non-disambiguated morphological analysis of the source text relying at the stochastic ranker at the end of the processing pipeline provides better results than the original architecture introduced

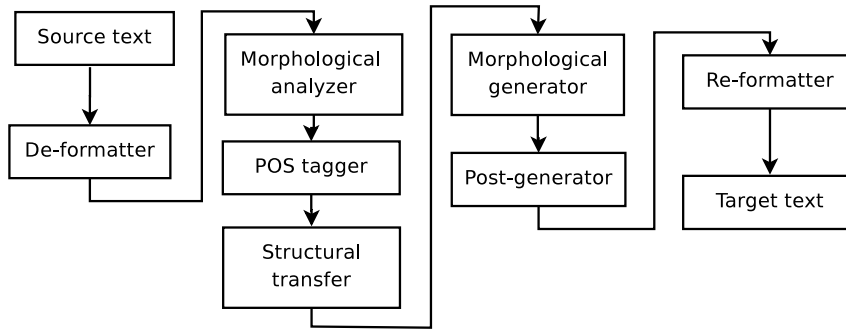

Figure 2. The modules of a typical shallow transfer translation system

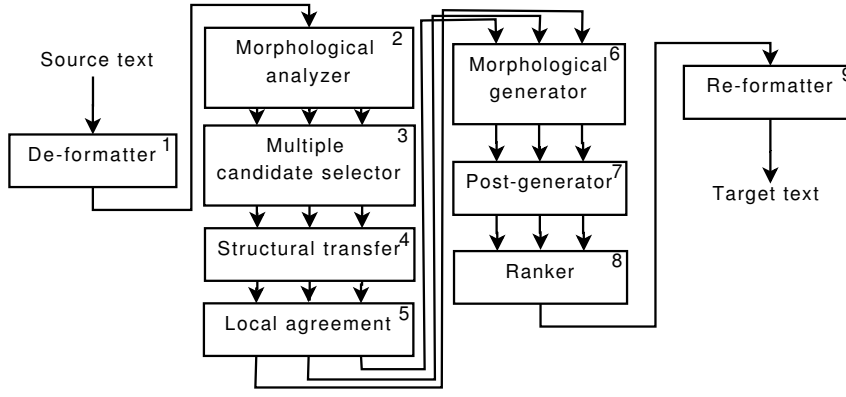

Figure 3. The modules of the proposed (changed) shallow transfer translation system. The system follows the basic design of [12, 22] with the omission of the POS tagger and the inclusion of the ranker.

in [21] and later used also in Apertium [12], we have decided to use the architecture described in Figure 3 for our experiments.

3.1 The Overview of the Used Data Types

For all modules of the system we need the following types of data:

1. Monolingual source dictionary with morphological information for source language parsing.
2. Monolingual target dictionary with morphological information for target language generation.
3. Bilingual translation dictionary.
4. Shallow transfer finite-state rules.

5. Finite-state local agreement rules of the source language.
6. Finite-state local agreement rules of the target language.
7. Statistical target language model[1].
8. Modelled source language tags[2].

The monolingual dictionaries, defined in Definition 8 and presented in Section 4.1, are used in the shallow parsing of the source text by the morphological analyser module, number 2 in Figure 3, and in the generation of the translated text in the target language by the morphological generator module, number 5 in Figure 3.

The bilingual dictionary, defined in Definition 13 and presented in Section 4.2, is used for word-by-word and phrase translation, in our case the translation is based on lemmata. The shallow transfer rules, defined in Definition 1 and presented in Section 2.4, are used to address local syntactic and morphological rules such as local word agreement and local word reordering. The module using the bilingual dictionary and the shallow transfer rules is the structural transfer module, number 4 in Figure 3.

The finite state local agreement rules of the target language are used in the local agreement module in order to eliminate errors produced by shallow transfer rules in the transfer phase module, number 5 in Figure 3. The method of this module is basically the same as in the Structural transfer module except using rules that discover only local agreement and it is used on the output of that module.

The language model, presented in Section 2.5, of the target language is used in the last stage of the pipeline by the ranker module, number 8 in Figure 3. The ranker chooses the best translation candidate from the list of possible translation candidates produced by the previous modules, on the basis of a statistical target language model.

The multiple candidate selector, number 3 in Figure 3, uses the source language morphological tags and the finite-state local agreement rules of the source language. The method uses the same technology as the Structural transfer module with local agreement rules induced from the source language as a heuristic to restrict the blow-up of hypotheses of the non-disambiguated morphologically analysed input and the same technology as the Ranker but the learning data is presented in the form of POS tags (modeled source language tags) rather than original words for scoring the morphologically analysed source language text in order to restrict the number of possible translation candidates. The method is presented into greater detail in [26].

Each item from the list was addressed by applying a known method or by introducing a new method. The methods are presented in more detail in Section 4.

A fully functional system was constructed using presented methods. Its overall performance was evaluated in Section 5.

[1] Used in the module of the stochastic ranker which extends the original system and which has been described in [24]

[2] Used in the Multiple candidates selector module proposed by the extension of the original system [12] and described in [26]

4 USED METHODOLOGY

Each module from Figure 3 consists of the basic software and language-specific data. The data in Apertium format is structured in human readable XML format. The following subsections present the descriptions for the data creation process for each module.

4.1 Monolingual Source and Target Dictionary Creation

Definitions 2 through 8 present the formal definition of the morphosyntactic dictionary.

Definition 2. Word w is composed of the prefix pr, stem s and postfix po: $w = pr \circ s \circ po$. $\circ$ defines a concatenation of two strings.

Figure 4 shows an example how the word miza (table) is decomposed.

```
word: miza
stem: miz
prefix: /
postfix: a
miza = / - miz - a
```

Figure 4. A word is composed of a prefix, a stem and a suffix

Definition 3. Morphosyntactic description MSD is a string that defines the morphosyntactic classes of a word as defined in Definition 2. Each word w is assigned an appropriate morphosyntactic description MSD. All words with the same MSD are grouped in one set. $w \in MSD_m \leftrightarrow m$ is the MSD of w.

Figure 5 shows a MSD example how word forms are grouped into MSD sets.

```
MSD:
Ncfsn - Noun common feminine singular nominative
Ncnsn - Noun common neuter singular nominative
Vmep-sm - Verb main perfective participle singular masculine
words:
miza, hi\v{s}a:  Ncfsn
drevo, kolo: Ncnsn
kupil: Vmep-sm
```

Figure 5. Word forms belong to one MSD set

Definition 4. A lexeme Le is a unit of lexical meaning. Lexeme Le is a set of word forms with the same meaning. A set of rules R_{Le}, which use the same stem and add prefixes and suffixes, defines all word forms for the lexeme Le. The set of rules R_{Le} is composed of the rules in the form: $r \in R_{Le}; r \equiv pr \circ s \circ po \Rightarrow MSD_m$. The first part of the rule is a pattern and all words forms that match this pattern belong to the set defined by the MSD on the right side of the rule. The rules usually prevent the inclusion of synonyms into one lexeme although technically this is possible.

Definition 5. A typical representative of a lexeme Le is a canonical form of a word named lemma lm.

Definition 6. A word form w belongs to a lexeme Le, if there exists a rule: $r \in R_{Le}$, for which the following statement is true: the pattern of the rule must define the word w and the right part of the rule, the MSD, defines the set the word belongs to. $w \in Le \Rightarrow \exists r \in R_{Le}; w = pr_i \circ s_i \circ po_i; r \equiv pr \circ s \circ po \Rightarrow MSD; pr_i = pr; s_i = s; po_i = po; w \in MSD_m; MSD = m$.

Definition 7. A paradigm P is a set of all lexemes with the appropriate set of rules that differ only by stems: $\forall Le_a, Le_b \in P : \forall r_a \in R_{Le_a} \exists r_b \in R_{Le_b} : r_a \equiv pr \circ s_a \circ po \Rightarrow MSD \wedge r_b \equiv pr \circ s_b \circ po \Rightarrow MSD$.

Figure 6 shows two example paradigm excerpts and two lemmata that link to the paradigms.

```
Paradigm:
žog/a__n (noun, )
change a into a feminine singular nominative
change a into e feminine singular genitive
change a into i feminine singular dative
...
življenjsk/i__adj (adjective, )
...
words:
<e lm="miza"><i>miz</i><par n="žog/a__n"/></e>
lemma = miza
stem = miz
paradigm = žog/a__n

<e lm="abramski"><i>abramsk</i><par n="življenjsk/i__adj"/></e>
lemma = abramski
stem = abramsk
paradigm = življenjsk/i__adj
```

Figure 6. Two example paradigm (žog/a__n, življenjsk/i__adj) excerpts and two lemmata that link to the paradigms

Definition 8. A morphosyntactic dictionary is composed of a set of paradigms (P) from the Definition 7.

The entries in the monolingual (morphosyntactic) dictionary are organized in morphological paradigms, as defined in [44]. Morphological paradigms are classes that contain all lemmata that share the same behavior (regarding all possible word forms). In other words, it contains all lemmata that change in the same manner for all the applicable MSD tags. Figure 7 shows a sample paradigm for nouns in female gender in Slovenian.

```
<pardef n="cerk/ev__n">
  <e><p>
    <l>ev</l>
    <r>va<s n="n"/><s n="f"/><s n="du"/><s n="gen"/></r>
  </p></e>
  <e><p>
    <l>ev</l>
    <r>ve<s n="n"/><s n="f"/><s n="pl"/><s n="nom"/></r>
  </p></e>
  <e><p>
    <l>ev</l>
    <r>vo<s n="n"/><s n="f"/><s n="sg"/><s n="acc"/></r>
  </p></e>
  <e><p>
    <l>ev</l>
    <r>ev<s n="n"/><s n="f"/><s n="sg"/><s n="nom"/></r>
  </p></e>
  <e><p>
...
</pardef>
```

Figure 7. A part of a sample paradigm for nouns in female gender in Slovenian. The sample lemma is cerkev – church. The ending -ev changes according to the possible MSD variants.

The data can be compacted using paradigms as shown in the following example: if we take an example from English which would handle the correct forms in the past tense; the transformation of regular verbs (as e.g. the word walk – walked) can be achieved by a morphological transformation rule (for past tense). This simple rule accompanied only by a list of irregular words and their specific forms in the past tense (as e.g. sleep – slept) is everything what we need for a given purpose. For languages that employ concatenative morphology[3] such as the majority of European languages, different forms of the same word are realised by changing the prefix

[3] words are composed of multiple morphemes concatenated together; the morphemes include the stem plus prefixes and suffixes

and/or the suffix of the word. Thus the Czech adjective *sladký* (sweet) can be derived into *nej-slad-ší-ho* (sweetest – masculine or neutral form in the genitive or accusative case), by adding the prefix nej- representing the superlative, by changing the suffix -ký (comparative) to the suffix -ší and by adding a case ending -ho for masculine and neutral gender. An example of this phenomenon, although only for the suffixes, in Slovenian is shown in Table 1.

Word Form	Number	Gender	Case
mest-o	singular	neuter	nominative
mest-a	singular	neuter	genitive
mest-u	singular	neuter	dative
mest-o	singular	neuter	accusative
mest-u	singular	neuter	locative
mest-om	singular	neuter	instrumental
mest-a	plural	neuter	nominative
mest-∅	plural	neuter	genitive
mest-om	plural	neuter	dative
mest-a	plural	neuter	accusative
mest-ih	plural	neuter	locative
mest-i	plural	neuter	instrumental
mest-i	dual	neuter	nominative
mest-∅	dual	neuter	genitive
mest-oma	dual	neuter	dative
mest-i	dual	neuter	accusative
mest-ih	dual	neuter	locative
mest-oma	dual	neuter	instrumental

Table 1. All word forms for Slovenian lemma mesto (place/city). The word forms change with suffixes.

4.1.1 Paradigm Creation

The words were grouped into paradigms in order to deal with multiple word forms as both Slovenian and Serbian are highly inflectional languages. Each paradigm is represented by:

- typical lemma – the lemma the paradigm was constructed from,
- stem – the longest common part of all words in the lemma,
- set of all words split into stems, prefixes, suffixes and Morpho-Syntactic Descriptors (MSDs) [16].

The Slovenian language uses only one prefix – *naj* for the superlative form of the adjective (dober – boljši – najboljši). An example of a paradigm is shown in Figure 8.

The annotated lexicons, the lists of unique words with lemma descriptor and MSD, were extracted from corpus for both languages and the paradigms were constructed using Algorithm 1.

```
lemma: cerkev
stem: cerk
example entries:
word form: cerkev
suffix: ev
MSD: noun+feminine+singular+nominative
word form: cerkvah
suffix: vah
MSD: noun+feminine+plural+locative
```

Figure 8. A part of a paradigm *cerkev* – church. Lemma: cerkev, stem: cerk, two word forms *cerkev* and *cerkvah*

All of the word forms of a lemma present in the corpus are grouped into a class representation of that lemma. For each lemma a paradigm is constructed from each class. Two paradigms are joined together if the lemmata of both paradigms have the same POS tag and if the entries, pairs of suffix and MSD, of one paradigm present a complete subset of the compared paradigm. The complexity of this pro-

Algorithm 1 Paradigm construction algorithm

Input: *lemmata*
Output: *paradigms*
$paradigms = \emptyset$
for all $lemma \in lemmata$ **do**
  $para \leftarrow$ create paradigm from $lemma$
  $paradigms \leftarrow paradigms + para$
end for
for all $p1 \in paradigms$ **do**
  for all $p2 \in paradigms$ **do**
    $joinParadigms = true$
    for all $(z1, z2) \in p1.entries \times p2.entries$ **do**
      if $p1.POS = p2.POS \ \wedge$
       $z1.MSD = z2.MSD \ \wedge$
       $z1.ending \neq z2.ending$ **then**
        $joinParadigms = false$
      end if
    end for
    if $joinParadigms$ **then**
      join paradigms $p1$ and $p2$ into paradigm $p1$
      $paradigms \leftarrow paradigms \setminus p2$ ▷ remove p2 from paradigms
    end if
  end for
end for

cess increases linearly as the number of lemmata in paradigms increases by joining paradigms. The information about all lemmata that generated the paradigm is stored in a list enabling easy lookup.

The monolingual source and target dictionaries were constructed using a starting lexicon extracted from the corpus and joined paradigms resulting in a lexicon that was roughly 20 times larger than the original lexicon. The new lexicon had (almost) all missing word forms from the corpus.

4.2 Bilingual Translation Dictionary Creation

Definition 9. A lexical category Lc presents a substring of a morphosyntactic description MSD (defined in Definition 3).

Definition 10. A pair $\langle lemma, tag\rangle; lemma \in Lm; tag \in Lc$; where Lc is defined in Definition 9 and Lm is defined in Definition 5, denotes a lemma with the associated string of lexical categories.

Definition 11. Direction of validity $direction \in \{LR, RL\}$; $LR \equiv$ from the source to the target; $RL \equiv$ from target to the source; denotes the translation direction of an entry. Omitted operator denotes arbitrary direction.

Definition 12. A translation pair $Tp \equiv direction << l_s, t_s >, < l_t, t_t >>; l_s, l_t \in Lm; t_s, t_t \in Lc$; where l_s is the lemma in the source language, l_t is the lemma in the target language, t_s is the tag in source language and t_t is the tag in target language, defines a lemma of the source language with the associated lexical category and the appropriate (translated) lemma of the target language with associated lexical category.

Definition 13. A bilingual translation dictionary Bd is a set of translation pairs Tp from Definition 12.

Definition 13 describes entries of a bilingual translation dictionary. The lexical category of the source lemma usually matches the lexical category of the target lemma, particularly often in similar language pairs. The usage of the lexical categories enables the disambiguation of the lemmata with the same name and different meaning.

An SMT word-to-word model [8], using GIZA++ tool [36], was trained on the parallel sentence aligned list extracted from the corpus. The list is shown in Figure 9. Each word in the corpus is represented by the *lemma* (lemma of the word), *ana* (morphosyntactic description – MSD [16]) and the word form used in the corpus. Only the lemma and POS tag, the first part of the MSD, of each word were extracted from the corpus for this task leaving parallel sentences in lemmatised form with the POS tag. Figure 9 shows the prepared data.

The lemmata alignment ensures much better alignment performance due to the search space reduction as described in Equation (1) and in Figure 10. The words

```
priti_V biti_V do_S podrt_A drevo_N,
o_S kateri_P on_P biti_V praviti_V.
```

Figure 9. Prepared data: lemma and POS of each word from the corpus

from the monolingual dictionaries are aligned to the translations (bilingual lemmata pairs) through paradigms that retain the information about the included lemmata, see Section 4.1.1.

The number of word forms in a text is much bigger for highly inflected languages like the Slavic languages. Table 2 shows the difference in the number of word forms for the same corpus [17] and [13] in five languages; three rich inflectional Slavic languages: Slovenian, Serbian, Czech along with English and Estonian for reference. The ratio column shows the ratio between word-forms and lemmata.

Language	Number of Words	Lemmata	Ratio
Slovenian	20 923	7 895	2.65
Serbian	21 505	8 392	2.56
Czech	22 273	9 060	2.46
English	11 078	7 020	1.58
Estonian	18 853	8 679	2.17

Table 2. Number of lemmata in the corpus MULTEXT-EAST [17]

The reduction of search space obviously increases the accuracy of the model (the word-by-word translation model). This result is not surprising, but a lot of information about the word form is lost in the process.

Let us observe the phenomenon to a greater extent. The word alignment model as described in [8, 36] can be used as the basis for a new model that uses *lemma+POS* descriptions of the actual word forms used in the bilingual parallel corpus.

Some simple definitions that will help the formulation of the Equation (1):

- L – language, all words
- E_L – lemmata of the language L
- $E_{L(i)}$ – i^{th} lemma with all word forms (from language L)

$$|L| = \sum_{i=0}^{|E_L|} |E_{L(i)}| \tag{1}$$

The search space is reduced from $|L|$ to $|E_L|$.

Let us look at the example: If we take George Orwell's novel "1984", which comprises the multilingual sentence-aligned part of the [17] corpus as a sample of Slovenian language, we get the values in Figure 10 taken from Table 2. The search space has been reduced from 20 923 word forms to 7 895 lemmata.

Original language $|L| = 20\,923$
Lematised language $|E_L| = 7\,895$

Figure 10. The reduction of the search space for Slovenian (small corpus MULTEXT-EAST [17])

The bilingual parallel annotated corpus [17] comprises original text with additional information in the form of XML tags according to the TEI-P4 [45] and the EAGLES [32] guidelines. An example excerpt is shown in Figure 11.

```
<s id="Osl.2.3.5.11">
  <w lemma="priti" ana="Vmps-dma">Prisla</w>
  <w lemma="biti" ana="Vcip3d--n">sta</w>
  <w lemma="do" ana="Spsg">do</w>
  <w lemma="podrt" ana="Afpnsg">podrtega</w>
  <w lemma="drevo" ana="Ncnsg">drevesa</w>
  <c>,</c>
  <w lemma="o" ana="Spsl">o</w>
  <w lemma="kateri" ana="Pr-nsl----a">katerem</w>
  <w lemma="on" ana="Pp3msd--y-n">mu</w>
  <w lemma="biti" ana="Vcip3s--n">je</w>
  <w lemma="praviti" ana="Vmps-sfa">pravila</w>
  <c>.</c>
</s>
```

Figure 11. A sentence in the corpus

4.3 The Induction of Rules for Shallow Transfer

The shallow, finite-state type transfer rules were constructed using available software from Apertium toolkit. The software is based on the technologies presented in [41]. The basic idea of the process is using statistical methods to construct templates from bilingual aligned corpus. These templates are later translated into finite-state rules in the Apertium format.

4.4 Automatic Induction of Local Agreement Rules

The automatic induction of the local agreement rules produces the same format of the rules as the method described in [41], but the method is limited to the discovery of local agreement. The method discovers only local context of maximum length 3 (using trigram language model). The requirements for the method are much simpler, just a monolingual, morphologically annotated corpus. The local agreement rules were used by two modules of the translation system; the Multiple candidate selector module and the Local agreement module. First module used the Local agreement

rules trained on the source language and the second module on the target language morphologically annotated corpus. The corpus used as training data was [17], which was hand checked for errors in morphosyntactic tags. The corpus is multilingual, all source and target language combinations were covered. Trigrams and bigrams with morphological descriptions were extracted from source and target language part of the corpus for each language pair. Each bigram and trigram was checked for agreement among tags of different words, the tags and their positions were free. If any agreements were found, a candidate for a rule was stored in the form presented in Figure 1. The POS tags of the source bigram or trigram present the pattern part of the rule. The action part of the rule is constructed from all the morphosyntactic tags with agreement information. The rule candidates were grouped according to the pattern and action definitions, each group with a predefined number of candidates was chosen as a valid rule. The threshold for the number of candidates was selected empirically on the basis of a small test; the authors admit that the threshold selection should be further explored. The Algorithm 2 describes this process.

Algorithm 2 The process of automatic rule construction from annotated corpus

Input: $trigrams \leftarrow$ construct bigrams and trigrams of MSDs from corpus;
Output: $allClasses$ (a set of constructed classes)
$allClasses = \emptyset$
for all $trigram \in trigrams$ **do**
  $agreement = false$
  for all $(msd1, msd2) \in trigram.msds \times trigram.msds$ **do** ▷ all pairs
    for all $(cat1, cat2) \in msd1.category \times msd2.category$ **do**
      if $cat1 = cat2$ **then** ▷ the msds agree in the selected categories
        $agreement = true$
      end if
    end for
  end for
  if $agreement$ **then**
    $tempRule \leftarrow rule(cat1, cat2)$ ▷ construct a rule
  end if
  if $r \in allClasses \wedge tempRule = r$ **then**
    $r.count \leftarrow r.count + 1$ ▷ found class r with same POS
  else
    $allClasses \leftarrow allClasses \cup \{tempRule\}$ ▷ new class
  end if
end for
for all $r \in allClasses$ **do**
  if $r.count \leq threshold$ **then**
    $allClasses \leftarrow allClasses \setminus \{r\}$ ▷ delete low frequency classes
  end if
end for

4.5 The Statistical Target Language Model Creation

An essential part of the whole MT system is the statistical post-processor. The main problem with our simple MT process described in the previous sections is that both the morphological analyzer and transfer preserve the morphological and local syntactic ambiguity, their combination then creates a huge number of variants (hypotheses) in the translation process. It would be very complicated (if possible at all) to resolve this kind of ambiguity by hand-written rules. Therefore we have implemented a stochastic post-processor which aims at the selection of one particular sentence that suits best the language model trained on corpora. If corpora selection process is valid, such model should represent the target language. The stochastic ranker selects the sentence that is most likely correct in the target language. The language models have been trained on corpora collected from randomly chosen articles from the Wikipedia of the languages concerned[4]. The size of the corpora was approx. 15 million words for Czech and English languages and approx. 7 million for Estonian and Serbian languages.

4.6 Agreement of Source Language Tags

The agreement of morphological descriptors can be modeled using rules based on regular expressions. The rules are described in Section 2.4. The same format of rules as defined in the Apertium framework was used as it was the most appropriate and already based on the same technology. The automatic induction of such rules is presented in Section 4.4. The mechanism of discarding improbable translation candidates is shown in Algorithm 3. A set of all possible sentences that are candidates for translation is constructed using the translation system. All applicable local agreement rules are applied to each candidate sentence. If a candidate sentence is changed by a rule, that means that words in local context should agree in more lexical categories. Such candidate sentence is discarded.

Algorithm 3 Discarding of (possibly) all improbable candidates for the translation using the agreement rules

Input: $candidateSentences \leftarrow constructAllCandidatesWithModules$
Output: $candidateSentences$ ▷ same set with removed elements
for all $candidate \in candidateSentences$ **do**
  $newSentence \leftarrow applyRules(candidate)$
  if $newSentence \neq candidate$ **then**
    $candidateSentences \leftarrow candidateSentences \setminus candidate$
  end if
end for

[4] http://cs.wikipedia.org, http://en.wikipedia.org, http://et.wikipedia.org, http://sr.wikipedia.org

All applicable rules, where a regular expression describes part of the translation candidate, are applied on the translation candidate. If a rule changes part of the translation candidate, the candidate is discarded.

5 EVALUATION METHODOLOGY AND RESULTS

The methods presented in this paper in Section 4 concentrate on the construction of machine translation systems for related morphologically rich languages. The experiments aim at showing the quality of automatically generated data on a fully functional translation system and also on the usability of presented methods for rapid development of a translation system for a new language pair.

Four fully functional translation systems were constructed and evaluated in this experiment:

1. SL-SR, Slovenian to Serbian translation system,
2. SL-CS, Slovenian to Czech translation system,
3. SL-EN, Slovenian to English translation system,
4. SL-ET, Slovenian to Estonian translation system.

5.1 Description of the Systems

The system using Slovenian-Serbian (SL-SR) language pair was constructed as a pilot system which served for testing of our method in the process of its development. The methods presented in this paper were checked through several iterations (the systematic errors were corrected and the corrections included into the basic framework). This language pair was used to check the quality of the presented methods on a fully functional translation system. Both languages are inflectionally, morphologically and derivationally rich. Although these languages are related, the high degree of inflection of both languages still requires the morphological analysis of the source language and morphological synthesis of the target language.

The SL-CS system was constructed to evaluate the applicability of the methods presented in Section 4 on a new language pair of related languages and to test how quickly a new system can be constructed. The properties of this language pair are very similar to the properties of the first language pair (SL-SR). The system was constructed from scratch in just two days by a single person on an ordinary personal computer[5].

The SL-EN and SL-ET systems were constructed to evaluate the applicability of the presented methods and the overall design to a distant language pair. The results presented in Table 12 and in Sections 5.2.2 and 5.2.3 show a clear decrease of the translation accuracy using the same methodology and same training data. The

[5] A notebook computer with 2 GB of RAM and an Intel Core2 duo processor.

Estonian language was chosen as a distant highly inflectional language and English language was chosen as a linear, distant language.

5.2 Description of the Evaluation Metrics

The evaluation of the translations was performed by means of three evaluation methods, each of them is described in detail in a separate subsection further in this section:

1. The automatic objective evaluation using METEOR [4, 30] metric.
2. The non-automatic evaluation by counting the number of edits needed to produce a correct target sentence from automatically translated sentence.
3. Non-automatic subjective evaluation following [31] guidelines.

The BLEU [39] metric was considered as one of the candidates for the evaluation metrics, but it was discarded as many authors agree that BLEU metric systematically penalises RBMT systems [9, 29] and it is not suited for highly inflective languages. Authors of METEOR [4, 30] state that their system fixes most of the problems encountered using BLEU metric; they state that it correlates highly with human judgment. Unfortunately, in order to use METEOR, additional software had to be written because original software does not support our language pairs.

5.2.1 Automatic Objective Evaluation Using METEOR Metric

The publicly available implementation of the METEOR metric [30] version v0.6 was used. The metric uses stemming mechanism as one of the algorithms that enhance correlation with human evaluation for highly inflectional languages. The stemming mechanism that is a side-product of our translation system was used. The results are presented in Table 12, values marked with * show the METEOR evaluation using no stemming and normal Porter-stem [40] for English language, the other values show METEOR evaluation using proprietary stemming approaches. The last two bars represent the reference translation systems, based on SMT. The Google system [20] was evaluated on the same test-set while the values for Moses [27] system are referenced in [34] tech report.

The bilingual parallel corpus [17] was used in automatic evaluation of translations. K-fold cross-validation [28] was used as the method for estimating the generalisation error as it is most suitable for small data sets. In our case five-fold cross validation was used instead of more frequently used ten-fold cross validation as construction of a fully functional system was not automated. The corpus was divided into five parts, each part consisting of roughly 1 700 sentences. The evaluation consisted of selecting one part of the corpus as testing set and remaining four parts as training set. The translation system was constructed according to the

methodology presented in Section 3.1 using the selected training set. The evaluated values in each fold and the average final values are presented further in the paper.

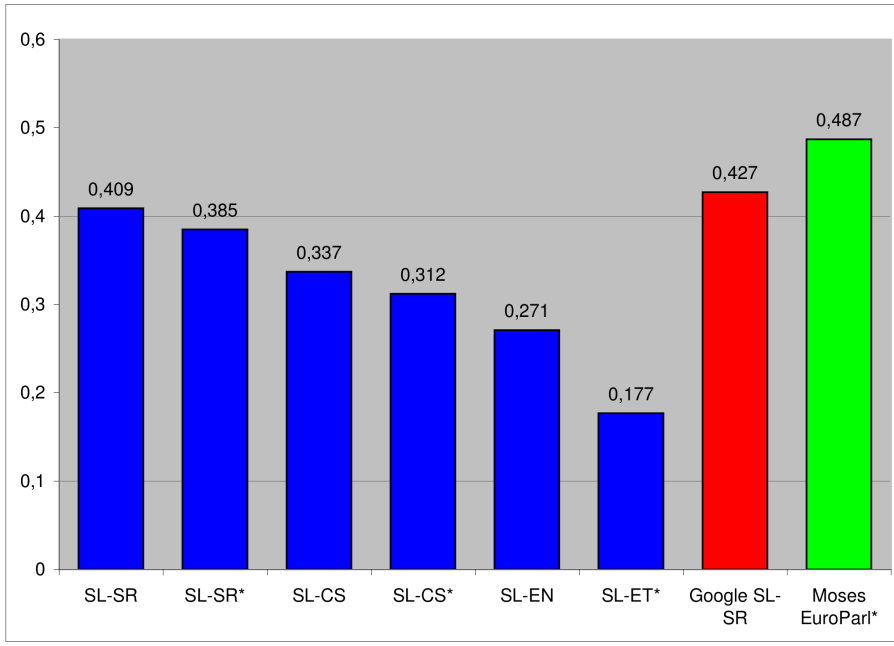

Figure 12. The METEOR metric scores. The evaluation was done using 5-fold cross validation. The values in the figure represent the average values of 5 folds with standard deviation. The evaluations marked with * use Porter-stem or no stemming, the other use proprietary stemming approaches.

5.2.2 Non-Automatic Evaluation Using Edit Distance

The weighted Levenshtein edit-distance [33] or more commonly known as Word Error Rate (WER) was used to count the number of edits needed to produce a correct target sentence from automatically translated sentence. This procedure shows how much work has to be done to produce a good translation. The metric roughly reflects the complexity of the post-editing task.

The evaluation comprised of selecting 200 sentences from the test data, translating these sentences using the translation system and manually counting the number of words that had to be changed in order to obtain a perfect translation. By perfect translation we mean a translation that is syntactically correct and expresses the same meaning as the source sentence.

The evaluations were performed mostly by students and researchers involved in the experiment. The evaluations for Slovenian language and Czech language were performed by two independent native speakers, the evaluations for the English language, Serbian language and and for the Estonian language were performed by one native speaker. The evaluation and the results presented in Figure 13 present the WRR, the Word Recognition rate (1 – WER), which presents the performance of the translation system instead of errors. The values of the evaluation of the systems for Catalan – Spanish (CA – ES) and Catalan – French (CA – FR) [47] are added for comparison. Both systems were manually built, the (CA – ES) system translates between closely related languages and translation quality is much higher than in our system, the translation quality values for (CA – FR) system are on the pair with our best results.

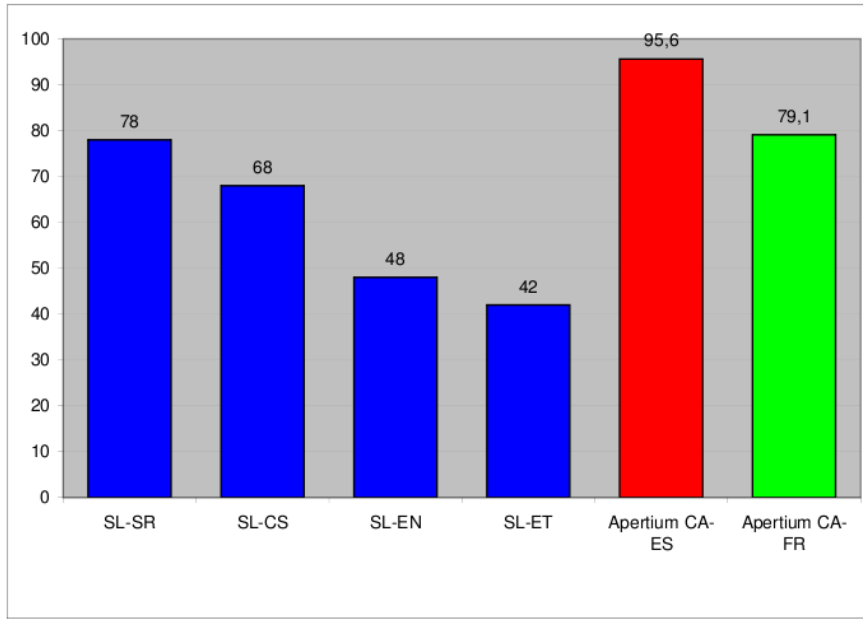

Figure 13. The evaluation results using the Word Recognition Rate metric

5.2.3 Non-Automatic Subjective Evaluation Following [31] Guidelines

Subjective manual evaluation of translation quality was performed according to the annual NIST Machine Translation Evaluation Workshop by the Linguistic Data Consortium guidelines. The most widely used methodology when manually evaluating MT is to assign values from two five-point scales representing fluency and adequacy.

These scales were developed for the annual NIST Machine Translation Evaluation Workshop by the Linguistic Data Consortium [31].

The five point scale for adequacy indicates how much of the meaning expressed in the reference translation is also expressed in a hypothesis translation:

- 5 = All
- 4 = Most
- 3 = Much
- 2 = Little
- 1 = None

The second five-point scale indicates how fluent the translation is. It expresses whether the translation is syntactically well formed. When translating into Serbian the values correspond to:

- 5 = Flawless translation
- 4 = Good target language
- 3 = Non-native target language
- 2 = Disfluent target language
- 1 = Incomprehensible text

Separate scales for fluency and adequacy were developed under the assumption that a translation might be disfluent but contain all the information from the source.

The same test-set of 100 sentences was randomly generated for all four languages. The test data was not used in the linguistic data production process.

Two independent evaluators, native speakers, were used in the evaluation process of the SL-SR and SL-CS systems and one native speaker for the SL-EN and SL-ET systems.

The results are presented in Figure 14. The scores for SL-SR and SL-CS systems are quite high, particularly the adequacy scores, the scores for the remaining two systems are lower due to the non-similarity of the language pairs.

The Table 3 shows a satisfactory to very high inter-rater agreement according to Cohen's kappa coefficient [11].

FF	SL-SR Agreement	SL-CS Agreement
kappa	0.86	0.69
95% Cl	0.70–0.90	0.57–0.81
observed agreement	0.86	0.79
expected agreement	0.300	0.317
examples	100	100

Table 3. The Cohen's kappa coefficient [11] for the SL-SR and SL-CS systems showing satisfactory (SL-CS) to very-high (SL-SR) inter-rater agreement

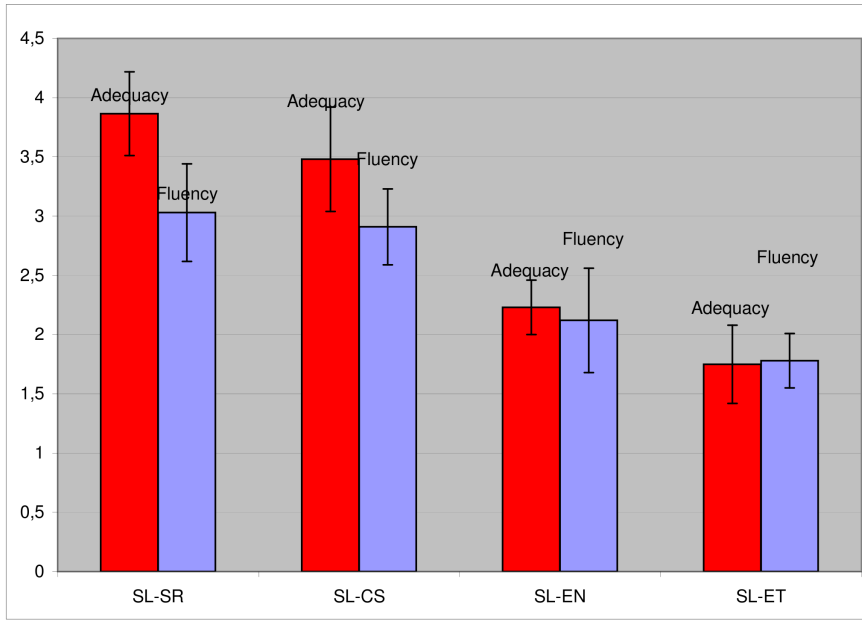

Figure 14. Evaluation results using [31] guidelines. Average values of four independent evaluations show high scores for adequacy and lower values for fluency.

6 DISCUSSION AND FUTURE WORK

The results presented in the paper support the claim that it is possible to use automatic methods for the development of an MT system for related languages. The relatedness constitutes a crucial condition for the success of the whole endeavor – if the languages are not related or if they at least do not have similar morphological and syntactic properties, it is impossible to use the simple architecture presented in the paper and it is necessary to apply standard MT methods. This fact is demonstrated by the results of our MT system for non-related language pairs.

The main contribution of the paper is the experiment documenting that even though some MT systems for related languages required a substantial amount of manual work for each new language pair (the construction of shallow parser or transfer rules), it is possible to replace this manual work by automatic methods which are still able to produce an MT system with acceptable quality. This is a huge step forward which will allow building MT systems for related languages automatically even for languages which have less developed linguistic resources and tools.

Although the results and experiments presented in this paper are encouraging, there is still a vast space for future work which should be directed both at the improvement of individual automatic methods and at the development of the system

as a whole (e.g. further improvements of the architecture). The language models in the experiments are limited to the harvested corpora from the Wikipedia to balance the available corpora for the language pairs. The research of possible improvements using bigger corpora will be done in the future. An interesting topic for further research would also be a question how to proceed in building MT systems for "virgin" languages, i.e. for languages which were still not affected by recent advances in natural language processing and which completely lack the linguistic tools and resources.

REFERENCES

[1] Ahrenberg, L.—Holmqvist, M.: Back to the Future? The Case for English-Swedish Direct Machine Translation. Proceedings of the Conference on Recent Advances in Scandinavian Machine Translation. University of Uppsala, 2004.

[2] Altintas, K.—Cicekli, I.: A Machine Translation System Between a Pair of Closely Related Languages. Proceedings of the 17th International Symposium on Computer and Information Sciences (ISCIS 2002), CRC Press, 2002, p. 5.

[3] Arnold, D.: Computers and Translation: A Translator's Guide. Benjamin Translation Library, 2003.

[4] Banerjee, S.—Lavie, A.: METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Association for Computational Linguistics, 2005, pp. 65–72.

[5] Beesley, K. R.—Karttunen, L.: Finite-State Non-Concatenative Morphotactics. Proceedings of the Workshop of the ACL-SIGCP, Association for Computational Linguistics, 2000, pp. 1–12.

[6] Bick, E.—Nygaard, L.: Using Danish as a CG Interlingua: A Wide-Coverage Norwegian-English Machine Translation System. Proceedings of NODALIDA, Tartu, University of Tartu, 2007.

[7] Brants, T.: TnT – A Statistical Part-of-Speech Tagger. Proceedings of the 6th Applied NLP Conference, Seattle, WA, 2000, pp. 224–231.

[8] Brown, P. F.—Pietra, S. A. D.—Pietra, V. J. D.—Mercer, R. L.: The Mathematics of Statistical Machine Translation: Parameter Estimation. Computational Linguistics, Vol. 19, 1993, pp. 163–311.

[9] Callison-Burch, C.—Osborne, M.—Koehn, P.: Re-Evaluating the Role of BLEU in Machine Translation Research. Proceedings of EACL, Association for Computational Linguistics, 2006, pp. 249–256.

[10] Caseli, H. M.—das Graças V.—Nunes, M.—Forcada, M. L.: From Free Shallow Monolingual Resources to Machine Translation Systems Easing the Task. Proceedings of the Workshop of the ACL-SIGCP, Association for Computational Linguistics, 2008, pp. 41–48.

[11] Cohen, J.: A Coefficient of Agreement for Nominal Scales. Educational and Psychological Measurement, Vol. 20, 1960, pp. 37–46.

[12] Corbi-Bellot, A. M.—Forcada, M. L.—Ortiz-Rojas, S.: An Open-Source Shallow-Transfer Machine Translation Engine for the Romance Languages of Spain. Proceedings of the EAMT Conference, HITEC e.V, 2005, pp. 79–86.

[13] Dimitrova, L.—Ide, N.—Petkevič, V.—Erjavec, T.—Kaalep, H. J.—Tufis, D.: Multext-East: Parallel and Comparable Corpora and Lexicons for Six Central and Eastern European Languages. 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics (COLING-ACL '98), Association for Computational Linguistics, 1998, pp. 315–319.

[14] Dyvik, H.: Exploiting Structural Similarities in Machine Translation. Computers and Humanities, Vol. 28, 1995, pp. 225–245.

[15] EAMT: European Association for Machine Translation, 2010.

[16] Erjavec, T.: MULTEXT-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. Proceedings of the 4th Conference on Language Resources and Evaluation (LREC '04), ELRA, 2004, pp. 1535–1538.

[17] Erjavec, T.: MULTEXT-East Version 4: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. In: Chair, N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D. (Eds.): Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC '04), Valletta, Malta. ELRA, 2010.

[18] Forcada, M. L.: Open-Source Machine Translation: An Opportunity for Minor Languages. Proceedings of the Workshop Strategies for Developing Machine Translation for Minority Languages, Conference on Language Resources and Evaluation (LREC 2006), Genoa, Italy, 2006, pp. 1–7.

[19] Forsberg, M.—Hammarström, H.—Ranta, A.: Morphological Lexicon Extraction from Raw Text Data. Advances in Natural Language Processing. Springer Berlin Heidelberg, Lecture Notes in Artificial Intelligence, Vol. 4139, 2006, pp. 488–499.

[20] Google Translate, 2012.

[21] Hajič, J.—Hric, J.—Kuboň, V.: Machine Translation of Very Close Languages. Proceedings of the 6th Applied Natural Language Processing Conference, Association for Computational Linguistics, 2000, pp. 7–12.

[22] Hajič, J.—Homola, P.—Kuboň, V.: A Simple Multilingual Machine Translation System. In: Hovy, E., Macklovitch, E. (Eds.): Proceedings of the MT Summit IX, New Orleans, USA, AMTA, 2003, pp. 157–164.

[23] Halácsy, P.—Kornai, A.—Oravecz, C.: HunPos – An Open Source Trigram Tagger. Proceedings of the ACL 2007 Demo and Poster Sessions, Association for Computational Linguistics, 2007, pp. 209–212.

[24] Homola, P.—Kuboň, V.: Improving Machine Translation Between Closely Related Romance Languages. Proceedings of EAMT, HITEC e.V, 2008, pp. 72–77.

[25] Homola, P.—Kuboň, V.: A Method of Hybrid MT for Related Languages. Control and Cybernetics, Vol. 39, 2010, No. 2, pp. 421–438.

[26] Homola, P.—Kuboň, V.—Vičič, J.: Shallow Transfer Between Slavic Languages. Recent Advances in Intelligent Information Systems, Academic Publishing House EXIT, Warsaw, 2009, pp. 219–232.

[27] Koehn, P.—Hoang, H.—Birch, A.—Callison-Burch, C.—Federico, M.—Bertoldi, N.—Cowan, B.—Shen, W.—Moran, C.—Zens, R.—Dyer, C.—Bojar, O.—Constantin, A.—Herbst, E.: Moses: Open Source Toolkit for Statistical Machine Translation. Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL '07), Association for Computational Linguistics, 2007, pp. 177–180.

[28] Kohavi, R.: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 1995, pp. 1137–1143.

[29] Labaka, G.—Stroppa, N.—Way, A.—Sarasola, K.: Comparing Rule-Based and Data-Driven Approaches to Spanish-to-Basque Machine Translation. Proceedings of the Machine Translation Summit XI, EAMT, 2007, pp. 41–48.

[30] Lavie, A.—Agarwal, A.: METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. Proceedings of Workshop on SMT at the ACL Conference, 2007.

[31] Linguistic Data Consortium: Linguistic Data Annotation Specification: Assessment of Fluency and Adequacy in Translations. Technical report, 2005.

[32] Leech, G.—Wilson, A.: EAGLES Recommendations for the Morphosyntactic Annotation of Corpora. Technical report, ILC-CNR, Pisa, 1996.

[33] Levenshtein, V.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Doklady Akademii Nauk, 1965, pp. 845–848.

[34] Ni, Y.—Niranjan, M.—Saunders, C.—Szedmak, S.: Distance Phrase Reordering for MOSES – User Manual and Code Guide. Technical report, School of Electronics and Computer Science, University of Southampton, 2010.

[35] Och, F. J.: Challenges in Machine Translation. Proceedings of the ISCSLP, Springer, 2006, p. 15.

[36] Och, F. J.—Ney, H.: A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics, Vol. 29, 2003, pp. 19–51.

[37] Oliva, K.: A Parser for Czech Implemented in Systems Q. MFF UK Prague, 1989.

[38] Oller, C. A.—Forcada, M. L.: Open-Source Machine Translation Between Small Languages: Catalan and Aranese Occitan. Strategies for Developing Machine Translation for Minority Languages (5^{th} SALTMIL Workshop on Minority Languages), 2006, pp. 51–54.

[39] Papineni, K.—Roukos, S.—Ward, T.—Zhu, W.-J.: BLEU: A Method for Automatic Evaluation of Machine Translation. Technical report, IBM, 2001.

[40] Porter, M. F.: An Algorithm for Suffix Stripping. Program Journal, Vol. 14, 1980, pp. 130–137.

[41] Sanchez-Martinez, F.—Forcada, M. L.: Inferring Shallow-Transfer Machine Translation Rules from Small Parallel Corpora. Journal of Artificial Intelligence Research, Vol. 34, 2009, pp. 605–635.

[42] Sánchez-Martínez, F.—Pérez-Ortiz, J. A.—Forcada, M. L.: Using Target-Language Information to Train Part-of-Speech Taggers for Machine Translation. Machine Translation, Vol. 22, 2008, No. 1-2, pp. 29–66.

[43] Scannell, K. P.: Machine Translation for Closely Related Language Pairs. Proceedings of the Workshop Strategies for Developing Machine Translation for Minority Languages, Genoa, Italy, 2006, pp. 103–109.

[44] Spencer, A.: Morphological Theory. Blackwell Publishing, 1991.

[45] TEI-Consortium: TEI P5: Guidelines for Electronic Text Encoding and Interchange. Technical report, TEI Consortium, 2007.

[46] Tyers, F. M.—Wiechetek, L.—Trosterud, T.: Developing Prototypes for Machine Translation Between Two Sámi Languages. Proceedings of EAMT. HITEC e.V, 2009, pp. 120–128.

[47] Villarejo, L.—Farrus, M.—Ramírez, G.—Ortíz, S.: A Web-Based Translation Service at the UOC Based on Apertium. Proceedings of the 2010 International Multiconference on Computer Science and Information Technology (IMCSIT), IEEE, 2010, pp. 525–530.

[48] Vičič, J.: Rapid Development of Data for Shallow Transfer RBMT Translation Systems for Highly Inflective Languages. Language Technologies: Proceedings of the Conference, pp. 98–103, Institut Jožef Stefan, Ljubljana, 2008, pp. 98–103.

Jernej Vičič is a research associate of the Primorska Institute for Natural Sciences and Technology in Koper, Slovenia. His main research interest is connected with the field of hybrid machine translation for related languages.

Petr Homola has finished his Ph.D. studies in the Institute of Formal and Applied Linguistics at the Faculty of Mathematics and Physics, Charles University in Prague. His research interests are mainly focused on the automatic translation of closely related languages.

Vladislav Kuboň is Assistant Professor in the Institute of Formal and Applied Linguistics at the Faculty of Mathematics and Physics, Charles University in Prague. His research aims primarily at the syntactic analysis and machine translation.