RECENT ADVANCES IN THE DEVELOPMENT AND SHARING OF LANGUAGE RESOURCES AND TOOLS FOR LATVIAN

ANDREJS VASIĻJEVS, TATIANA GORNOSTAY, INGUNA SKADIŅA, DAIGA DEKSNE, RAIVIS SKADIŅŠ AND MĀRCIS PINNIS

This chapter presents an overview of recent advances in the development and sharing of language resources and tools for Latvian[1] as one of the under-resourced languages[2]. The first section briefly describes linguistic and sociolinguistic characteristics of the Latvian language, the history of language technology for Latvian, as well as national and EU cooperation activities in Latvian language technology. The second section introduces the concept of terminology entry compounding for the identification and unification of matching multilingual entries in terminology databases from different terminological resources. The third section discusses approaches to morphological analysis and tagging for Latvian as a morphologically-rich language. The fourth section focuses on the applied grammar checking methods for the Latvian language. The fifth section reports on recent research in the combination of knowledge-based and data-driven approaches in machine translation, including factored models for statistical machine translation and application of spatial ontologies to improve the translation of toponyms. The final section provides an overview of activities in Latvia to create an infrastructure for distribution and sharing of language resources and tools.

[1] Some of the chapter sections are based on recent publications about the development of language technology for Latvian.

[2] The term an *under-resourced language* refers to the language which is not well-studied in the light of language technology and, as a result, lacks language resources and tools freely available to the community.

1 Overview of Language Technology Development for Latvian

Latvian is the sole official language in the Republic of Latvia, an official language of the European Union, and one of the oldest European languages with about 1.5 million native speakers worldwide (1.34[3] million are living in Latvia while others are scattered throughout the USA, Russia, Australia, Canada, UK, Germany, Ireland, as well as Lithuania, Estonia, Sweden, Brazil, and other countries). Latvian, though apparently small, is in fact approximately the 150th most spoken language out of about 6,900 languages of the world. At least 500,000 non-Latvians speak Latvian besides their own native language.

Since Latvian became the official language after Latvia regained independence in 1990, more and more minority language speakers have acquired Latvian language skills. According to census data, in 1989 only 23% of national minorities spoke Latvian. The next census in 2000 showed that the number of Latvian speakers among national minorities increased to 53%. However, due to low birth rates, Latvian speakers have decreased by approximately 5,000 people (0.3%) annually. Latvian is the native language of 95.6% of Latvians. Among national minorities, Latvian is considered as the native language more often by Lithuanians (42.5%), Estonians (39.2%) and Germans (24.6%). In comparison, 39.6% of Latvia’s citizens are native speakers of Russian. For a large number of other national minorities (Jews, Belarusians, Ukrainians, and Poles) the Russian language is their mother tongue and everyday communication language (META-NET White Paper Series: Latvian, 2011).

The Latvian language belongs to the Baltic branch of the Indo-European language family. The Baltic languages are divided into East Baltic and West Baltic languages. There are two living Baltic languages: Latvian and Lithuanian, which both belong to the East Baltic group. The Latvian language is a synthetically inflected language and exhibits some specific linguistic characteristics such as rich morphology due to

[3] Central Statistical Bureau of the Republic of Latvia – official data from 2009; available online at http://www.csb.gov.lv.

inflection[4], a rich set of derivational means, relatively free word order with morphological means for marking syntactic relations[5], and others.

1.1 History of Language Technology for Latvian

The history of language technology (LT) in Latvia can be traced back to the end of the 50’s. It started with the collection of various statistics about the Latvian language, creation of language resources, and development of the first machine translation (MT) system. The work on different morphological processing tools as well as the creation of different linguistic resources, for example, the Latvian frequency dictionary (Latviešu valodas biežuma vārdnīca, 1966 -1972) and the inverse dictionary of Latvian (Soida and Kļaviņa, 1970 continued during the 70’s. Different probabilistic and statistical methods were developed during that period for Latvian text analysis, including statistics of preposition usage, properties of graphemes, and characteristics of parts of speech.

Since 1988, the Institute of Mathematics and Computer Science (IMCS) of the University of Latvia, which currently is the main public research institution in LT in Latvia, has been involved in natural language processing (NLP). In 1991 Tilde, which is currently the most significant industry player, was established.

The past 20 years of LT research and development in Latvia have been very dynamic. Different linguistic resources, for example, corpora, electronic dictionaries and text collections have been developed. Although encoding systems have changed no less than three times during this period, most of the resources have been maintained to the requirements of each particular period. Many of these resources are publicly available and are included in CLARIN and META[6] catalogues.

The Latvian research community, although being rather small, has always aimed to support Latvian with modern technologies at the same level as for the so-called “large” languages. It started with the creation of the first character set for Latvian which is now a large font collection

[4] For example, Latvian, nouns have 29 graphically different endings, adjectives – 24 and verbs – 28, and only a half of the endings are unambiguous.

[5] However, “subject, predicate, complement” tends to be the most common order of sentence parts.

[6] See section 6 about CLARIN and META infrastructures.

(many have been created in Latvia), and encoding conversion utilities between different computer platforms. That work was followed by declinators and conjugators, morphological analysers, and statistical analysis tools. Since 1994, MT has been a hot topic in Latvia (both rule-based and statistical approaches were researched).

The main achievements during the past 20 years have been summarized in the seminar "Language and Technology in Europe 2000" (1994) materials, and several publications in the proceeding series of the Baltic HLT conference (International Conference "Human Language Technologies – The Baltic Perspective"). Milčonoka et al. (2004) which describes the main achievements at IMCS during 1994-2004 and Vasiļjevs et al (2004a) introduces with HLT at Tilde, while Skadiņa et al. (2010b) describes the most recent six years (2004-2010) of HLT in Latvia.

1.2 National and EU Cooperation Activities in Language Technology for Latvian

Language resources and tools have an important role in the Latvian national language policy defined in the two major documents "Guidelines of the National Language Policy for 2005-2014" and "The National Language Policy Programme for 2006-2010". Several tasks of the programme are directly related to LT:

- provide financial and administrative support to research in computational linguistics for the Latvian language;
- organize and create a modern computer-aided Latvian language database and ensure its wide usage (the result of this task should be corpora of the Latvian written and spoken language, tools for corpora management and lexicography, standards and schemas for lexical and other data);
- ensure education in computational linguistics in Latvian universities;
- ensure development of terminology in Latvian, creation of terminological databases and dictionaries, terminology harmonization and international cooperation in terminology development (Skadiņa et al., 2010b).

Most of research activities in Latvia are funded by the Latvian Council of Science (LCS). In 2005-2009, two LT related projects of IMCS were supported by LCS as a part of National Research Programs "Scientific Foundations of Information Technology" and "Latvian Studies (Letonica):

Culture, Language and History". The SemTi-Kamols project[7] aimed at the development and adaptation of the semantic web technologies for semantic analysis in Latvian. The project "Database of Latvian Explanatory Dictionaries and Recent Loanwords" was mainly concerned with semi-automatic transformation of the Dictionary of Standard Latvian Language into a machine-readable format. Work on semantic technologies continues in two large projects: "Novel information technologies based on ontologies and model transformations" of the National Research Program and "Semantic database platform for domain specialists" funded by the European Structural Funds. In addition, few smaller projects of IMCS related to LT have been funded by LCS.

Taking care of the synergy of the Latvian language and technologies, the Latvian Language in New Technologies National Language subcommission has set the following key goal – the Latvian language shall be ensured full software support in information technologies and the support shall be high-quality and maintained and developed in pace with the development of new technology, and should be widely accessible and applied. To reach these goals, the subcommission has set the following priority tasks:

- develop language computer technologies;
- ensure the availability and application of language computer technologies in widely used systems;
- develop the regulations for the use of the Latvian language in computer systems;
- promote the development and implementation of IT and telecommunication terminology (META-NET White Paper Series: Latvian, 2011).

During recent six years there has been an active period in the development of LT for Latvian in the light of the above mentioned State language policy – a number of spoken and written resources, language tools (MT systems, part of speech tagging application, named-entity recognizer, and others) were developed, as well as the cooperation between researchers from a number of academic institutions and companies on national and international levels was established.

Taking into account the importance of LT in ensuring sustainable development of LT for Latvian and other smaller languages, the Language Shore initiative was launched in 2009 under the patronage of the former President of Latvia, Valdis Zatlers. This initiative fosters the creation of a

[7] http://www.semti-kamols.lv/

partnership between the government, academia, and industry to develop an international expertise cluster around LT in Latvia. In order to ensure the successful development of the initiative at the government level, a Language Shore Steering Group composed of five line ministers was established. The initial Language Shore pilot projects have been started by Tilde and Microsoft Research increasing the speed of development of MT for Latvian, developing a new crowd-sourcing model in MT data collection, and establishing cooperation in terminology data sharing[8]. Several Language Shore related projects in MT, speech technologies, content analysis and other LT areas are planned as part of activities undertaken by the Latvian IT Competence Centre, which is being organized by leading Latvian IT companies and universities.

Several EC co-funded international collaborative projects have been initiated for the advanced research and development of MT for under-resourced languages, including Latvian. The CIP ICT PSP LetsMT![9] project, coordinated by Tilde, builds an innovative online collaborative platform for data sharing and MT generation. This cloud-based platform provides all categories of users with an opportunity to upload their proprietary resources to the repository and train tailored statistical MT systems. The latter can be shared with other users who can exploit them further on (Vasiļjevs et al., 2010). The FP7 ACCURAT[10] project, also coordinated by Tilde, researches novel methods that exploit comparable corpora to compensate for the shortage of language resources to improve MT output quality for under-resourced languages and narrow domains (Skadiņa et al., 2010a). The ACCURAT project aims to achieve strong improvement in MT output quality for a number of new EU official languages and languages of associated countries (Croatian, Estonian, Greek, Latvian, Lithuanian and Romanian), and propose novel approaches for adapting existing MT technologies to specific narrow domains, significantly increasing the language and domain coverage of MT applications.

2 Implementation of Terminology Entry Compounding

[8] www.valodukrasts.lv
[9] http://www.letsmt.eu
[10] www.accurat-project.eu

Terminology resources are among the most widely used language resources. The critical role of terminology is to ensure unambiguous and reliable communication. Insufficient distribution and reutilization of existing terminology resources has long been identified as one of the major weaknesses in the European terminology landscape (Ahmad et al., 1996). In Latvia a large effort has been made to consolidate numerous terminology resources into an online terminology bank which includes more than 145,000 terms in about 30 domains (Skadiņš and Vasiļjevs, 2004). Further integration of national terminologies into the European federated terminology bank EuroTermBank[11] will further identify and merge existing resources matching multilingual terms from different resources (Rirdance & Vasiļjevs, 2006).

Entry compounding solves the problem of unified representation of multiple potentially overlapping term entries that are present in a consolidation of a huge number of multilingual terminology sources. The majority of terminology resources in Latvia are bilingual with the source language mostly being English. A much smaller number of resources are monolingual or have terms in three or more languages.

Since multiple terms in multiple languages can refer to the same concept, the concept is the shared element that must be used to link the terms together in a multidimensional database (Wright, 2001). Henriksen et al. (2006) strongly advocate modelling the data structure according to a concept-oriented approach to terminology. Terminology entry denotes an abstract concept that has designations or terms as well as definitions in one or more languages. If a terminology bank contains entries coming from different collections and designating the same concept we have an obvious interest to merge them into one unified multilingual entry.

For example, if we have a term pair *EN computer – LV dators* coming from a Latvian IT terminology resource and another term pair *EN computer – LT kompiuteris* from a Lithuanian IT terminology resource we may want to join these two into a unified entry *EN computer – LV dators – LT kompiuteris.* Such a multilingual entry enables correspondence between language terms that are not directly available in any terminology resource (in our example new term pair *LV dators – LT kompiuteris).*

However merging entries just on the basis of a matching term in one language that is common for these entries will lead to many erroneous term correspondences. For example, if we have LV-EN entry *stumbrs – stick* and ET-EN entry *kang – stick*, we may want to merge these entries

[11] www.eurotermbank.com

into a compound entry LV-ET-EN *stumbrs – kang – stick*. But if we add to this alignment LV-EN entry *rokturis – stick* it would lead to a wrong LV-ET translation *rokturis – kang*.

Such problems obviously appear due to the frequent ambiguity of terms among subject fields or rarer cases of ambiguity in the context within one subject field. We can conclude that the only error-free method for merging entries is evaluating whether these entries denote the same concept. Unfortunately, in practice it is often impossible or very expensive to make comparisons of cross-lingual terminology concepts. There is a lack of experts with sufficient knowledge of respective languages and subject fields. The task is considerably hindered by the fact that majority of terminology collections created in Latvia do not have term definitions included.

To solve these problems we propose a new method to consolidate data representation – *terminology entry compounding*. Entry compounding is an automated approach for matching terminology entries based on available data.

The most reliable indication for matching entries is to have unique and unambiguous concept identifiers. The best example are terms from ISO terminology standards with a unique identifier for every term entry in the form *[Standard_identifier].[term_number]*. Accordingly, all national standards share the same identifier for corresponding entries and can be merged with a very high degree of reliability. Another case of unique internationally applied identification is the usage of Latin names in medicine and biology (with a number of exceptions with different Latin names designating the same concept). If there is no unique identification for concepts in collections, less precise matching criteria are used, namely, the English term and the subject field. English was chosen as the most popular language in term resources.

As the majority of Latvian terminology resources are bilingual, we would like to transform data representation from a number of separate bilingual entries to a unified multilingual record.

Entry compounding solves the problem of visual representation of multiple potentially overlapping term entries that are present in a consolidation of a huge number of multilingual terminology sources. At present, the EuroTermBank database contains over 585,711 term entries with more than 1,500,500 terms. When applying entry compounding, over 135,000 or 23% of entries are compounded. Hence entry compounding is a considerable aid for the user in finding the required term, for example, in the translation scenario between language pairs for which term equivalence is not established in existing collections.

Unfortunately, the abovementioned criteria for entry compounding are insufficient and generate too many incorrect alignments. A high recall rate also leads to a relatively low precision although we currently do not have exact precision evaluation figures. However, the majority of Latvian terminology resources do not include definitions and we need to look for other sources to depict meaning of terms.

We suggest using a multilingual text corpus as a source for term usage patterns and attempt to disambiguate its meaning. Of course it is impossible to get a term definition from a regular text corpus. But we can assume that the term meaning is related to the context where the term usually appears.

We can assume that a term *t* in a language *L1* and *s* in a language *L2* are matching (or denoting the same concept) if *t* and *s* have similar context patterns in an *L1* corpus and *L2* corpus respectively. By the context pattern we mean characteristic collocates frequently appearing in proximity of the term. Because terminology is related to the special language (special language uses specific words with specific, preferably unambiguous meaning, in contrast to the general language with a wide lexicon of usually very ambiguous words) we are interested in those collocate words that are terms from the same subject field.

In the proposed method we try to grasp the intuition that if two terms in different corpora have similar context patterns then they might denote the same concept and more frequent collocations have more impact on the term context pattern than less frequent ones (Vasiljevs and Balodis, 2010).

Let's assume that we have applied simple term compounding for bilingual terminology resources as described previously. For the language *L1* term *t* we have several translation candidates s_1, s_2, ..., s_n in the language *L2*. Our task is to select the most probable from these candidates by analysing context patterns of these terms.

Let's denote the frequency of the term *t* in the language *L1* corpus with *count(t)*. The frequency of respective translation candidates s_1, s_2, ..., s_n in the *L2* corpus will be denoted with $count(s_1)$, $count(s_2)$, ..., $count(s_n)$.

We denote collocations of the term *t* with $coll_1(t)$, $coll_2(t)$, ..., $coll_m(t)$ and the respective frequency of these collocations in proximity with *t* with $count(t, coll_1(t))$, $count(t, coll_2(t))$, ..., $count(t, coll_m(t))$.

We will select those collocations of the term *t* in the language *L1* whose frequency is higher than a certain threshold *p*.

This means that we will select $coll_j(t)$, where $\frac{count(t, coll_j(t))}{count(t)} > p$.

For every such collocation we will find translation candidate $x_1, x_2, ..., x_k$ in the language *L2*. For every candidate translation *s_i* of the term *t*:

if $\frac{count(s_i, x_1) + count(s_i, x_2) + ... + count(s_i, x_k)}{count(s_i)} > p$ then we will add to the score of this candidate the lowest from the numbers:

$$\frac{count(s_i, x_1) + count(s_i, x_2) + ... + count(s_i, x_k)}{count(s_i)}$$ and $$\frac{count(t, coll_j(t))}{count(t)}.$$

Now let's do the same calculation in the other direction – for every translation candidate s_i in the language *L2* we will select collocations whose frequency is higher than a certain threshold *p*.

This means that we will select $coll_j(s_i)$, where $\frac{count(s_i, coll_j(s_i))}{count(s_i)} > p$.

For every such collocation we will find translation candidates $x_1, x_2, ..., x_k$ in language *L1*.

If these translations appear in the context with *t* frequently enough passing our threshold *p*:

$\frac{count(t, x_1) + count(t, x_2) + ... + count(t, x_k)}{count(t)} > p$, then we will add to the score of this candidate the lowest from the numbers:

$$\frac{count(t, x_1) + count(t, x_2) + ... + count(t, x_k)}{count(t)}$$ and $$\frac{count(s_i, coll_j(s_i))}{count(s_i)}.$$

We will assume that translation candidate s_i with the highest resulting score is the most probable equivalent of the term *t* in the language *L2*.

To test the proposed method we carried out an experiment on compounding Latvian and Lithuanian terms. For this experiment we used JRC-Acquis Multilingual corpus v3.0 which is the largest publicly available source of corpus data for Latvian and Lithuanian (Steinberger et al., 2006)[12]. As we wanted to find a method for the more general case of lack of parallel in-domain data, we split this corpus in 2 parts. For the Latvian corpus we used the first part and for the Lithuanian corpus – the second one. In such a way we got a sufficiently large corpus of non-parallel texts for Latvian and Lithuanian.

For the experiment we selected 27 Lithuanian terms and 80 corresponding Latvian term candidates. Only terms with at least 50 occurrences in the corpus were selected and only Lithuanian terms for which there were at least one correct and one incorrect Latvian term were selected. The correct translation was depicted by a terminologist. Every Lithuanian term had from 2 to 8 candidate translations in Latvian from which only 1 to 4 were correct.

The size of the window for collocations was 10 words to the left and right of the term occurrences. As Latvian and Lithuanian are highly inflected languages, the morphological normalization was applied. To measure the usefulness of our method we chose the value of the threshold parameter p = 0.002.

Results of the experiment showed that our method returned the correct answer in 61% and the wrong answer in 18% of cases. In 21% of cases the difference in score was not statistically significant enough to provide an answer.

Experimental results demonstrate that the statistical context analysis in non-parallel texts can indeed improve entry compounding and is practically applicable in the representation of consolidated data. Entry compounding serves a visualization aid that displays matching entries across collections in a consolidated way. A clear indication should be provided to warn the user that this consolidation was done by an automated process and additional human validation is mandatory.

3 Development of Latvian Morphological Tools

[12] Latvian and Lithuanian corpora each contains about 27 million words.

3.1 Morphological Analyser for Latvian

Tilde has developed a morphological analysis and synthesis system for Latvian that is based on lemma lexicon and inflectional rules (Vasiljevs et al., 2004b). It can analyse correct Latvian word forms, find their basic forms, and determine the form morphological categories. The morphological synthesizer, on the other hand, can transform a given lemma into any required inflectional form.

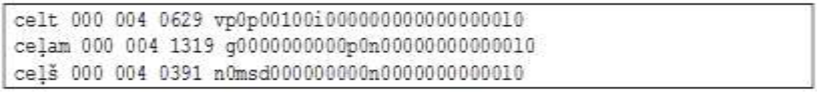
The morphological analyser features 24 different morphological and syntactic categories that describe words in Latvian: part of speech (noun, verb, punctuation, etc.), tense (present, past, future, etc.), gender (masculine, feminine), number (singular, plural), case (nominative, genitive, dative, etc.), degree of comparison (positive, comparative, superlative), person (first, second, third), adjective definiteness marker (indefinite, definite), numeral type (cardinal, ordinal), mode (indicative, imperative, subjunctive, etc.), voice (active, passive), semantic subclass of pronouns (personal, reflexive, possessive, etc.), subtype of participles (indeclinable, partly declinable), diminutive marker for nouns (diminutive, not diminutive), reflexivity of verbs (non-reflexive, reflexive), negative prefix marker (negative, affirmative), number required for agreement with prepositions (singular, plural), case required for agreement with prepositions (genitive, dative, accusative, etc.), place of preposition (preposition, postposition), verb group (1 to 9), semantic type of adverb (gradual), relation type of conjunction (coordinating, subordinating), usage of capital letters (lowercase, starts with a capital, uppercase), punctuation mark (".", "?", "!", etc.). Single part of speech types, of course, do not require all categories; therefore, an analysis of a word may contain only a limited number of categories, which are used to describe the morphological and to a limited extent also syntactic properties of the word (or any other token type). As Latvian exhibits high ambiguity of possible morphological analyses of a word, which can be explained by the language's fusional nature, with several inflections sharing the same morphemes, a single word can have multiple analyses.

The morphological analyser features a data base in which word stems with similar declension patterns are grouped together in stem groups (70133 stems in 287 stem group). As Latvian is a highly inflected language, the consonant palatalization cannot be easily described by rules (see Table X-1). Therefore, the morphology data base holds all palatalized form stems as well as basic form stems.

b →bj	c →č	d →ž	dz →dž	l →ļ	ln →ļņ	m →mj	n →ņ	kst →kš
p →pj	s →š	sl →šļ	sn →šņ	st →š	t →š	v →vj	z →ž	zl →žļ
zn →žņ	ll →ļļ	nn →ņņ	g →dz	k →c	s →t	s →d	s →z	š →t

Table X-1 Samples of consonant palatalization in Latvian

Certain prefix groups are assigned to every stem group (60 prefixes in 9 prefix groups) and ending groups (2029 form endings in 246 ending groups). All word forms also have linkage to the corresponding basic forms. The links between basic stems and palatalized stems are listed in a separate table. The form table contains full descriptions of all forms; there are properties like part of speech, gender, number, person, time, degree, case, etc. When analysing the word, the morphology component cuts the symbols from the beginning and the end of the word and tries to determine the prefix, stem and ending boundaries. After determining the stem group, to which a stem belongs, the morphology component checks whether the stem group can have the particular prefix and ending. After that, the form description and the basic form is extracted. Results are returned in text format where every analysis is on a separate line. Each line contains the lemma, word's position in a text, internal form identifier and form description (also known as the morphological tag; see Fig. X-1). The morphological tag is a string of symbols, where certain morphological category values are found at certain positions in the string. All possible categories and their corresponding values form the tagset of the morphological analyser. For example, on the 0th position is the value of the Part of speech, on the 1st position is the value of the Tense, on the 2nd position is the value of the Gender, etc. The morphology component can synthesize any form if the basic form and the full set of required form's properties is given. It can also return all word forms, which are derived from a given base form.



```
celt 000 004 0629 vp0p00100i000000000000000010
ceļam 000 004 1319 g0000000000p0n00000000000010
ceļš 000 004 0391 n0msd000000000n00000000000010
```

Fig. X-1 The result of morphological analysis of the word “ceļam”

The spelling checker for Latvian uses the same lexical data base above described for determining the correct spelling.

3.2 Maximum Entropy-Based Morphological Tagger

Morphological tagging is a process where morphological analysis and morphological disambiguation is performed for each token (words, punctuation marks and other textual fragments, for instance, numbers, codes, etc.) in a given text. As described in section **Error! Reference source not found.** morphological analysis for words is in general ambiguous. For instance, the Latvian word "*rokas*" can be a noun ("*roka*" – lit. "hand") or a verb ("*rakt*" – lit. "to dig"). As a noun it can be also in singular genitive, plural nominative, plural accusative or plural vocative forms. As a verb it can be in an imperative or indicative mode.

Many NLP tasks that do not analyse words and their structure have to cope with high morphological ambiguity. The addition of a morphological tagger, however, can solve the ambiguity issues, as long as the context contains enough information and the morphological tagger is able to understand enough of the context to remove the ambiguity.

The morphological ambiguity, the morphological richness and the relatively free order of constituents in sentences of the Latvian language would require many years of research in order to create a complete rule based morphological tagging system. A good compromise, therefore, is a combination of rule based and data driven statistical machine learning approaches, which require training data, but can be developed in a foreseeable time and achieve decent results that are comparable with rule based approaches.

The task of a statistical morphological tagger is to predict the most likely (or even the second or further best) analysis for each ambiguous token in a given context - the text to the left and to the right of the predicted token. In the case when morphological analysis is done as a data pre-processing step, the task of morphological tagging can also be viewed as a classification problem for a given token sequence (typically sentence) (Hajič and Vidová-Hladká, 1998). In such a situation the morphological tagger operates in two steps:

- Morphological analysis that assigns all possible morphological tags for each token. The Tilde system uses the morphological analyser described in section **Error! Reference source not found.**.
- Token classification, the task of which is to disambiguate each separate token and select the most likely tag using the given token sequence as context. This step is statistical, employing machine learning methods; Tilde uses exponential probabilistic tagging

models, which have been created using maximum entropy training.

The classification problem is solved (Pinnis and Goba, 2011) with a statistical morphological disambiguation system, which consists of a training model and a tagging model. The training is based on maximum entropy (Berger et al., 1996) based weight estimation for feature functions (also known as context functions) and morphological category values (for instance, "*noun*", "*verb*", etc.) in different ambiguity classes. Following the work for Czech (Hajič and Vidová-Hladká, 1998), Tilde uses the notion of ambiguity classes to describe possible morphological ambiguities within a morphological category, for instance "*noun-verb*" ambiguity in part of speech or "*nominative-genitive-accusative*" ambiguity in case.

This approach requires human annotated training data (also development and test data, if tuning and evaluation of the system is performed). The training data preparation for Latvian was effectively done in the following way: at first a corpus of sentences (selected from a balanced Latvian language corpus; about 120 000 tokens, including test data) was automatically analysed with the morphological analyser and then a human annotator selected the correct analysis of the ambiguous tokens. Such an approach to training data creation allows faster annotation as the annotator's task is limited to disambiguation of the ambiguous tokens.

Once a training corpus is created, the next step is to create a set of feature functions that describe the context around each token. Feature functions are binary (values "*0*" and "*1*") and, therefore, can either trigger or not in each position of a text. Knowledge, on which feature functions trigger in each position of a text, allows a maximum entropy based training algorithm to classify feature functions (by assigning weights), which are important for different morphological categories (part of speech, gender, number, etc.) and also different values of morphological categories (noun, verb for part of speech, etc.). Therefore, the selection of good feature functions is very important in maximum entropy based (or similar) machine learning methods. The feature functions in a morphological tagger may be:

- lexical (for instance, usage of conjunctions and prepositions);
- morphological (prefixes, suffixes, morphological categories like, gender, number, case, etc.);
- syntactic (punctuation marks, capitalization, required agreement of words in a sentence, etc.);

- other (for instance, describing word shape, occurrence elsewhere in the text, etc.).

Feature functions may be important to identify some properties of preceding or successive words. For instance, a very important feature function for Latvian has proven to be gender, number and case agreement of tokens.

Selection of feature functions can also be done automatically, for instance, two possible feature function selection approaches are:

- Iterative training of maximum entropy models, where feature functions that increase precision in the development data are iteratively added to the already selected feature functions. This approach requires good seed functions in order to achieve better results; therefore, language knowledge is required.
- Maximum mutual information (performed before training), where the feature functions that trigger unevenly within morphological categories are rated higher than feature functions that are distributed evenly. Although, the second approach assumes that all feature functions are independent, it produces high quality feature function sets (as shown by the results achieved with Latvian by (Pinnis and Goba, 2011)).

Tilde's approach to feature function selection is to differentiate feature functions for separate ambiguity class maximum entropy models, that is, the number of feature function sets is equal to the ambiguity class amount in the training data. A default feature function set is also added to account for out of vocabulary ambiguity classes.

Using human annotated training data and the selected set of feature functions maximum entropy models are trained for all ambiguity classes. During training each feature function gets a weight assigned in each ambiguity class for each possible morphological value (noun, verb, etc.), which in combination creates the maximum entropy based morphological tagging model of Latvian.

The statistical tagging model is based on an exponential probabilistic model introduced by (Hajič and Vidová-Hladká, 1998) where each subtag probability is modelled as a weighted exponential sum of feature functions that trigger in the given context. As subtag probabilities may also be evenly distributed, Tilde uses smoothing with frequency distributions of the particular subtag values of the ambiguity class with linear interpolation. Individual subtag (morphological category) probabilities are combined as the product of individual subtag probabilities, which represents the probability of a candidate morphological analysis.

The described methods applied for morphological tagging of Latvian texts achieve 91.5% accuracy. This result, however, cannot be compared to languages with simpler morphology, for instance, English, as the complexity of morphological properties in Latvian is much higher. In comparison with similar complexity inflectional languages, the results are comparable (for instance, for Czech the best current methods achieve 95-96% accuracy (Spoustová et al., 2007); Tilde also shows that the same method for Lithuanian achieves 94.35% accuracy (Pinnis and Goba, 2011)).

The morphological tagger developed by Tilde for Latvian is used as an integrated component in multiple NLP tasks, for instance, English-Latvian SMT systems that use factored models (factored models have proven to achieve better results than systems without factored models (Koehn et al., 2007a)). Also named entity recognition and terminology extraction tools for Latvian[13] use the morphological tagger in data pre-processing steps and rely on the morphological information provided by the tagger.

Although the tagger achieves decent performance, there are many possible ways how to improve the tagger in terms of accuracy, for instance:

- Out of vocabulary (OOV) words have to be handled with a guessing component. It can be a module added to morphological analysis that produces possible analyses of a word by analysing its structure. This way the guesser can be smoothly integrated within the morphological tagger. A guessing module can also be trained with a machine learning approach on training data without OOV words, thereby creating a parallel system to the existing morphological tagger. This way the results of both systems would have to be combined to form a final sentence analysis.
- Iterative feature function selection during training can improve results as not all feature functions are independent and with the feature function independence assumption some valuable information may be lost.

[13] Toolkit for multi-level alignment and information extraction from comparable corpora: the ACCURAT Project deliverable D2.6. Electronic resource:
http://www.accurat-project.eu/uploads/Deliverables/ACCURAT_D2.6_Toolkit%20for%20multi-level%20alignment%20and%20information%20extraction%20from%20comparable%20corpora-v1.0_final.pdf

- The individual subtags are not equally important, although the current method makes such assumptions. In order to remove the independence assumption, subtag weighing (and also minimum error rate training) has to be integrated within the training methods and the maximum entropy models.

4 CFG-Based Grammar Checker for Latvian

A spelling checker verifies the spelling of a single word, whereas the grammar checker works with a wider context. The aim of the grammar checker is to verify the sentence structure and punctuation and suggest corrections to the user. The grammar checker for Latvian is based on a full syntactic analysis of the text (Deksne and Skadiņš, 2011). It identifies the most common grammar mistakes, including agreement between words, punctuation and comma errors, as well as numerous stylistic errors. This approach allows the program to find long distance syntactical errors between different sub parts of the sentence. In addition, calques, slang and some other undesirable words or language constructions are identified. Tilde's Grammar checker is integrated in Microsoft Word and Open Office text editors.

4.1 System Architecture

The grammar checking system consists of separate components each having its own task. Most of them must be called in a certain order as each component relies on data structures prepared by the previous component (see Fig. X-2).

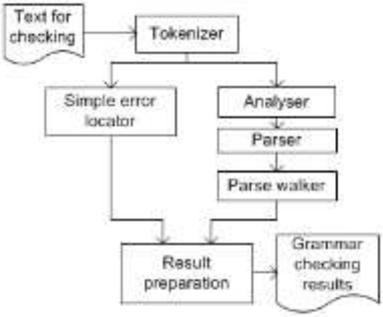


Fig. X-2 Architecture of the grammar checker

The incoming text is split into separate token objects and sentence boundaries are detected in a tokenizer module. Subsequent components work only with a sentence, not with all incoming text at once. One of the following token types is assigned to every token object: word, abbreviation, punctuation and numeric. In a simple error location module simple formatting error rules are defined using regular expressions. The analyser module adds morphological analysis to every token (see section 2.1). The parser component performs syntactic parsing using a given rule set. The rule format is based on context free grammar. The parse walker component extracts the error trees from the parse result matrix and generates suggestions for error fixing. Results from this component and from the simple error locator are passed to the result preparation module which merges results and returns to a calling application.

4.2 Grammar Checking Methods

Errors in the Latvian grammar checker are classified into 21 error types. The different grammar checking methods are applicable for different error groups. There are three different methods – error location using predefined regular expression, error location by parsing the text using capitalization pattern describing rules or rules with specific lexical parts and error location by performing a full syntactical parsing of the text using rules, which describe the correct and incorrect syntactic constructions. For faster processing simple punctuation errors are hardcoded as symbol patterns. The tokenizer while separating text also tries to locate the predefined simple error patterns. Capitalization errors are processed using the parser. A set of error rules have been developed describing correct capitalization patterns and assigning lexical parts to the rules. There are 260 rules describing correct capitalization patterns. After parsing the text using these rules, the erroneous phrase will be located and error correction suggestions generated. To locate the syntactical errors the system performs a full parsing of the sentence. A set of correct grammar rules work together with the rules describing the errors. There are 477 syntactically correct constructions describing rules and 237 error rules.

The following example illustrates the work of the parser. If we parse the Latvian text "*Manam piemēram ir jābūt skaidram. Piemēram es saprotu to.*" (Lit. "My example must be clear. For example I understand it.") (see Fig. X-3) the first sentence is fully parsed with correct grammar rules therefore we can consider it to be grammatical, the second sentence is only partially parsed with correct grammar rules therefore it is either ungrammatical or it is too complex to be fully parsed by the current set of correct grammar rules. But the parser has applied an error rule, which finds the adverb '*piemēram*' followed by a pronoun. The parser has applied a similar error rule in the first sentence too. We can ignore this error rule in the first sentence because we know that that sentence is fully parsed (grammatical). But an error rule in the second sentence really highlights a grammar error as the sentence (or phrases containing words marked by error rule) has not been fully parsed.

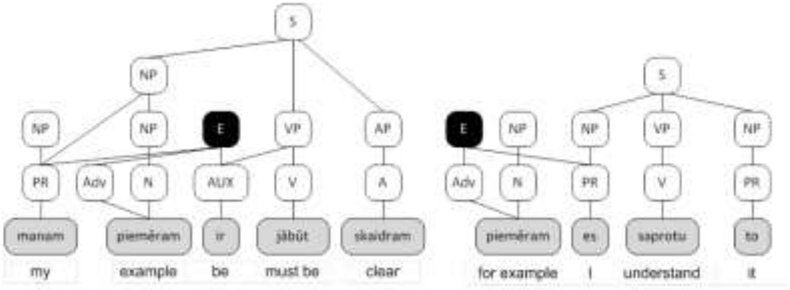
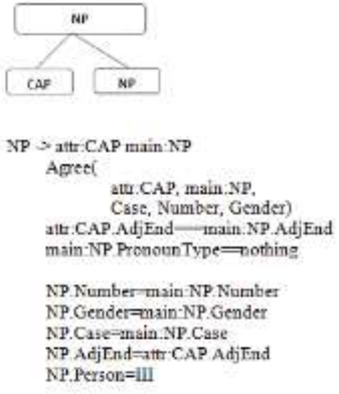


Fig. X-3 Result of parsing

4.3 Parser and Grammar Rules

There are some requirements for the parser in order to use it to find grammar errors in the way described. (i) The parser must be robust and return partial parses if the sentence cannot be fully parsed; (ii) The parser must be able to return all possible parses not only the one. As seen in the example error rules are not a part of parse trees; (iii) The parser must mark as correct only syntactic structures which really are correct; (iv) As we are working with Latvian, the parser rules must be powerful enough to deal with high morphological variance and ambiguity, word agreement and a rather free word order.

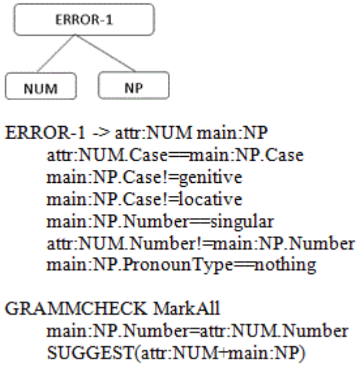
As Latvian is a morphologically rich language its grammar cannot be described with simple CFG rules like NP→N; NP→N N; S→NP V NP. The CFG used in the Latvian parser uses attributes for terminal and non-terminal symbols. For example, the noun phrase NP has attributes number, gender, case, person and some more (Fig. X-4 shows a correct grammar rule and its corresponding parse tree). The error rules operate with terminals and phrases which were created with correct grammar rules. In the rule body there are usually some agreement or disagreement statements between attributes of several, correct in themselves, phrases. There also might be an attribute comparison with an exact value. Also, lexical parts might figure in such rules. Often there is a correct grammar rule with the same right side constituents as in some error rule, only the comparison operators are different. The error rules have a section where the correct attribute values are assigned and instructions for suggestion generation are given (Fig. X-5 shows an error rule and its corresponding parse tree).



```
NP -> attr:CAP main:NP
      Agree(
               attr:CAP, main:NP,
               Case, Number, Gender)
      attr:CAP.AdjEnd==main:NP.AdjEnd
      main:NP.PronounType==nothing

      NP.Number=main:NP.Number
      NP.Gender=main:NP.Gender
      NP.Case=main:NP.Case
      NP.AdjEnd=attr:CAP.AdjEnd
      NP.Person=III
```

Fig. X-4 Sample of correct



```
ERROR-1 -> attr:NUM main:NP
      attr:NUM.Case==main:NP.Case
      main:NP.Case!=genitive
      main:NP.Case!=locative
      main:NP.Number==singular
      attr:NUM.Number!=main:NP.Number
      main:NP.PronounType==nothing

GRAMMCHECK MarkAll
      main:NP.Number=attr:NUM.Number
      SUGGEST(attr:NUM+main:NP)
```

Fig. X-5 Sample of error rule

grammar rule

4.4 Results and Evaluation

The grammar checking system described above has been evaluated on development and test corpora which contain about an equal amount of texts of the following types: high school student essays, university student papers, blogs, e-mails, non-edited marketing texts, non-edited written texts from non-native Latvian speakers with good Latvian language knowledge, news texts, draft of some project tender, the works of amateur writers, texts from specialists in certain fields (physics teachers, programmers, doctors, lawyers, etc.). Recall, precision, f-measure, confidence interval for the precision are calculated for every error type. The value of recall shows the possibility of finding all existing errors in the text. The value of precision shows the possibility of correctly finding errors in the text. Recall is given only for the development corpus, as the test corpus was not previously marked.

The recall and precision values might be influenced by the fact that a sentence can contain several errors. A human evaluator marks sentences with only a single error type. The grammar checking system also selects a single error per sentence – the one which covers the largest phrase. The error types of the human evaluator and the grammar checking system might not match.

In the Table X-2 are shown the measure values for the eight most common error types.

Error type	**Recall for development corpus**	**Precision for development corpus**	**Precision for test corpus**
Agreement between words	0.247	0.543	0.426
Punctuation error at the end of sentence	0.240	0.957	—
Words must be written together	0.761	0.962	1.000
Comma error in insertions	0.563	0.913	0.892
Comma error in participial phrase	0.427	0.704	0.660
Wrong writing style	0.397	1.0	0.950

Error type	Recall for development corpus	Precision for development corpus	Precision for test corpus
Comma error in equal parts of sentence	0.140	0.773	0.583
Comma error in sub clause	0.329	0.773	0.758
All error types	**0.290**	**0.833**	**0.710**

Table X-2 Evaluation results

The evaluation showed that users prefer a grammar checker with a high precision rather than a high recall. Some important error types, for example, agreement errors, do not perform very well. In the future work will be continued on such error types. Also more errors describing the style errors will be added to the Grammar Checker as at the moment such errors are not covered very well.

5 Combining Knowledge-Based and Data-Driven Approaches in Machine Translation for Latvian

In recent years, several machine translation systems have been built for Latvian. In addition to Google and Microsoft machine translation engines and research experiments with statistical machine translation for Latvian (Skadiņa and Brālītis, 2009) there is an English-Latvian rule-based machine translation system available (Skadiņš et al., 2007). Numerous inflectional word forms, relatively free word order and limited availability of parallel corpora pose a sparseness problem for statistical machine translation. In this section we describe the application of knowledge-based techniques to improve the quality of statistical machine translation.

5.1 Improving Latvian Statistical Machine Translation with Factored Models

For training the SMT systems, both monolingual and bilingual sentence-aligned parallel corpora of a substantial size are required. The corpus size largely determines the quality of translation, as has been shown both in case of multilingual SMT (Koehn et al., 2003) and English-Latvian SMT (Skadiņa and Brālītis, 2009). For the factor-based SMT system for Latvian developed by Tilde publicly available DGT-TM and OPUS corpora were used, as well as Tilde's proprietary corpus. Word and phrase translation from bilingual dictionaries were additionally included to increase the word coverage. The total size of the English-Latvian parallel data was 3,23 million sentence pairs. Monolingual corpora were prepared from the corresponding monolingual part of parallel corpora, as well as news articles from the web for Latvian and European Parliament Proceedings and News Commentary[14] for English. The total size of the Latvian monolingual corpus is 319 million words and 521 million words for English.

The Moses SMT toolkit (Koehn et al., 2007b) for SMT training and decoding was used for the baseline SMT system. The inflectional variation increases data sparseness at the boundaries of translated phrases, where a language model over surface forms might be inadequate to estimate the probability of the target sentence reliably. The baseline SMT system was particularly weak at an adjective-noun and subject-object agreement. To address that, we introduced an additional language model over morphologic tags in the English-Latvian system. For this, the tokens in the parallel corpus have been tagged with the Latvian morphologic tagger[15]. The tags contain relevant morphologic properties (case, number, gender, etc.) that are generated by a morphological tagger. The order of the tag LM has been increased to 7, as the tag data has had significantly smaller vocabulary.

When translating from a morphologically rich language, the SMT baseline system will not give the translation for all forms of a word that is not fully represented in the training data. The solution addressing this problem would be to separate the richness of morphology from the words and translate lemmas instead. Morphological tags can be used as additional factors to improve the quality of translation.

As a result, a human evaluation showed a clear preference for the factored SMT over the baseline system, which operated only with surface forms. Automated measurement of the translation quality using BLEU

[14] The monolingual training data from the Fourth Workshop on Statistical Machine Translation (http://www.statmt.org/wmt09/translation-task.html).

[15] For the Latvian language Tilde morphological tagger was used, for English – Connexor parser (http://www.connexor.eu/technology/machinese/machinesesyntax/).

score showed an improvement from 24.8% for the baseline system to 25.6% for the system factored with morphological tags. Human evaluation further confirmed this by users preferring output of the factored system in 58.67% of cases (for more details see Skadiņš et al., 2010).

5.2 Application of Spatial Ontologies to Improve Statistical Machine Translation

Modern SMT methods use different kinds of additional morphological and syntactical knowledge to build more sophisticated statistical models and improve the output quality of translation. Innovative research has resulted in the English-Latvian statistical factored phrase-based MT system with spatial knowledge[16]. The system is based on the Moses toolkit and enriched with semantic knowledge inferred from the spatial ontology.

Using semantic knowledge in rule-based MT is not new in the field. In SMT, however, there has been little research in this area[17]. The English-Latvian SMT system implemented uses semantic knowledge to improve the quality of MT, in particular with regard to the disambiguation of geographical names, or toponyms.

Toponyms are geographical names, or names of places (hydronyms, oronyms, geonyms, oeconyms, etc.). A natural language is ambiguous and toponyms are not exceptions. This fact makes toponyms difficult for processing (for example, resolution, cross-language information retrieval, human translation and especially MT), and due to their linguistic and extralinguistic nature toponyms require special treatment (Gornostay and Skadiņa, 2009). There are cases when real-world geographical knowledge is required for the resolution of ambiguous toponyms. The implemented SMT system deals with two types of ambiguity (see Leidner (2007) for the description of possible types of toponym ambiguity). The first type is a referential ambiguity, where a toponym may refer to more than one location of the same type, for example, *Georgia* as the US state and as the country in Caucasus (English) or *Riga* as the populated place and as the

[16] The English-Lithuanian SMT system with spatial knowledge is described in Skadiņš et al. (2011).

[17] See, for example, the research on extracting phrasal correspondences that are approximately semantically equivalent for building a full-sentence paraphrasing model that then is applied to a single good reference translation for each sentence in an SMT development set in Madnani et al. (2008).

capital of Latvia and as the populated place in the USA, state Michigan (Latvian). The second type of ambiguity is a feature type ambiguity, where a toponym may refer to more than one place of a different type, for example, *Tanfield* refers to the populated place as well as the castle in the UK (English) and *Gauja* refers to the populated place as well as the river in Latvia. The two types of toponym ambiguity are resolved in the system using semantic knowledge inferred from the spatial ontology.

The spatial ontology was developed using the ontology language, designed and implemented in the web ontology language (OWL) using RCC-8 properties (Region Connection Calculus) (Randell et al., 1992), tools developed in the SOLIM project[18], and the GeoNames database[19] (more than 15,000 toponyms), and integrated into the MT process. Spatial knowledge is added to toponyms in the source text as additional semantic tags, or factors, providing additional information for the Moses decoder. By adding factors into the source text, the translation accuracy is improved. This is the result of resolving semantic ambiguities in the source language.

The described MT approach was evaluated against the baseline system without the spatial knowledge. A multifaceted evaluation strategy including automatic (black-box) evaluation, human evaluation, and linguistic analysis, was implemented to perform evaluation experiments that showed that the quality of MT can be improved by using the semantic information from the spatial ontology. It was noticed during the linguistic evaluation that some RCC-8 properties seem to be much more useful than others (for example, *EC:externally connected* and *EQ:equal*), but a detailed evaluation of the impact of each relation has not been done yet. The EQ property can be used for machine translation of toponyms which are synonyms, for example, a full name and an abbreviation (the *United States of America* and *USA*). Moreover, the spatial ontology was not used for the disambiguation of common nouns since they were not represented in the ontology. However, a morpho-syntactic type of toponym ambiguity (when a word itself can be a toponym or a common noun in a language) and its resolution can be performed with the help of the spatial ontology, for example, *Hook* refers to the populated place in the UK and *hook* is a common noun (English) and *Liepa* refers to the populated place in Latvia and *liepa* (lime-tree) is a common noun (Latvian).

The proposed approach to toponym disambiguation is not limited to:

[18] www.solim.eu
[19] www.geonames.org

- MT per se and can be regarded as generic, i.e. it can be also applied to other fields of NLP, for example, information retrieval;
- use of spatial knowledge solely: other types of implicit or inferred knowledge can be used in a similar way.

6 Towards Distribution Infrastructure of Language Resources and Tools for Latvian

In the previous sections we demonstrated the current status of language resources and tools for the Latvian language. However, their availability, sustainability and interoperability are important issues which recently have been discussed not only in the framework of the Latvian language, but at the European level and around the world. Recently Latvian has been included in two European open linguistic infrastructure initiatives to work towards consolidating the fragmentation of language resources and tools – CLARIN[20] and META-NET[21], and this section overviews the presence of the Latvian language in these infrastructures.

6.1 Latvian in the CLARIN Infrastructure

CLARIN (Common Language Resources and Technology Infrastructure) is the pan-European initiative aimed at the creation of an integrated, interoperable, stable, persistent, accessible and extendable research infrastructure of language resources and tools for the whole European Humanities and Social Sciences community (Váradi et al., 2008). The work on CLARIN infrastructure was planned into three phases: preparatory (2008-2011), construction (2011-2016) and exploitation (from 2016). Recently CLARIN preparatory phase has been finished and further implementation of the infrastructure is planned through CLARIN ERIC (European Research Infrastructure Consortium).

Latvia has been an active member of the CLARIN initiative since its beginning in 2006. This initiative is supported by the Latvian government, i.e., the participation in the CLARIN preparatory phase project was funded by the Ministry of Education and Science of the Republic of Latvia. The

[20] www.clarin.eu
[21] www.meta-net.eu

advancement of CLARIN is mentioned in strategic documents on the development of science in Latvia. Recently the Cabinet of Ministers has approved "Action Plan for Implementation of Guidelines for Science and Technology Development". Similarly to CLARIN EU, in Latvia the work on CLARIN aims was organized around four dimensions: technology, users, language and legal (Skadiņa, 2009).

The central element of the CLARIN infrastructure is language resources and tools, their availability and usability. CLARIN started with the creation of Language Resource and Tools inventory in member countries and collected information about 890 language resources and 231 tools. Although the Latvian language belongs to under-resourced languages, CLARIN's language resource inventory contains information about 11 tools and 34 language resources for Latvian.

Through the questionnaire we identified different categories of language resources. However, most of the resources could be classified in the following three categories:

- electronic dictionaries (9);
- text collections and corpora (9);
- folklore related materials (8).

These language resources have been created in different research centres, universities and companies in Latvia, such as Daugavpils University, Institute of Mathematics and Computer Science (University of Latvia) (Skadiņa et al, 2010b), Latvian Language institute (University of Latvia), Latvian Academy of Sciences, Liepāja University, Rēzekne Higher Education Institution, State Language Centre, Tilde and University of Latvia and its institutes.

The National Library of Latvia is among active language resource providers. Since 2006, the Library has worked on the creation of *Latvian National Digital Library "Letonica"*[22]. Currently the Digital Library holds collections of newspapers, pictures, maps, books, sheet music and audio recordings. Collection *Periodicals*[23] offers 40 newspapers and magazines in Latvian, German, and Russian from 1895 to 1957 (more than 350,000 pages).

Many of these language resources are publicly available from the websites of their owners. However, not all the identified resources are in machine readable form: there are still resources in text or document format

[22] www.lnb.lv/en/digital-library

[23] periodika.lv

for which initial processing is necessary to make them available for a larger community of researchers.

In contrary to language resources, Latvian language processing tools have for the most part been developed by the Institute of Mathematics and Computer Science (University of Latvia) and by Tilde. These tools include basic language resources, such as sentence breakers, tokinezers, morphological analysers, taggers and chunkers. Some more advanced technologies, such as machine translation tools are also provided by these institutions.

The availability of Latvian language resources and tools is included in the CLARIN BLARK[24] (Basic Language Resource Kit) matrix for humanities. From the matrix we can conclude that language resources and tools for Latvian have been developed on the same level as for other under-resourced languages. However there is a difference with "large" languages, e.g. English, German or French. One of the ways how the gaps in BLARK matrix could be filled is to establish a targeted national program for HLT research and development.

Another important building block of the CLARIN infrastructure is technologies allowing access, storage, maintenance and use of CLARIN infrastructure. During the preparatory phase much work has been done on defining CLARIN centres (and establishment some of them), setting up authorization/authentification mechanisms, working on common standards for metadata and services and implementing user oriented workflows. Several Latvian language processing tools were made available through experimental web services[25]. Some tools, i.e., a Latvian tokenizer, sentence splitter, POS tagger and a morphological analyser were standardized according to the ISO family of standards. The main annotation formats used are *Morpho-syntactic annotation framework*[26] and *Lexical markup framework*[27]. For specifying morphological categories ISOcat was used.

Now, when the preparatory phase is finished the legal framework for further CLARIN implementation through CLARIN ERIC is prepared and initial request is made to the EC. For this, fourteen countries, including Latvia, have signed the Memorandum of Understanding on the construction of CLARIN-ERIC.

[24] http://www-sk.let.uu.nl/u/D5C-4.pdf

[25] http://valoda.ailab.lv/ws/

[26] MAF; ISO/DIS 24611

[27] LMF; ISO/IS 24613:2008

In Latvia IMCS is appointed as CLARIN National contact point by Ministry of Education and Research. National workshops are organized to inform the CLARIN network not only about CLARIN activities and progress, but also to inform about related national and international activities. The CLARIN National Advisory Board, established and approved by the Ministry of Education of Science, is the main instrument to coordinate creation of the CLARIN infrastructure in Latvia. The Advisory Board consists of 17 members from universities, research institutes, government organizations and enterprises. Tasks of the Advisory Board include setting priorities and providing recommendations related to the goals of the CLARIN project. It is the central body to coordinate the work of the CLARIN network in Latvia.

6.2 Latvian in the META-NET network

In the last decade language resources have grown rapidly for all EU languages, including under-resourced languages. However they are located in different places, have developed in different standards (if any) and in many cases are not well documented. High fragmentation and a lack of unified access to language resources are among key factors that hinder European innovation potential in language technology development and research.

To address these issues European Commission (EC) has dedicated specific activities in its FP7 R&D and ICT-PSP programmes[28]. The overall objective is to ease and speed up the provision of online services centred around computer-based translation and cross-lingual information access and delivery. The focus is on assembling, linking across languages, and making the basic language resources (models, tools and datasets) used by developers, professionals and researchers to build specific products and applications widely available.

Several projects have been selected and started to facilitate the creation of a comprehensive infrastructure enabling and supporting large-scale multi- and cross-lingual services and applications. South East and Central part of the META-NET is covered by the CESAR project coordinated by Hungarian Academy of Sciences. United Kingdom and Southern European countries are represented by the METANET4U project coordinated by

[28] http://ec.europa.eu/information_society/activities/ict_psp/documents/ict_psp_wp2010_final.pdf

University of Lisbon. The META-NORD project (Skadiņa et al, 2011) aims to establish an open linguistic infrastructure in the Baltic and Nordic countries to serve the needs of the industry and research communities in the development language technology.

These projects closely cooperate and form a common META-NET network with 44 members, representing 31 European countries, including Latvia. At the core of the META-NET is T4ME project which is funded under FP7 programme and coordinated by DFKI.

Latvian is among 8 focus languages (Estonian, Finnish, Icelandic, Latvian, Lithuanian, Norwegian, and Swedish) of the META-NORD project[29]. Recently the META-NORD consortium has prepared a series of reports on the language service and language technology industry in their countries. Reports for each language are compiled on the basis of the same framework that is used in the whole META-NET network.[30] They present information on general facts of the language (number of speakers, official status, dialects, etc.), particularities of the language, recent developments in the language and language technology support, core application areas of language and speech technology, and the situation in the language with respect to these areas. The Latvian language is described in META-NET White Paper Series Languages in the European Information Society: Latvia (2011).

These reports also present detailed tables with ratings of language technology resources and tools for each language. Experts from each country were asked to rate existing tools and resources with respect to seven criteria: quantity, availability, quality, coverage, maturity, sustainability, and adaptability. For each criterion a score between 0 and 6 was assigned. Later some calibration of the scores was performed by the consortium partners responsible for a particular language group. Fig. X-6 provides average scores for language resources and tools for all META-NORD languages. If we compare languages of the Baltic countries, these scores are higher for Estonian and lower for Lithuanian.

[29] http://www.meta-nord.eu/

[30] http://www.meta-net.eu/whitepapers

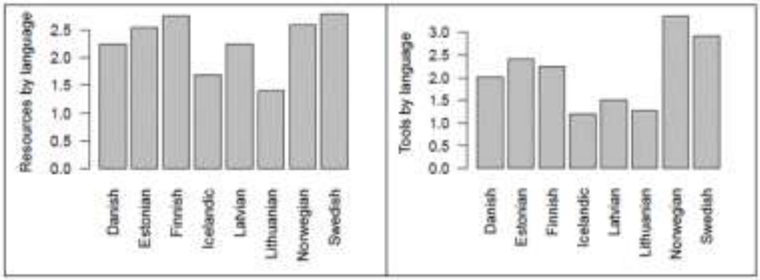


Fig. X-6 Average scores
for META-NORD language resources and tools

Fig. X-7 provides an overview of language resources for META-NORD languages. Several language resources, e.g. terminological resources and reference corpora are quite well developed for Latvian. However, more advanced language resources, e.g. multimodal data, discourse corpora and semantic corpora are missing and perhaps will not be available in the near future.

In Fig. X-7 the availability scores of tools for different META-NORD languages are provided. In general, basic LT tools are quite well presented for all languages while most of the advanced technologies are available only for few languages. For the Latvian language most of basic language technologies, e.g. tokenization and morphology processing tools, has good availability, while from most advanced tools only machine translation and speech synthesis tools are available. The development of tools for speech recognition, question answering, information retrieval and others requires a lot of human efforts and thus these tools have not been developed yet or are in very early prototype stage (e.g. semantic processing tools).

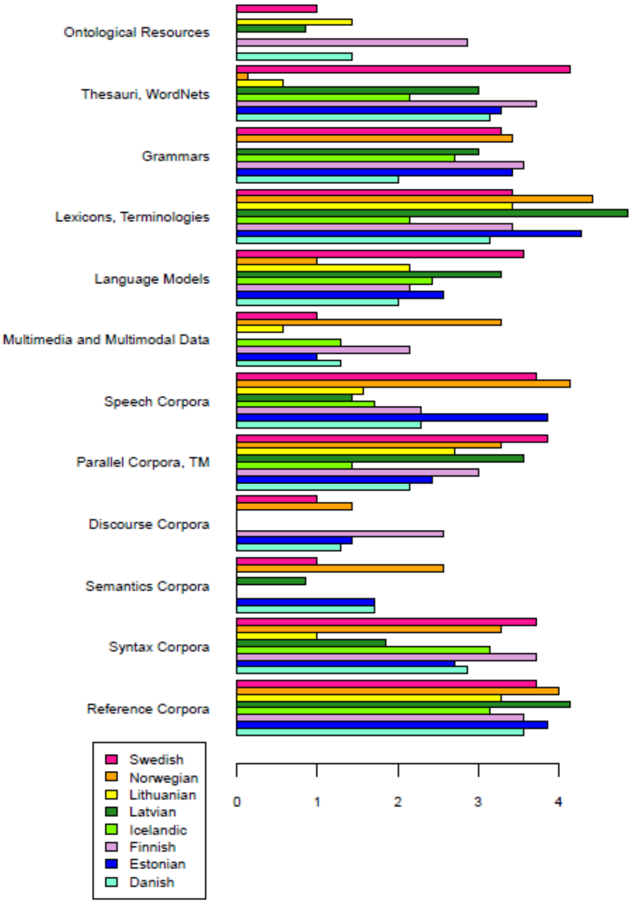


Fig. X-7 Evaluation results for resources

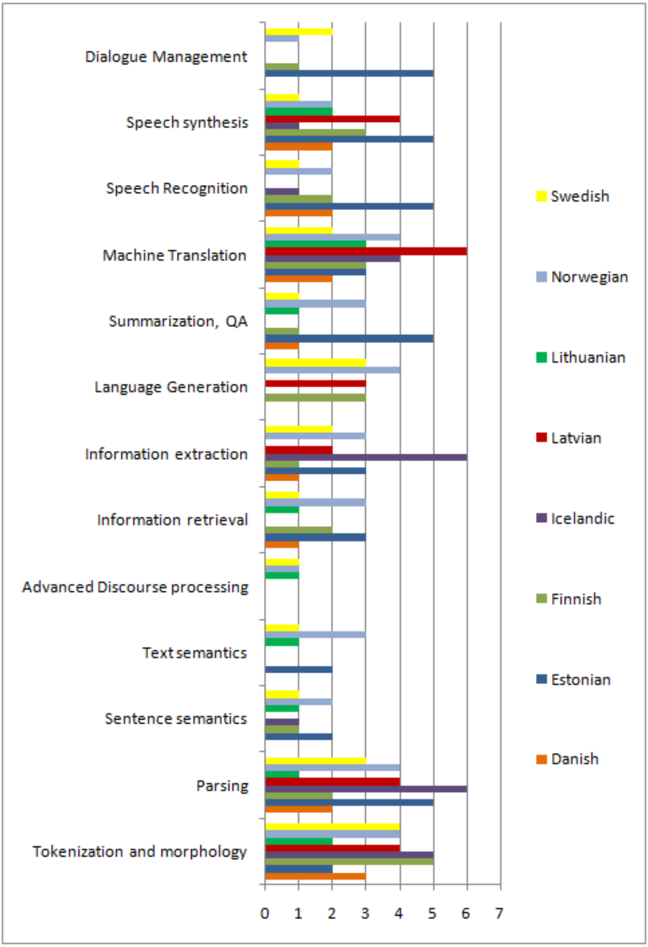


Fig. X-8 Availability of tools for META-NORD languages

The results presented in Fig. X-7 and Fig. X-8 indicate that only with respect to the most basic tools and resources such as tokenizers, PoS taggers, morphological analysers/generators, syntactic parsers, reference corpora, and lexicons/terminologies, the status is reasonably positive not only for Latvian but for all the META-NORD languages. Furthermore, all the languages seem to have some tools for information extraction, machine translation and speech recognition and synthesis, as well as resources such as parallel corpora, speech corpora, and grammar, although these tools and resources are rather simple and have limited functionality for some of the languages. When it comes to more advanced fields such as sentence and text semantics, information retrieval, language generation, and multimodal data, it appears that one or more of the languages lack tools and resources for these fields.

7 Conclusions

Although there are only several active research and development institutions in the language technology field in Latvia, strong progress has been achieved in the creation of basic language resources and tools as well as advanced applications such as machine translation.

In this chapter we described several examples of recent research results for Latvian language – entry compounding to facilitate consolidation of bilingual terminology resources in multilingual term base, maximum entropy based approach for development of morphological tagger, enriching statistical machine translation models with knowledge-based components. Current development of these techniques has reached the level where they can be implemented in practical applications addressing the needs of large user groups in a variety of application scenarios.

We presented implementation and evaluation of these methods for Latvian but they can be applied for other under-resourced languages as well.

We also demonstrate the importance and potential of making language resources and tools widely accessible and usable through establishing rich online infrastructures for resource distribution and addressing spectrum of issues including technical, legal and need to involve and educate users.

8 Acknowledgement

We would like to acknowledge the contribution of several EU co-funded projects in achieving some of the results presented in this chapter:

- the EuroTermBank project co-funded by the EU eContent programme;
- the LetsMT! project co-funded by the ICT Policy Support Programme (ICT PSP Theme 5 – Multilingual web) under grant agreement no^{o}250456;

- the META-NORD project (Baltic and Nordic Parts of the European Open Linguistic Infrastructure) co-funded by the Competitiveness and Innovation Framework Programme (CIP-Pilot Actions) under grant agreement no 270899;
- the ACCURAT project (Analysis and Evaluation of Comparable Corpora for Under-Resourced Areas of Machine Translation) co-funded by the EC Seventh Framework Programme (Theme ICT-2009.2.2 – Language-based interaction) under grant agreement no 248347;
- the EC Seventh Framework Programme project CLARIN (Common Language Resources and technology Infrastructure) under grant agreement no 212230 and the CLARIN project in Latvia supported by the Ministry of Education and Science of the Republic of Latvia.

9 Bibliography

Ahmad, K. et al. 1996, “POINTER Final Report”, http://www.computing.surrey.ac.uk/ai/pointer/report/index.html

Berger, A., S. Della Pietra and V. Della Pietra. 1996, “A Maximum Entropy Approach to Natural Language Processing”, *Computational Linguistics (22)1*.

Deksne, D. and R. Skadiņš. 2011, “CFG Based Grammar Checker for Latvian”, *Proceedings of the 18th Nordic Conference of Computational Linguistics NODALIDA 2011, Riga, Latvia*.

Gornostay, T. and I. Skadiņa. 2009, “English-Latvian Toponym Processing: Translation Strategies and Linguistic Patterns”, *Proceedings of EAMT-2009 the 13th Annual Conference of the European Association for Machine Translation, Spain*.

Hajič, J. and B. Vidová-Hladká. 1998, “Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset”, *Proceedings of the COLING-ACL Conference, Canada*.

Henriksen, L., C. Povlsen and A. Vasiljevs. 2006, “EuroTermBank – a Terminology Resource based on Best Practice”, *Proceedings of the Fifth International Conference on Language Resources and Evaluation: LREC’06.*

Koehn, P., J. F. Och and D. Marcu. 2003, “Statistical Phrase-Based Translation”, *Proceedings of HLT/NAACL*.

Koehn, P. and H. Hoang. 2007a, “Factored translation models”, *Proceedings of EMNLP-CoNLL*.

Koehn, P., M. Federico, B. Cowan, R. Zens, C. Duer, O. Bojar, A. Constantin and E. Herbst. 2007b, “Moses: Open Source Toolkit for Statistical Machine Translation”, *Proceedings of the ACL 2007 Demo and Poster Sessions, Prague*.

Language & Technology in Europe 2000: Reports of Seminar. 1994, Riga, November 10-11.

Latviešu valodas biežuma vārdnīca. Rīga: Zinātne, Vol. 1 – 1966, Vol. 2 – 1969, Vol 3 – 1972.

Leidner, J. 2007, “Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding of Place Names”. *PhD thesis. Institute for Communicating and Collaborative Systems School of Informatics, University of Edinburgh*.

Madnani, N., P. Resnik, B. Dorr and R. Schwartz. 2008, “Applying Automatically Generated Semantic Knowledge: A Case Study in Machine Translation”. *Proceedings of the Symposium on Semantic Knowledge Discovery, Organization and Use*.

META-NET White Paper Series: Languages in the European Information Society – Latvia. 2011, http://www.meta-nord.eu/uploads/Deliverables/D2.1%20Language%20Report%20for%20each%20language%20covered%20in%20the%20project_Latvia.pdf (early release edition).

Milčonoka, E., N. Grūzītis and A. Spektors. 2004, “Natural Language Processing at the Institute of Mathematics and Computer Science: 10 Years Later”, *Proceedings of the first Baltic conference „Human Language Technologies – the Baltic Perspective*.

Pinnis, M. and K. Goba. 2011, “Maximum Entropy Model for Disambiguation of Rich Morphological Tags”, *Proceedings of the Second Workshop on Systems and Frameworks for Computational Morphology, Communications in Computer and Information Science, Vol. 100*.

Randell, D. A., Z. Cui and A. G. Cohn. 1992, “A spatial logic based on regions and connection”, *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning, Morgan Kaufmann, San Mateo*.

Rirdance, S. and A. Vasiljevs (eds.). 2006, *Towards Consolidation of European Terminology Resources. Experience and Recommendations from EuroTermBank Project*. Riga: EurotermBank Consortium.

Skadiņa, I. and E. Brālītis. 2009, “English-Latvian SMT: knowledge or data”, *Proceedings of the 17th Nordic Conference on Computational Linguistics NODALIDA, Odense, Denmark, NEALT Proceedings Series, Vol. 4*.

Skadiņa, I. 2009, “CLARIN in Latvia: current situation and future perspectives”, *Proceedings of the NODALIDA 2009 workshop Nordic Perspectives on the CLARIN Infrastructure of Common Language Resources, Odense, Denmark, NEALT Proceedings Series, Vol. 5*.

Skadiņa, I., A. Vasiļjevs, R. Skadiņš, R. Gaizauskas, D. Tufis and T. Gornostay. 2010a, “Analysis and Evaluation of Comparable Corpora for Under Resourced Areas of Machine Translation”, *Proceedings of the 3rd Workshop on Building and Using Comparable Corpora. European Language Resources Association (ELRA), La Valletta, Malta*.

Skadiņa, I., I. Auziņa, N. Grūzītis, K. Levāne-Petrova, G. Nešpore, R. Skadiņš and A. Vasiļjevs. 2010b, “Language Resources and Technology for the Humanities in Latvia (2004–2010)”, *Proceedings of the Fourth International Conference Baltic HLT*.

Skadiņa, I., A. Vasiļjevs, L. Borin, De K. Smedt, K. Lindén and E. Rögnvaldsson. 2011, “META-NORD: Towards Sharing of Language Resources in Nordic and Baltic Countries”, *Proceedings of the Workshop on Language Resources, Technology and Services in the Sharing Paradigm, Chiang Mai*.

Skadiņš, R. and A. Vasiļjevs. 2004, “Multilingual Terminology Portal – termini.letonika.lv”, *Proceedings of the First Baltic Conference “Human Language Technologies – the Baltic Perspective”, Riga*.

Skadiņš, R., I. Skadiņa, D. Deksne and T. Gornostay. 2007, “English/Russian-Latvian Machine Translation System”, *Proceedings of HLT 2007, Kaunas, Lithuania*.

Skadiņš, R., K. Goba and V. Šics. 2010, “Improving SMT for Baltic languages with factored models”, *Proceedings of the Fourth International conference “Human Language Technologies – the Baltic Perspective”*.

Skadiņš, R., T. Gornostay and V. Šics. 2011, “Toponym Disambiguation in an English-Lithuanian SMT System with Spatial Knowledge”, *Proceedings of the 18th Nordic Conference of Computational Linguistics NODALIDA 2011, Riga, Latvia, the NEALT Proceedings Series, Vol. 11*.

Soida, E. and S. Kļaviņa. 2007, *Latviešu valodas inversā vārdnīca*, Rīga: LVU.

Spoustová, D., J. Hajič, P. Krbec, P. Květoň and J. Votrubec (Raab). 2007, “The Best of Two Worlds: Cooperation of Statistical and Rule-Based Taggers for Czech”, *Proceedings of Balto-Slavonic Natural Language Processing*.

Steinberger, R., B. Pouliquen, A. Widiger, C. Ignat, T. Erjavec, D. Tufiş and D. Varga. 2006, "The JRCAcquis: A multilingual aligned parallel corpus with 20+ languages", *Proceedings of the Fifth International Conference on Language Resources and Evaluation: LREC'06*.

Váradi, T., S. Krauwer, P. Wittenburg, M. Wynn and K. Koskenniemi. 2008, "CLARIN: common language resources and technology infrastructure", *Proceedings of the Sixth International Conference on Language Resources and Evaluation: LREC'08*.

Vasiļjevs, A., J. Ķikāne and R. Skadiņš. 2004a, "Development of HLT for Baltic languages in widely used applications", *Proceedings of the First Baltic Conference "Human Language Technologies – the Baltic Perspective", Riga*.

Vasiljevs, A., I. Skadina, D. Deksne and R. Skadins. 2004, "Human Language Technologies for Baltic Languages – Developments and Perspectives", *Proceedings of the Workshop on Proofing Tools and Language Technologies, Patras, Greece*.

Vasiļjevs, A., T. Gornostay and R. Skadins. 2010, "LetsMT! – Online Platform for Sharing Training Data and Building User Tailored Machine Translation", *Proceedings of the Fourth Baltic conference "Human Language Technologies – the Baltic Perspective"*.

Vasiļjevs, A. and K. Balodis. 2010, "Corpus based analysis for multilingual terminology entry compounding", *Proceedings of the Seventh International Conference on Language Resources and Evaluation: LREC'10*.

Wright, S. E. and G. Budin. 2011, *Handbook of Terminology Management. Vol. 2: Application-Oriented Terminology Management,* Amsterdam and Philadelphia: John Benjamins Publishing Company.