First International Workshop on Computational Linguistics for Uralic Languages

Proceedings of the Workshop

January 16th, 2015
Tromsø, Norway
Preface

The Uralic languages are an interesting group of languages from computational-linguistic perspective. They share large parts of morphological and morphophonological complexity that is not present in the Indo-European group which has traditionally dominated computational-linguistic research. This can be seen for example in number of word forms per word, which in Indo-European languages is in range of ones or tens whereas for Uralic languages it is in range of hundreds and thousands. Furthermore, Uralic languages share a lot of geo-political aspects: the national languages of the group—Finnish, Estonian and Hungarian—are small languages and only moderately resourced in terms of computational-linguistic resources while being stable and not in threat of extinction, the recognised minority languages of western-European states—such as North Sámi and Võro—are clearly in category of lesser resourced and more threatened, whereas the majority of Uralic languages in the east of Europe and Russia are close to extinction. Common to all rapid development of more advanced computational-linguistic methods is required for continued vitality of the languages in everyday life, to enable archiving and use of the languages with computers and other devices such as mobile applications.

The research of computational linguistics and Uralistics is being carried out in a handful of universities, research institutes and other sites by relatively few researchers. Our intention with organising this conference is to gather these researchers together in order to share ideas and resources, and avoid duplicating efforts in gathering and enriching these scarce resources, and hopefully to found an ongoing tradition of concentrated effort in collecting and improving language resources and technologies for the survival of the Uralic languages.

For the conference we received 14 high-quality submissions about topics including computational lexicography, language documentation, optical character recognition, web-as-corpus and automatic and rule-based morphological analysis methods. These are all very important for preservation and development of Uralic languages. We also received a broad coverage of languages in the submissions: North Sámi, Khanty, Mansi, Udmurt, Erzya, Moksha, Finnish and Estonian.

The conference was held at UiT Norgga árktalaš universitehta, Norway, on January
16th 2015, and consisted of poster sessions, three talks, two tutorials, and an invited speech. The articles related to poster sessions and the talks are included in this proceedings.
—Tommi A Pirinen, Francis M. Tyers, Trond Trosterud, Conference organisers, 2015, Tromsø
Organisers

• Tommi A. Pirinen, Ollscoil Chathair Bhaile Átha Cliath
• Francis M. Tyers, UiT Norgga árktalaš universitehta
• Trond Trosterud, UiT Norgga árktalaš universitehta
Programme committee

- Тимофей Архангельский, Национальный исследовательский университет "Высшая школа экономики"
- Lars Borin, Göteborgs universitet
- Марина Сергеевна Федина, Финн-йогра кывъяслы информатика отсёг кузя регионалостсга лаборатория
- Mark Fishel, Tartu ülikool
- Mikel L. Forcada, Universitat d’Alacant
- Mans Hulden, University of Colorado at Boulder
- Heiki-Jaan Kaalep, Tartu ülikool
- Андрас Корнаи, Budapesti Műszaki és Gazdaságtudományi Egyetem
- Krister Lindén, Helsingin yliopisto
- Tommi A. Pirinen, Ollscoil Chathair Bhaile Átha Cliath
- Gabór Prószéky, Pázmány Péter Katolikus Egyetem
- Aarne Ranta, Chalmers tekniska högskola
- Jack Rueter, Helsingin yliopisto
- Trond Trosterud, UiT Norgga árktalaš universitehta
- Francis M. Tyers, UiT Norgga árktalaš universitehta
- Sami Virpioja, Aalto-yliopisto
- Anssi Yli-Jyrä, Helsingin yliopisto

vii
Contents

1 Invited speech
  1.1 Direct comparison of language forms in two-level framework 1

2 Tutorials
  2.1 Grammatical Framework Tutorial with a Focus on Fenno-Ugric Languages 4
  2.2 Language Documentation meets Language Technology 8

3 Accepted Papers
  3.1 Low-Resource Active Learning of North Sámi Morphological Segmentation 20
  3.2 Compiling the Uralic Dataset for NorthEuraLex a Lexicostatistical Database of Northern Eurasia 34
  3.3 Can Morphological Analyzers Improve the Quality of Optical Character Recognition? 45
  3.4 Corpus.mari-language.com: A Rudimentary Corpus Searchable by Syntactic and Morphological Patterns 57
  3.5 Infinite Monkeys of Babel — Crowdsourcing for the betterment of OCR language material 69
  3.6 Multilingual Semantic MediaWiki for Finno-Ugric dictionaries 75
  3.7 The Finno-Ugric Languages and The Internet project 87
  3.8 On the Road to a Dialect Dictionary of Khanty Postpositions 99
  3.9 FinUgRevita: Developing Language Technology Tools for Udmurt and Mans 108
  3.10 Automatic creation of bilingual dictionaries for Finno-Ugric languages 119
Chapter 1

Invited speech

1.1 Direct comparison of language forms in two-level framework
1.1. Direct comparison of language forms in two-level framework
Chapter 2

Tutorials
Grammatical Framework Tutorial, with a Focus on Fenno-Ugric Languages

Aarne Ranta and Inari Listenmaa

Göteborgs universitet och Chalmers tekniska högskola

1 GF in a nutshell

Grammatical Framework, GF, is a grammar formalism designed to support multilingual grammars. A multilingual grammar has an abstract syntax, which is shared by a set of languages. Each of these languages has a concrete syntax, which defines a relation between trees in the abstract syntax and strings in the language. For example, here is an abstract syntax tree representing predication with the verb love, subject I, and object you:

```
   PredVP
    /   \   
  i_NP  ComplV2
     /   \    
    love_V2 youSg_NP
```

With appropriate concrete syntaxes for Chinese, English, Finnish, and French, we obtain the following strings, together with word alignments determined by the tree structure:
The process of generating the strings from the tree is called **linearization**, and the reverse process is **parsing**. All GF grammars are reversible, in the sense that they can be used for both linearization and parsing.

The main application of GF is **translation**, where the abstract syntax is used as an **interlingua**. But GF can also be used for **language comparison**, because the abstract syntax gives a formally precise way to express common structures. The predication structure (PredVP in the above tree) is an example: it is a common structure for the languages shown, even though the concrete syntaxes are quite different.

Technically, GF is based on **constructive type theory**, which is used for the abstract syntax, and **parallel multiple context-free grammars** (PM-CFG), which are used for the concrete syntax. PMCFG is more expressive than context-free grammars, but it still enjoys polynomial parsing; in practice, the parsing speed is usually close to linear. The advantage of PMCFG is that it supports things like morphological variation and discontinuous constituents. This is what makes it possible to share abstract syntax for seemingly very different languages.

Writing PMCFG grammars manually would be very time-consuming and error-prone. An important feature of GF is therefore its support for **functional programming**, as a concise way of writing concrete syntaxes. Functional programming makes it easy to share common parts of code and thereby to avoid repetitive coding almost entirely. GF source code can hence be much more concise than e.g. context-free grammars or regular expressions, and GF has in fact been used as a method to generate code in these formats, independently of its
multilingual uses.

Together with the abstract syntax, the abstractions provided by functional programming methods make it possible to express linguistic generalizations. Such generalizations can be interesting for both linguistic theory (e.g. language typology) and practical engineering (reuse of code within and across languages). In particular, grammars for closely related languages can share major parts of code even in concrete syntax.

With the first release in 1998, GF has been applied to over 30 languages and has over 100 active developers around the world. Its main focus has been on controlled languages and precision-oriented translation. But the comprehensive resource grammar library (RGL) also makes it possible to build large-scale translation systems. An experimental system is currently available for 11 languages, covering all 110 language pairs.

GF is open-source software, released under licenses such as GLP (the grammar compiler) and LGPL and BSD (the run-time and RGL). This makes it possible to use GF for any purpose, including proprietary commercial applications. At least 6 companies have used GF in their projects.

2 The tutorial

The goal of the tutorial is to enable the participants to
- explore and reuse the resources currently available in GF
- contribute to the RGL, especially to its lexical resources
- get started with their own grammars

The ultimate goal is the development of RGL implementations for new Uralic languages, but some more training will probably be needed for this than this short tutorial. This can of course be done by achieved by using free material from the GF web page.

The contents of the tutorial can be divided to four lessons, which optimally need three hours in total.

2.1 Hands-on introduction

We will first show a demo of web-based translation in GF. After that, we will build a simple multilingual grammar together, by using the cloud-based grammar editor. Simple examples are enough to show that languages differ in non-trivial ways but can still share an abstract syntax.
The cloud-based grammar editor is a pedagogical tool enabling the use of GF without installing any software. Grammars created in it are also readily usable for translation and other functionalities in the GF cloud.

2.2 Tool and resource overview

We make a tour of the GF web pages to show what is available, how to install the tools, and how to reuse GF grammars on other platforms.

2.3 Morphology

We introduce the technique of smart paradigms and show how it is used to define inflection and lexica, with Finnish and Estonian inflection as examples.

2.4 Syntax

We look at some peculiarities of Finnish and Estonian agreement, word order, and nominalized verb phrase constructions as examples. For instance, we see how to deal with the rich system of participles and infinitives of Finnish in translation.

References

- GF home page: http://www.grammaticalframework.org/
- GF Summer School: http://school.grammaticalframework.org/, 19–31 July 2015 in Malta
Language Documentation meets Language Technology

Rogier Blokland
Uppsala universitet (Institutionen för moderna språk)
rogier.blokland@moderna.uu.se

Marina Fedina
Komi Republican Academy of State Service and Administration
(Centre for Innovative Language Technology)
fedinamarina@gmail.com

Ciprian Gerstenberger
UiT Norges arktiske universitet (Giellatekno)
ciprian.gerstenberger@uit.no

Niko Partanen
Albert-Ludwigs-Universität Freiburg (Skandinavisches Seminar)
niko.partanen@skandinavistik.uni-freiburg.de

Michael Rießler
Albert-Ludwigs-Universität Freiburg (Skandinavisches Seminar)
michael.riessler@skandinavistik.uni-freiburg.de

Joshua Wilbur
Albert-Ludwigs-Universität Freiburg (Skandinavisches Seminar)
joshua.wilbur@skandinavistik.uni-freiburg.de

December 16, 2014
Abstract

The paper describes work-in-progress by the Pite Saami, Kola Saami and Izhva Komi language documentation projects, all of which use similar data and technical frameworks and are carried out collaboratively in Uppsala, Tromsø, Syktyvkar and Freiburg. Our projects record and annotate spoken language data in order to provide comprehensive speech corpora as databases for future research on and for these endangered – and under-described – Uralic speech communities. Applying language technology in language documentation helps us to create more systematically annotated corpora, rather than eclectic data collections. Ultimately, the multimodal corpora created by our projects will be useful for scientifically significant quantitative investigations on these languages in the future.

1 Introduction

Language documentation (aka documentary linguistics) is an emerging sub-field of applied linguistics. Research in language documentation aims at the provision of long lasting, comprehensive, multi-faceted and multi-purpose records of linguistic practices characteristic of a given speech community. Although it evolved out of traditional fieldwork methodology used primarily by descriptive linguists and language anthropologists, language documentation is no longer merely a method, as it has its own primary aims and methodologies. One of the most important purposes of language documentation is making data available for further research on and for endangered languages, for both further theoretical and applied research, as well as for direct use by the relevant language communities. Ideally, the data pool provided by the language documenter includes a comprehensive, deeply annotated and easily accessible corpus of primary spoken language data. Metadata annotations are crucial for the intellectual accessibility of the documented data and concern both the content of the recorded speech sample (typically represented as phonological, morphological or syntactic transcriptions and translations) as well as the context (such as actors, places, speech events, but also meta-documentation about the actual project).

Along with methodologies and best practices related to fieldwork and archiving (including questions of research ethics, protection of copyrights, resource discoverability, data standards and long term data safety), the usefulness of the actual product of language documentation for linguistic research hinges on the quality and quantity of annotations as the basis for further analyses and data derivations. The use of language documentations for corpus-based investigations on endangered and less-known languages and the role of computational linguistics for the field has frequently been a driving topic over the last years. In fact, with respect to the data types involved, documentary linguistics generally seems similar to corpus building in principle. Both
provide primary data for secondary (synchronic or diachronic) data derivations and analyses. The main difference is that traditional corpus and computational linguistics deal predominantly with larger non-endangered languages for which huge amounts of mainly written corpus data are available. The documentation of endangered languages, on the other hand, results in rather small corpora of exclusively spoken genres. Furthermore, corpus annotations in language documentation projects are often created manually. Significant quantitative investigations based on corpora from language documentation projects are therefore normally excluded.

Language documentation has made huge technological progress in regard to collaborative tools and user interfaces for transcribing, archiving and browsing multimedia recordings. However, paradoxically, the field has only rarely considered applying automated methods to more efficiently (both qualitatively and quantitatively) annotate data in creating a basis for new and better corpus-based linguistic research on smaller languages.

Although the relevant methods and tools would be completely functional even for relatively small languages such as North Saami today, they are being applied exclusively for corpus-building of written language varieties. Current language technology projects on endangered languages (e.g. Giellatekno¹) seem to have simply copied their approach from already established research on larger non-endangered languages, including the focus on written language. The resulting corpora are impressively large for such minority languages, but represent a rather limited range of text genres. Furthermore, as the current written standards of small endangered languages (e.g. North Saami) are to a large part evolving as the result of institutional language planning, the bulk of texts in the North Saami corpus consists of translations from the majority languages, and even original Saami texts (e.g. on official webpages and in the few newspapers) are most typically produced only by a few writers.

The restriction on written language is even more crucial in the case of smaller languages such as Skolt Saami, for which language technology is also under development. Although active language planning for Skolt Saami was already initiated several decades ago and the amount (and quality) of written texts is ever growing, the language is still most typically used in speech only. As a consequence, there is a need to enrich the existing corpora for languages such as North Saami and Skolt Saami with new data from spoken genres. For exceptionally small Saami languages such as Pite Saami, the texts available for corpus creation are almost exclusively in non-written modi, and an efficient and consistent method for incorporating spoken texts is vital for corpus creation. In fact, spoken language documentations for these languages exist and several projects continue collecting new and annotating legacy

¹http://giellatekno.uit.no
speech samples. However, as much as endangered language documentation and language technology seem to overlap in their respective general agendas towards applied linguistic research, both fields have scarcely met so far.

Our projects are concerned with the building of multimodal corpora (at least, spoken and written, i.e., transcribed), and thus form an interface between endangered language documentation and technology. We understand language technology as the functional application of computational linguistics as it is aimed at analyzing and generating natural language in various ways and for a variety of purposes. Machine-based translation or automatic language analyzers are but two examples of such practical applications. We hope to show that all combined efforts between language technology and language documentation can clearly be directly profitable both for corpus-based theoretical investigations and for language planning and revitalization of endangered languages. Whereas the language documenters provide the speech corpora and linguistic analyzes necessary for the computational modeling of the languages in question, language technologists apply formal-descriptive linguistic and corpus linguistic methods to the programming of machine-readable grammatical and lexical descriptions of the relevant languages. Spoken language documentations can thus increase the size of the data pool utilized in computational linguistic research. Language technology, on the other hand, can create tools for effectively analyzing spoken language corpora and carrying out better linguistic documentation and description on the endangered languages in question.

2 Language Documentation meets Language Technology

This paper describes our current work on recording and annotation spoken language data and discusses the combined methods from language documentation and language technology used by our projects. The languages we are working on at present are Pite Saami, Skolt Saami, Kildin Saami² and the Izhva variety of Komi-Zyrian.³ Illustrated with data examples from our current projects we will show how language documentation profits from the application of automated corpus data annotation, specifically Finite State Transducer technology (hereinafter FST), which not only helps provide (quantitatively and qualitatively) enhanced annotations, but ultimately results in better databases useful for (quantitative and qualitative) corpus-linguistic research. Language technology, on the other hand, can profit from the use of more extensive

²http://www.skandinavistik.uni-freiburg.de/forschung/forschungsprojekte/saami
³http://komikyv.ru/page/about
and more diverse data.

In addition to designing annotation schemata of appropriate granularity for corpus building, two essential aspects of language documentation remain important for our own approach: the archiving of primary data linked to all data derivations as well as proper contextualization by means of deep metadata. By ‘deep medatada’ we mean metadata concerning a variety of levels of description in addition to basic cataloguing facts (such as time and place of a recording). Computational and corpus linguistic approaches to applied research on endangered languages (including Giellatekno) have scarcely considered the latter aspects, which are nevertheless crucial for language documentation aiming at long lasting comprehensive, multi-faceted and multi-purpose records of linguistic practices. It is also worth mentioning that our approach is perfectly in line with the endeavors made by recent programs such as CLARIN⁴ and opens digital humanities for marginalized Uralic minority speech communities specifically.

3 ELAN as a tool for annotating multimodal corpora

The language corpora we are building represent spoken and written text modi of formal and informal registers and a variety of genres. Our transcribed (in standard orthography) spoken text data as well as the written text data are stored in XML format and structured to be utilized by the multimedia annotation program ELAN.⁵ This software allows audio and video recordings to be time aligned with detailed, multi-level transcriptions, translations and further annotations. Furthermore, with ELAN basic frequency statistics can be calculated, concordances created, and data for statistical analysis exported (e.g., using R⁶ or similar tools).

Annotation tiers in the ELAN files from our projects are organized hierarchically based on the minimal template in Figure 1 for each speaking participant in a recording. Since each speaker has his/her own tier node ref, including dependent tiers, annotating simultaneous speech by multiple speakers (a common feature of spoken language) is not problematic.

While ELAN is intended mainly as an interface between written transcriptions/annotations and the original audio/video medium in which every annotation is time aligned with the medium, it is also possible to use ELAN for texts in written form and without audio/video. In this way, written legacy texts are also included in the corpora.

⁴https://www.clarin.eu
⁵ELAN is free software developed by the Technical Group of the Max Planck Institute for Psycholinguistics, see https://tla.mpi.nl/tools/tla-tools/elan.
⁶http://www.r-project.org
we create.

Metadata stored in IMDI format\(^7\) can also be linked to each ELAN annotation file in order to keep track of situational or contextual factors that are related to the data in one way or another. For instance, in order to preserve more pieces of information about the sessions, details about different speakers, the recording setting, or the instruments used, as well as about work with specific projects or persons can be included into metadata. It is also desirable to store metadata separated from basic annotations, as this makes it very easy to control access to more sensitive pieces of information that might be stored in the metadata. In our model, individual session names and actor IDs can be used to associate any metadata with any transcription.

We are already able to carry out corpus-wide searches on instances of a specific genre by using constraints, for instance, on participants’ ages or regional affiliations. This provides a solid fundament for more fine-grained sociolinguistically oriented research. In principle, the transcription files also contain small traces of metadata, as the filenames themselves are standardized to include the language ISO-code and the recording date. Yet, with this data alone it is not possible to filter results with more contextual factors. Such filtering is only possible using with the associated metadata. Furthermore, it is possible to execute complex searches on multiple ELAN files – on the entire corpus or only specific parts of it. In this, search constraints on the type of tier, contextual information, etc. and regular expressions can be specified.

Search results can be shown in a *key-words in context* (KWIC) format, i.e. in a concordance where up to eight words on either side of the search term are visible. As for exporting, all search results can be saved in plain-text comma-separated-value

\(^7\)For the ISLE Meta Data Initiative format see [http://www.mpi.nl/imdi](http://www.mpi.nl/imdi).

---

**Figure 1:** The basic ELAN tier hierarchy used in the documentation corpora
Finally, ELAN files are plain text files in XML structure (with the file extension .eaf), and as such are archive-friendly, somewhat human-readable, and will likely be supported well into the future as XML is a common and open-source format.

Figure 2: ELAN in player/annotation mode showing annotations for the overlapping speech of three speakers, the audio waveform, the accompanying video

One significant advantage of working with ELAN is that the same search functionalities of a local version of ELAN (see the description of ELAN above) can be utilized online – and thus off-site – to access all corpus files archived in the IMDI archive at the Max Planck Institute for Psycholinguistics in Nijmegen/Netherlands. For this, the tool ANNEX⁸ is an interface that links annotations and media files from the archive online (just like ELAN on a local computer). The TROVA tool⁹ can be used to perform complex searches on multi-layers of the corpus and across multiple files.

⁸https://tla.mpi.nl/tools/tla-tools/annex
⁹https://tla.mpi.nl/tools/tla-tools/trova
4 Automated FST-based corpus annotation

Unlike many other endangered language documentation projects, which annotate spoken language data manually – or occasionally semi-manually – we apply a more automated way of corpus data annotation. Using the Giellatekno infrastructure, we have started by implementing FST-based language tools for Kildin Saami and employing these for corpus annotation. Since the ELAN files are XML files, they can be accessed by virtually any programming language with XML-processing support.

The process of annotation enrichment is quite simple. The input file for the whole process is an ELAN file without part of speech, lemma, or morphological description tiers. A Python script accesses the input file, takes each item from the word form tier and passes this to the morphosyntactic analyzer. The result is then segmented into lemma, pos and morph parts, transformed into the appropriate XML structure, and then loaded back into the input file.

As it is still in an initial development phase, Kildin Saami lacks language analysis tools on higher levels such as a disambiguation or a parsing module, usually implemented by means of a Constraint Grammar. This means that the result can consist of multiple analyses for the same word form. Since the analyses are split by lemma, pos, and morph, one might think that decoupling them and putting them on different levels (see Fig. 1) would lead to even more ambiguity. For instance, the morph category Perf does not fit the pos N. Yet, this is only superficially the case, internally, ELAN has a good pointing system between the tiers, hence, it is possible to point from, for instance, the morph annotation Comp to a pos annotation A. That way, the pieces of information coming from the FST are guaranteed to be places and linked properly.

This method is also beneficial for the further development of language analysis data. As mentioned above, the resources for Kildin Saami are still in an initial phase, and therefore the FST does not produce an analysis for some word forms. In such cases, the word form under scrutiny would be assigned a specific value for non-extant results. These can then be corrected manually by means of the ELAN tool and the improvements would then flow back into the FST resources. Subsequently running a corpus analysis would then produce better results.

5 Conclusion and prospects

We hope to have shown that combining language documentation and language technology is a very promising undertaking for both fields, albeit for differing reasons. It is precisely in the overlapping areas between the two fields that a large amount of potential for the creation of resources useful in both fields can and should take place.
Up to now, these complementary resources have hardly been utilized.

The simple yet effective example presented in this paper demonstrates how our language documentation projects take advantage of various tools of language technology. As a result of using our projects’ corpora, which have both quantitatively and qualitatively superior annotations, language technology – in this case, Giellatekno – has access to new resources for further research. This is particularly the case concerning multimodal corpora, which language technology and computer linguistics for Uralic languages have hardly dealt with up to now.

Our projects are a work in progress. Currently, we have only developed a part-of-speech tagger for Kildin Saami. At the next stage, we intend to have complete morphological analyses (lemma-pos-morph) created automatically. Analyzing the corpus with help of FSTs on the morphosyntactic level can sometimes lead to cases of ambiguity. For disambiguation and syntactic analysis, Giellatekno uses Constraint Grammar (CG), which takes morphologically analyzed text as input, and ideally returns only the appropriate reading, enriched with grammatical functions and dependency relations. Since the output of a CG is a dependency structure for a particular sentence, the output may also be converted into phrase structure representations.

We plan to implement the infrastructure that we now are building for Kildin Saami for other languages for which FST already exists as well. As our projects are carried out, we will continue to supplement and revise the FSTs for these languages incrementally.

The corpus data that we will archive in the near future shall also be available to interested parties in a variety of ways. On the one hand, ANNEX and TROVA can be used to browse and search the multimodal corpora online. On a purely textual level – i.e., without links to multimedia – our corpora can also be integrated into the Korp interface (a tool for online browsing of written corpora), which is already in place for a number of languages at Giellatekno. Another possible user interface, which is particularly useful for language users, is an integrated dictionary with links to corpus data, such as Neahttadigisánit – this already works very well for North Saami. However, this interface is only textual, and does not have any links to multimedia recordings.

---

¹⁰Cf. the respective documentation at Giellatekno.

¹¹http://gtweb.uit.no/korp

¹²http://sanit.oahpa.no
References


Chapter 3

Accepted Papers
Low-Resource Active Learning of North Sámi Morphological Segmentation

Stig-Arne Grönroos¹
stig-arne.gronroos@aalto.fi

Kristiina Jokinen²
kristiina.jokinen@helsinki.fi

Katri Hiovain²
katri.hiovain@helsinki.fi

Mikko Kurimo¹
mikko.kurimo@aalto.fi

Sami Virpioja³
sami.virpioja@aalto.fi

¹Department of Signal Processing and Acoustics, Aalto University, Finland
²Institute of Behavioural Sciences, University of Helsinki, Finland
³Department of Information and Computer Science, Aalto University, Finland

December 15, 2014

Abstract

Many Uralic languages have a rich morphological structure, but lack tools of morphological analysis needed for efficient language processing. While creating a high-quality morphological analyzer requires a significant amount of expert labor, data-driven approaches may provide sufficient quality for many applications. We study how to create a statistical model for morphological segmentation of North Sámi language with a large unannotated corpus and a small amount of human-annotated word forms selected using an active learning approach. For statistical learning, we use the semi-supervised Morfessor Baseline and FlatCat methods. After annotating 237 words with our active learning setup, we improve morph boundary recall over 20% with no loss of precision.

This work is licensed under a Creative Commons Attribution–NoDerivatives 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by-nd/4.0/
1 Introduction

In morphologically rich languages, such as the Uralic languages, the number of observed word forms grows rapidly with increasing corpus size. This vocabulary growth can be problematic for natural language processing (NLP) applications, because it causes sparsity in the calculated statistics. Thus it is essential to model such languages on a sub-word level, using for example morphological analysis.

Despite the improvement of development tools and increase of computational resources since the introduction of finite-state transducer (FST) based morphological analyzers in the 1980s [1], the bottleneck for the traditional method of building such analyzers is still the large amounts of manual labor and skill that are required [2]. The strength of such analyzers is the potential to produce output of high quality and richly informative morphological tags.

Morphological surface segmentation is a relaxed variant of morphological analysis, in which the surface form of a word is divided into segments that correspond to morphemes. The segments, called morphs, are not mapped onto underlying abstract morphemes as in FST-based analyzers, but concatenating the sequence of morphs results directly in the observed word form. Allomorphic variation is left unresolved.

Although unsupervised learning of morphological segmenters does not reach the detail and accuracy of hand-built analyzers, it has proven useful for many NLP applications, including speech recognition [3], information retrieval [4], and machine translation [5]. Unsupervised methods are especially valuable for low-resource languages, as they do not require any expensive resources produced by human experts.

While hand built morphological analyzers and large annotated corpora may be unavailable due to the expense, a small amount of linguistic expertise is easier to obtain. Given word forms embedded in sentence contexts, a well-informed native speaker of a language can mark the prefixes, stems, and suffixes of the words in question. A brief collection effort of this type will result in a very small set of annotated words.

Small annotated data of this type can be used to augment large unannotated data by using semi-supervised methods, which are able to learn from such mixed data. As little as one hundred manually segmented words have been shown to provide significant improvements to the quality of the output when comparing to a linguistic gold standard [6]. Adding more annotated data improves the results, with rapid improvement to one thousand words or beyond.

When gathering annotated training samples for a specific model, active learning may provide better results than selecting the samples randomly. In each iteration of active learning, the current best model, trained with all training samples collected up to that point, is used in selection of the new samples to annotate for the next iteration. In this work, we use active learning for morphological segmentation of North Sámi.
1.1 North Sámi

North Sámi (davvisámegiella) belongs to Finno-Ugrian languages and is related to Finnish and other Baltic-Finnic languages. It is one of the nine Sámi languages spoken in the Northern Polar Cap, and spread along Norway, Sweden, Finland and Russia. The speakers of the Sámi languages do not necessarily understand each other but the languages form a chain of adjacent groups. North Sámi is the most widely used Sámi language with around 20,000 speakers, functioning as a lingua franca among the Sámi speakers and used in textbooks, children’s books, newspapers, and broadcasts.

Linguistically North Sámi is characterized as an inflected language, with cases, numbers, persons, tense and mood. The inflectional system has seven cases. It is accompanied by complicated although regular morphophonological variation. The inflected forms follow weak and strong grades which concern almost all consonants. North Sámi is also fusional: a single word form can stand for more than one morphological category. The nouns have four inflection categories (stems with a vowel or a consonant, the so-called contracting is-nouns, and alternating u-nouns), while the verbs have three conjugation categories (gradation, three syllable, two syllable). The only one syllable verbs are “leat” (to be) and the negation verb. In syntax, the Sámi has separate dual forms for pronouns and verbs besides singular and plural forms.

1.2 Related work

While unsupervised morphological segmentation has recently been an active topic of research [7], semi-supervised morphological segmentation has not received as much attention. One approach is to seed the learning with a small amount of linguistic knowledge in addition to the unannotated corpus [8]. Some semi-supervised methods where a part of the training corpus is supplied with correct outputs have also been presented, including generative [6, 9, 10] and discriminative [11, 12] methods.

Active learning methods have been applied for constructing FST-based analyzers by eliciting new rules from a user with linguistic expertise [13, 14]. These development efforts are fast for rule-based systems, but still require months of work. There has been research effort into FST-based morphology for Sámi languages [15, 16, 17]

North Sámi is the focus of the DigiSami project, which attempts to increase the digital viability of minor Finno-Ugric languages by technology development, analysis, data collection (read and conversational speech), and encouragement of community effort in online content creation [18, 19]. This work directly supports the ultimate goal of the project, which is to produce tools and technology that would allow Sámi speech-based applications to be developed. Although North Sámi has various linguistic resources, there are not many related to speech technology.
2 Methods

As a method for morphological segmentation of words, we use Morfessor. It is a family of methods for learning morphological segmentations primarily from unannotated data. The methods are based on a generative probabilistic model which generates the observed word forms by concatenating morphs. The model parameters $\theta$ define a morph lexicon. The morph $m_i$ is considered to be stored in the morph lexicon, if it has a non-zero probability $P(m_i | \theta)$ given the parameters.

Morfessor utilizes a prior distribution $P(\theta)$ over morph lexicons, derived from the Minimum Description Length principle [20]. The prior favors lexicons that contain fewer, shorter morphs. The purpose is to balance the size of the lexicon, and the size of the corpus $D$ when encoded using the lexicon. This balance can be expressed as finding the following Maximum a Posteriori (MAP) estimate:

$$\hat{\theta} = \arg \max_{\theta} P(D | \theta) = \arg \min_{\theta} \left( -\log P(\theta) - \log P(D | \theta) \right).$$

(1)

In the Morfessor variants used in this work, the lexicon encodes the forms of the morphs directly as strings: each letter requires a certain number of bits to encode.

2.1 Morfessor Baseline

Morfessor Baseline [21, 22] employs a morph lexicon $P(m_i | \theta)$ that is simply a categorical distribution over morphs $m_i$ in other words a unigram model. The model parameters $\theta$ are optimized utilizing a greedy local search, in which one training word at a time is reanalyzed and the model parameters updated accordingly.

In order to use the annotations produced in the active learning for training Morfessor, we employ the semi-supervised extension to Morfessor Baseline [6]. This involves replacing the MAP estimate (1) with the optimization

$$\hat{\theta} = \arg \min_{\theta} \left( -\log P(\theta) - \log \alpha P(D | \theta) - \log \beta P(A | \theta) \right),$$

(2)

where $D$ is the unannotated and $A$ the annotated training corpus, $\alpha$ and $\beta$ are the weights for the likelihood of the unannotated corpus and annotated corpus, respectively. The hyper-parameters $\alpha$ and $\beta$ affect the overall amount of segmentation and the relative importance of using the morphs present in the annotated corpus.

2.2 Morfessor FlatCat

The most recent Morfessor variant is called Morfessor FlatCat [10]. The main difference between Morfessor Baseline and Morfessor FlatCat is the use of morph categories
in the latter. Each morph token is categorized as **PREFIX**, **STEM**, or **SUFFIX**. Internally to the algorithm, a **NON-MORPH** category is used, intended to model frequent substrings that are not morphs but fragments of a morph. Word formation is modeled using a hidden Markov model (HMM) having morph categories as hidden states and morphs as observations. HMM morphotactics were previously used in the Categories-ML [23] and Categories-MAP [24] variants of Morfessor, but Morfessor FlatCat is the first method to combine the approach with semi-supervised training.

The benefit of the HMM morphotactics is increased context-sensitivity, which improves the precision of the segmentation. For example, in English, the model can prevent splitting a single s, a common suffix, from the beginning of a word. Modeling of morphotactics also improves the segmentation of compound words, by allowing the overall level of segmentation to be increased. The main benefits of semi-supervised learning are in the modeling of suffixation. [10]

The prior of Morfessor FlatCat is otherwise the same as in Morfessor Baseline, but it also includes encoding of the right and left perplexity of the morph. The perplexity measures describe the predictability of the contexts in which the morph occurs. The perplexities, together with the length of the morph, are used to calculate the emission probability of a morph conditioned on the morph category, $P(m \mid c)$.

### 2.3 Pool-based active learning

Pool-based active learning [25] has been successfully applied in NLP [26]. In pool-based active learning, the system has access to a pool of unlabeled data $A$ and can request from the annotator true labels for a certain number of samples in the pool.

A method for choosing which samples to annotate still needs to be defined. A well suited approach for generative models is to use the model’s estimate of the uncertainty of the decision associated with a particular sample in order to select the additional samples to annotate [27]. In the case of morphological segmentation, we use the uncertainty of a word’s current segmentation in order to assess its value as an additional annotation. The next word to annotate $A_{(t+1)}$ at time step $t$ is selected from $A$ based on the uncertainty of the current best segmentation $Z_i$

$$A_{(t+1)} = \arg \min_{w_i \in A} \frac{P(Z_i \mid \theta_t)}{P(w_i \mid \theta_t)},$$

where the likelihood of the current segmentation $P(Z_i \mid \theta_t)$ is given by the Viterbi algorithm [28] and the likelihood of the word with any segmentation $P(w_i \mid \theta_t)$ is given by the forward algorithm [29].
<table>
<thead>
<tr>
<th>Corpus</th>
<th>Word tokens</th>
<th>Word types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Den samiske tekstbanken</td>
<td>17 985 140</td>
<td>691 190</td>
</tr>
<tr>
<td>UIT-SME-TTS</td>
<td>42 150</td>
<td>8194</td>
</tr>
<tr>
<td>Development set</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>Evaluation pool</td>
<td></td>
<td>900</td>
</tr>
<tr>
<td>Training pool</td>
<td></td>
<td>7194</td>
</tr>
</tbody>
</table>

Table 1: Sizes of the unannotated corpora and the initial division into subsets.

3 Experiments

We used two different text corpora in our experiments. The sizes of the corpora are shown in Table 1. The larger *Den samiske tekstbanken* corpus was only used to construct a word list, to use as the unannotated training data. The smaller *UIT-SME-TTS* corpus was divided into separate pools from which evaluation and training words were drawn for annotation. The sentences in which the words occur were also extracted for use as contexts. To ensure that the evaluation words are unseen, the words in the evaluation pool were removed from the other subsets.

The use of two corpora enables the release of the annotations with their sentence contexts, which would have been precluded by the restrictive corpus of the *Tekstbanken* corpus. It also demonstrates the effectiveness of the system under the realistic scenario where a large general-domain word list for the language is available for use, even though the corpora themselves are restricted by licensing. A similar scenario would be selection from a specific target domain corpus.

Initially we use Morfessor Baseline, but towards the end of the experiment we switch the method to Morfessor FlatCat. As prefixes are very rare in North Sámi, and none were seen in the annotations, we disabled the prefix category.

3.1 Active learning

Our active learning procedure starts from nothing but an unannotated corpus collected for other purposes. An initial model is trained in an unsupervised fashion. The procedure then applies three components iteratively: (i) selection of new words to annotate using the current model, (ii) elicitation of annotations for the selected words, and (iii) training of the new segmentation model using all available training data.

For the elicitation step, we developed a web-based annotation interface. A javascript app using the jQuery framework was used as a front-end and a RESTful Python wsgi-app built on the bottle framework as a back-end. Screenshots of the annotation
Figure 1: Screenshots of the annotation interface.

interface are shown in Figure 1. For words in the training pool, the interface shows the segmentation of the current model as a suggestion to the annotator. Words in the evaluation pool are shown unsegmented, in order not to bias the annotator.

There are no efficient on-line training algorithms for Morfessor FlatCat. Thus we gather a list of 50 new words to annotate, by ranking the potential words according to (3), and re-train once the whole list has been annotated. Re-training includes hyper-parameter optimization (HPO) for $\alpha$ and $\beta$. Due to the very limited amount of training data, and a lack of previously collected annotated development set, we initially decided to use 3-fold cross-validation on the annotated training set for HPO. This initial approach was quickly shown to be flawed, as the values of the hyper-parameters did not begin to converge after multiple iterations. This divergence can be explained by HPO requiring the development set to be an unbiased sample of the data distribution. A subset biased towards maximally informative words is desired for use as training words, but using them for HPO introduces an undesired bias.

To remedy this situation, we constructed a development set of 100 randomly selected words with annotations. We then restarted the iterations, now using the development set for HPO. In this second approach, when a word that had already been annotated was reselected for annotation, the old annotation was used, making it unnecessary to re-elicit from the annotator. The training iterations and the respective hyper-parameter values are shown in Table 2.

### 3.2 Annotation details

The annotations were produced by a single trained linguist, who is not a native speaker of Sámi. In total 457 randomly selected word types and 346 actively selected word types were annotated under a time span of 17 days. The total time spent by the annotator was 19 hours (over 30 min breaks omitted).
### Table 2: The model parameters and number of annotations for the active learning iterations. U and S stand for unsupervised and semi-supervised training, C and D for setting hyper-parameters by cross-validation and development set, respectively.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Training</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>Dev</th>
<th>Train</th>
<th>Tot</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>C 0</td>
<td>U Baseline</td>
<td>0.42</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>0</td>
<td>.67</td>
</tr>
<tr>
<td>C 1</td>
<td>U Baseline</td>
<td>1.3</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>49</td>
<td>.68</td>
</tr>
<tr>
<td>C 2</td>
<td>S Baseline</td>
<td>2.4</td>
<td>1300</td>
<td>–</td>
<td>98</td>
<td>98</td>
<td>.67</td>
</tr>
<tr>
<td>C 3</td>
<td>S Baseline</td>
<td>2.7</td>
<td>800</td>
<td>–</td>
<td>148</td>
<td>148</td>
<td>.66</td>
</tr>
<tr>
<td>C 4</td>
<td>S Baseline</td>
<td>3.4</td>
<td>900</td>
<td>–</td>
<td>198</td>
<td>198</td>
<td>.66</td>
</tr>
<tr>
<td>D 0</td>
<td>U Baseline</td>
<td>1.1</td>
<td>–</td>
<td>100</td>
<td>–</td>
<td>100</td>
<td>.68</td>
</tr>
<tr>
<td>D 1</td>
<td>S Baseline</td>
<td>1.4</td>
<td>800</td>
<td>100</td>
<td>50</td>
<td>150</td>
<td>.69</td>
</tr>
<tr>
<td>D 2</td>
<td>S Baseline</td>
<td>1.5</td>
<td>700</td>
<td>100</td>
<td>123</td>
<td>223</td>
<td>.70</td>
</tr>
<tr>
<td>D 3</td>
<td>S FlatCat</td>
<td>0.2</td>
<td>1400</td>
<td>100</td>
<td>182</td>
<td>282</td>
<td>.74</td>
</tr>
<tr>
<td>D 4</td>
<td>S FlatCat</td>
<td>0.5</td>
<td>2200</td>
<td>100</td>
<td>237</td>
<td>337</td>
<td>.76</td>
</tr>
</tbody>
</table>

Most of the annotated word tokens had an unambiguous segmentation agreeing with established linguistic interpretation. These words contain only easily separated suffixes: markers for case and person, and derivational endings. However, some words required the annotator to make choices on where to place the boundary.

One challenge was posed by the extensive stem alternation and fusion in Sámi. To maximize consistency, the segmentation boundary was usually placed so that all of the morphophonological alternation remains in the stem. Exceptions include the passive derivational suffix, which is found as variants -ojuvvo- and -juvvo- depending on the inflectional category and stem type. Another challenge were lexicalized stems. These stems appear to end with a derivational suffix, but removal of the suffix does not yield a morpheme at all, or results in a morpheme with very weak semantic relation to the lexicalized stem. An example is râhkadit (make, produce).

### 3.3 Evaluation

The word segmentations generated by the model are evaluated by comparison with annotated morph boundaries using boundary precision, boundary recall, and boundary $F_1$-score [30]. The boundary $F_1$-score equals the harmonic mean of precision (the percentage of correctly assigned boundaries with respect to all assigned boundaries) and recall (the percentage of correctly assigned boundaries with respect to the reference boundaries). Precision and recall are calculated using macro-averages over the
words in the evaluation set. In the case that the word has more than one annotated segmentation, we take the one that gives the highest score.

We also report the scores for subsets of words consisting of different morph category patterns found in the evaluation set. These categories are words that should not be segmented (STM), compound words consisting of exactly two stems (STM+STM), a stem followed by a single suffix (STM+SUF) and a stem and exactly two suffixes (STM+SUF+SUF). Only precision is reported for the STM pattern, as recall is not defined for an empty set of true boundaries.

3.4 Results

Figure 2 shows the improvement of the $F_1$-score as more annotations became available. Training a Morfessor FlatCat model after three iterations provided a large boost, even though the annotated words had so far been selected by Baseline models. In contrast, the words selected by the FlatCat model (3FC) for annotation did not benefit the Baseline model (4BL).

Table 3 shows scores for sets of words with different morphological patterns. For the full test set, we improve morph boundary recall over 20% (relative) with no loss of precision, when comparing the first model of the second approach (D0) to the last
Table 3: Boundary precision (Pre), recall (Rec), and $F_1$-scores for different subsets of the evaluation data.

<table>
<thead>
<tr>
<th>Model</th>
<th>STM</th>
<th>STM+STM</th>
<th>STM+SUF</th>
<th>STM+SUF+SUF</th>
<th>Full test set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pre</td>
<td>Pre</td>
<td>Rec</td>
<td>$F_1$</td>
<td>Pre</td>
</tr>
<tr>
<td>C0 Baseline</td>
<td>.57</td>
<td>.85</td>
<td>.88</td>
<td>.87</td>
<td>.63</td>
</tr>
<tr>
<td>D0 Baseline</td>
<td>.70</td>
<td>.89</td>
<td>.85</td>
<td>.87</td>
<td>.74</td>
</tr>
<tr>
<td>D4 FlatCat</td>
<td>.73</td>
<td>.85</td>
<td>.92</td>
<td>.89</td>
<td>.75</td>
</tr>
</tbody>
</table>

model (D4). The performance has improved for all morph patterns. The $\text{STM+SUF}$ pattern has the largest increase, with improvements both in precision and recall. The recall scores of compound words ($\text{STM+STM}$) and suffix sequences ($\text{STM+SUF+SUF}$) are also clearly improved.

4 Conclusions

We have applied an active learning approach to modeling morphological segmentation of North Sámi. The work was accomplished using open-source software \(^1\). We present the collected language resources for the use of the scientific community \(^2\).

The performance of the segmentation model was shown to increase rapidly as the amount of human-annotated data was increased. One of our findings is the importance of collecting an unbiased development set for optimization of hyper-parameters, even though this reduces the amount of human-labeled data available for training. Cross-validating using the selected samples is not an adequate compromise.

One avenue for future work is exploring other measures for selecting the words to annotate. These can include applying other language models, but can also be based on direct statistics of the language, e.g. frequencies and lengths of the words or substrings of the words. A thorough comparison to random selection should also be performed. Another question is how well this approach extends to other languages and corpora.

Acknowledgments

This research has been supported by EC’s Seventh Framework Programme under grant n°287678 and the Academy of Finland under the Finnish Centre of Excellence Program 2012–2017 (grant n°251170), LASTU Programme (grants n°256887 and 259934) and Fenno-Ugric Digital Citizens (grant n°270082). Computer resources within the Aalto University School of Science “Science-IT” project were used.

---


References


Abstract

This paper presents a large comparative lexical database which covers about a thousand concepts across twenty Uralic languages. The dataset will be released as the first part of NorthEuraLex, a lexicostatistical database of Northern Eurasia which is being compiled within the EVOLAEMP project.

The chief purpose of the lexical database is to serve as a basis of benchmarks for different tasks within computational historical linguistics, but it might also be valuable to researchers who work on the application of computational methods to open research questions within the language family.

The paper describes and motivates the decisions taken concerning data collection methodology, also discussing some of the problems involved in compiling and unifying data from lexical resources in six different gloss languages.

The dataset is already publicly available in various PDF formats for inspection and review, and is scheduled for release in machine-readable form in early 2015.

1 Introduction

Recent years have seen a surge of interest in computational historical linguistics, where computational methods are used to analyze phenomena of interest to historical linguistics, such as language relationships, language contacts, and language change.
The main focus of the field has been on applying phylogenetic methods from bioinformatics to automated language classification. These methods can be separated into two basic approaches which differ in the type of data they operate on.

In the more popular character-based approaches, expert cognacy judgments are encoded as binary matrices where each line represents the presence or absence of some cognate set in each language. Character-based approaches have recently started to be applied to Uralic languages [1].

In the competing distance-based approaches, phonetic representations of word forms are used to compute a measure of lexical distance between languages. While in general, these approaches perform worse than character-based approaches in the language classification task, they have the advantage of much wider applicability because they do not rely on expert cognacy judgments. Furthermore, work on actual realizations is closer to the interests and goals of mainstream historical linguistics. For instance, work in the field includes the automation of some parts of the comparative method, such as the detection of regular sound correspondences.

A major weakness of the datasets commonly used in research on distance-based methods is that the concept lists they build on are very short (40 concepts in the case of the global-coverage ASJP database [2], and Swadesh lists of not more than 200 concepts in sources for individual language families). While longer lists provably do not lead to higher performance in current approaches to language classification [3], such short lists are clearly not sufficient for more advanced comparative tasks.

One of the key goals of the EVOLAEMP project at the University of Tübingen is to improve the state of the art in distance-based approaches. The small size of the word lists in existing databases necessarily makes large-scale data collection a major part of our work. We are collecting substantial amounts of lexicostatistical data for more than a hundred languages of Northern Eurasia, which we plan to release under the name NorthEuraLex.

Since the Uralic family is among the most thoroughly investigated language families, we do not expect to find out anything new about their internal classification based on this dataset. Given the extensive knowledge of cognate sets gained in more than two centuries of Uralic studies, character-based methods as applied by [1] will remain the preferred methods for automated classification within the family.

However, the wealth of established knowledge also makes the Uralic languages an ideal benchmark for other methods, since automatically extracted sound correspondences and cognacy judgments can be evaluated against comprehensive expert knowledge. The central role of Uralic in the EVOLAEMP project is motivated by this fact, and the advantages of choosing a relatively small and therefore tractable family compared to the even more thoroughly researched Indo-European languages.

In addition to Uralic, NorthEuraLex is intended to also cover the contact languages
of all branches of Uralic throughout their development (including samples from all branches of Indo-European, and from all the language families sometimes summarized as Altaic and Paleosiberian), with promising perspectives for evaluating computational models of language contact.

With the subproject of compiling the Uralic dataset close to completion, the author (who is responsible for Uralic and Paleosiberian data collection within the project) has decided to release this data set to the Uralist community in order to receive feedback on errors and possible improvements. Collaboration with experts in the individual languages as well as in computational methods for Uralic languages are the necessary next steps for improving the quality of the dataset.

2 Data Collection

This section describes and motivates the many decisions taken during data collection, including the choice of languages, concepts, and sources. As an introductory remark, it should be emphasized that our goal is not to compile high-coverage general-purpose electronic dictionaries, but a lexicostatistical database. This means that only the most common or natural realization for each concept is to be included, not all the synonyms commonly listed in large dictionaries. Multiple translations are only included in the very few cases where there is no clear preference for one realization. As an example, the database only contains the Finnish word *koira* for “dog”, rather than the dialectal *hunttu*, the poetic *hurtta*, and even the archaic *peni*, even though the latter is most interesting to the Uralist because of its cognates in many other branches.

2.1 The Language Sample

The database covers a sample of twenty Uralic languages, chosen mainly according to the availability of high-coverage lexical resources, but also in order to cover all branches of the family. In the following list, the twenty selected languages are given with the corresponding ISO 639-3 language codes, which are used throughout the released resources as well as this paper as shorthands:

2.2 Selecting the Concepts

Since the concept lists used in lexicostatistical work are typically much shorter than our goal of 1,000 concepts, we decided to build our own concept list for data collection. The concept list used by [1] mainly builds on Swadesh-type lists of stable concepts such as the Leipzig-Jakarta list [4], and a list of 100 concepts especially selected for their stability within Uralic. However, they also sample a hundred concepts from the 1,460-item basic vocabulary list used by the World Loanword Database WOLD [5] to assess the consequences of using a set of less stable concepts.

While our list includes all concepts from the various Swadesh-type lists, basing the extension to the desired size on the full WOLD list would have led to the inclusion of many concepts which have appeared in the material culture of many Uralic minorities only very recently, so that the frequency of loanwords in a database based on such a list would be very high.

Moreover, for some of the languages in our sample (sjd, mns, kca, sel, nio), the best available dictionaries only contain a few thousand entries. Often, the space which recent Russian loans would have occupied is saved to provide better coverage of the inherited lexicon. In the interest of achieving full coverage for as large a concept set as possible, our concept set is therefore based partly on data availability.

Even within the inherited lexicon, many concepts which are only relevant to some climate zones or modes of subsistence (“tent”, “reindeer”, different types of boats and sleighs), had to be left out because of low overall dictionary coverage. Conversely, we opted to include some very commonly borrowed words with near-complete coverage (e.g. “bread”, “book”, “church”, names of weekdays and months) as useful benchmarks for loanword detection.

Our current list of 1,016 concepts consists of about 480 nouns (including 48 parts of the body and 35 animal names), 100 adjectives, 45 adverbs, 340 verbs, and 50 words from smaller word classes (pronouns, question words, numerals). For inspection of the concept list, the reader is referred to the table of contents in the preliminary PDF release of the database (see below).

2.3 Sources

While sufficient lexical resources are available for all the languages in the sample, a major obstacle to data collection was that to build on the best resources for each language, six different gloss languages had to be bridged. In order of relevance, the six most important gloss languages were Russian, German, Norwegian, Finnish, English, and Estonian. In order to clarify translations, some additional information needed to be retrieved from sources in Swedish, Hungarian, and Latvian.
Within the EVOLAEMP project, German serves as the pivot language for the entire lexical infrastructure. Choosing the native language of most project members minimizes the risk of additional errors and imprecisions which occur when glosses are written by non-native speakers of the gloss language. German is an especially useful choice for Uralic because of its long tradition as the primary language of the field, leading to high availability of dictionaries for target languages across the sample. To make the data more accessible, English and Russian glosses are included in our databases, sometimes also serving as an informal way of disambiguating polysemous German glosses.

To avoid duplication of effort, freely available electronic resources were used wherever possible. This includes downloadable lexical databases such as the mhr-eng dictionary developed by the Mari Web Project¹, and the electronic dictionaries maintained by the Giellatekno project². For four languages (krl, liv, sjd, mhr), online dictionaries were used either because they constitute the highest-coverage resources for their respective languages, or because their contents are identical to the best published dictionaries, making them convenient tools for access to printed material.

While these electronic resources were extremely helpful for data collection, the bulk of the data still needed to be extracted from dictionaries published on paper. We avoided using etymological dictionaries as primary sources because of their possible bias towards including lexemes with parallels in related languages (see the *peni* example above), instead of the most natural realizations in the current standard languages. Whenever available, school dictionaries were preferred over these more comparatively oriented dictionaries. Despite their shortcomings in accurately representing pronunciation, the standardized orthographies employed by these dictionaries make it easier to aggregate information across resources, and to abstract away from dialectal distinctions that would otherwise make it very hard for a non-specialist to compile a word list which consistently represents a single variant. School dictionaries, which tend to only give a few translations sorted by frequency or salience, are the most reliable source if one wants to find the most common realization for some basic concept in an unfamiliar target language. Still, large dictionaries of the standard language and scientific dictionaries were found invaluable as fallback options for problematic cases. A full list of all sources employed can be found on the author’s website, and is also distributed together with the data.

¹http://www.univie.ac.at/maridict/site-2014/
²http://giellatekno.uit.no/words/dicts/
2.4 Methodology

The process of compiling the wordlist for a language (notation: lng) in NorthEuraLex is organized in five stages. First, the initial German concept list is translated into the relevant gloss language (notation: glo), with the requirement that at least one resource which allows lookup in both directions (glo-lng and lng-glo) must be available. Resources where both directions are treated separately are always preferred to resources where one of the two directions is only available as an index or a mechanical reordering of the translation pairs in the other.

In the second stage, the gloss language lemmas are looked up in the glo-lng direction. All lng lemmas occurring in the relevant dictionary equations are collected, possibly extended by annotations given in the sources. The purpose of this stage is to collect a set of lng lemmas which cover the concept set as completely as possible.

The third stage consists in looking up the collected lemmas in the lng-glo resource. The glo lemmas corresponding to each lng lemma are collected, including all polysemies and annotations. In many cases, the example sentences given by the dictionaries are interpreted for further usage hints, which are stored as additional annotations for later reference.

The fourth stage is where information in different gloss languages is aggregated by translating the lng-glo entries into lng-deu entries. This process is assisted by additional electronic dictionaries in both relevant directions (glo-deu and deu-glo) to ensure consistency of translations across target languages.

In the fifth stage, the decisions on which lng lemmas to include for each concept are made in a final pass over the entire data. This step involves a complex and not fully formalizable decision process based on best fit of glosses, position of the translation pairs in the lng-glo and glo-lng resources, the disambiguating information collected in stages 2 and 3, some ad-hoc research in online resources such as Wikipedia entries in various gloss and target languages, and the author’s varying levels of familiarity with the languages involved. In order to reduce the impact of translation errors, the glosses in the translated lng-deu entries are only used as indices bridging the different gloss languages, while the decision process itself only relies on the lng-glo and glo-lng entries extracted from the original resources.

2.5 Issues and Difficulties

Of the many problems encountered in the various stages of data collection, the problem of picking a small set of good gloss language lemmas for each concept turned out as being the most crucial and problematic. Beyond the expected problem of polysemous glosses that overlap between gloss languages, one of the most challenging
problems for unifiability is the different granularity of meanings lexicalized in a given domain in different gloss languages.

As a case in point, consider the representation of verbal meanings in Russian. In stage 2 of our lookup process, we used the following six Russian equivalents of the German verb *legen* “to lay”: клать, положить, укладывать, уложить, складывать, сложить. While German can also express many of the distinctions expressed by the Russian equivalents using prefixes to the basic verb (*weglegen*, *ablegen*, etc.), the prefixless verb form is clearly perceived as most basic, and can therefore be expected to be used in any deu-1ng dictionary to index the 1ng equivalents of “to lay”.

By contrast, it is largely unpredictable which of the corresponding Russian verbs is used for this purpose in a small rus-1ng dictionary. While this merely complicates the lookup process, the real problem is the treatment of verbs in larger dictionaries. Where the Russian verbs lexicalize slight differences in grammatical aspect and other meaning components, regular derivational morphology of the target language is often used to faithfully represent these nuances in rus-1ng dictionary equations. For a significant part of the verbal concepts in our list, this leads to a very large number of candidate lexemes. For instance, our lookup process leads to not less than 16 possible kpv equivalents for *legen*.

The task of selecting the single most natural equivalent of the German verb from such a list is a challenge even with good knowledge of the target language, and cannot reliably be accomplished by a non-expert. Our preliminary solution is to adapt a selection process based on a hierarchy of preferences. Preference is first given to glosses mentioned earlier in the rus-1ng direction, then to non-derived 1ng verbs, then to lexemes where multiple equivalents of the intended concept are mentioned early in the 1ng-rus direction, and finally, to less polysemous glosses. For instance, *положить* receives more weight than *сложить* as an equivalent of “to lay” because the latter can also mean “to fold up”.

Measured against the goal of retrieving the most natural lexeme for each concept, our data collection strategy inevitably leads to many erroneous entries due to missing frequency information, in addition to the problems caused by misinterpretation of dictionary entries in one of the less familiar gloss languages. Without good knowledge of all the target languages, better results could only be achieved by considering parallel texts which describe many prototypical situations. However, in the absence of parallel corpora of any useful size for any of the minority languages, this would be a large project in itself, requiring extensive fieldwork. Contributions by experts on all the target languages are therefore needed if the quality of the dataset, especially in the verbal domain, is to be improved much beyond the current state.
3 Phonetic Transcription

In a lexicostatistical database, some form of unified phonetic description is needed to achieve comparability across languages. The aim for such a description is to be as faithful as possible, while still abstracting away from dialectal and speaker-specific phenomena. While within Uralic studies, there are good reasons for using the traditional Uralic Phonetic Alphabet (UPA) for such purposes, our database requires a unified phonetic notation that also covers all the relevant contact languages, and is as accessible as possible to non-Uralists. For these reasons, we decided to use the International Phonetic Alphabet (IPA) as the common representation system.

To implement the orthography-to-IPA transducers, sets of very simple greedy replacement rules were compiled for each language. For many languages, a second processing stage for treating palatalization as represented in Russian-based Cyrillic orthographies was needed. While simple toolchains of this type work reasonably well in most cases, a number of imperfections and challenges remain.

An obvious challenge is any information that is not immediately visible in the dictionary forms because the standard orthography does not fully specify or faithfully represent pronunciation. Examples include the non-representation of non-phonemic weakly voiced vowels in Tundra Nenets and Skolt Sami, palatalization in the nominative case of some Estonian nouns (caused by an elided front vowel which is only visible in other case forms), and epenthetic vowels which split up consonant clusters in Northern Sami and other languages. While in many cases, some effort was made to predict and implement these phenomena, in others we opted to reduce complexity by aiming for a phonemic notation that more closely corresponds to the orthography.

Another area of difficulty is the treatment of suprasegmental phenomena. This includes issues like the impact of stress on vowel quality in Moksha and other languages, and suprasegmental palatalization in Skolt Sami. Since a correct implementation of these phenomena presupposes a level of understanding which is difficult to derive from literature alone without the help of experts, they are not yet fully covered by the current version of our transliterators.

Since many of these distinctions are not of central importance to historical linguistics, we decided that even a transcription which does not fully cover these phenomena was good enough for a first release. Even though the shortcomings leave the current transcriptions in an imperfect state somewhere between the phonetic and phonemic levels, already the current version is detailed enough to accurately represent many of the relevant sound correspondences between languages.

Still, there is considerable room for improvement. In addition to extending the coverage of the described phenomena, the phonetic transcription pipeline could be streamlined a lot more by using full-blown finite state technology. While some work
in this direction will eventually be done within our project, we would be happy to collaborate with any researchers who work on grapheme-to-phoneme conversion for any Uralic language. Feedback about errors of and possible improvements to our transliterators is very welcome as well.

4 The Dataset

From the full set of 1.016 concepts across 20 languages, we have so far been able to retrieve some information for 97% of all concept-language pairs. For about 92%, we are reasonably confident that the extracted translations are correct. For 5% of the data, our translations are still classified as uncertain, the main causes being lack of precision in or insecurity about the interpretation of source entries. Figure 1 gives an overview of the current coverage for each language. Because some work is still being done, we expect a slightly smaller proportion of uncertain entries in the final release.

Current snapshots can be retrieved in different PDF formats from the author’s webpage³. The main file contains all the parallel translations and their transliterations in a one-concept-per-page format. Alternatively, single-language wordlists with glosses in German, English, and Russian are available as well. The purpose of these pre-release snapshots is to make it as easy as possible for specialists in the individual languages to inspect the data and to give us feedback on our unavoidably many errors.

Releases in various machine-readable formats will follow after some feedback was received. New versions will continually be released based on expert feedback or additional knowledge gained by the author. All materials will be published under an open license, allowing other researchers to build upon the database as they wish.

³http://www.sfs.uni-tuebingen.de/~jdellert/northeuralex

<table>
<thead>
<tr>
<th>Language</th>
<th>fin</th>
<th>krl</th>
<th>vep</th>
<th>ekk</th>
<th>liv</th>
<th>sma</th>
<th>sme</th>
<th>sms</th>
<th>sjd</th>
<th>mhr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certain</td>
<td>1015</td>
<td>976</td>
<td>914</td>
<td>1014</td>
<td>894</td>
<td>874</td>
<td>889</td>
<td>956</td>
<td>951</td>
<td>913</td>
</tr>
<tr>
<td>Uncertain</td>
<td>1</td>
<td>39</td>
<td>81</td>
<td>2</td>
<td>96</td>
<td>72</td>
<td>73</td>
<td>26</td>
<td>101</td>
<td></td>
</tr>
<tr>
<td>Missing</td>
<td>0</td>
<td>1</td>
<td>21</td>
<td>0</td>
<td>26</td>
<td>70</td>
<td>54</td>
<td>37</td>
<td>39</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Language</th>
<th>mdf</th>
<th>myv</th>
<th>udm</th>
<th>kpv</th>
<th>hun</th>
<th>mns</th>
<th>kca</th>
<th>sel</th>
<th>yrk</th>
<th>nio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certain</td>
<td>920</td>
<td>955</td>
<td>934</td>
<td>883</td>
<td>1015</td>
<td>941</td>
<td>938</td>
<td>877</td>
<td>945</td>
<td>888</td>
</tr>
<tr>
<td>Uncertain</td>
<td>64</td>
<td>53</td>
<td>72</td>
<td>119</td>
<td>1</td>
<td>52</td>
<td>45</td>
<td>56</td>
<td>50</td>
<td>47</td>
</tr>
<tr>
<td>Missing</td>
<td>32</td>
<td>8</td>
<td>10</td>
<td>14</td>
<td>0</td>
<td>23</td>
<td>33</td>
<td>83</td>
<td>21</td>
<td>81</td>
</tr>
</tbody>
</table>

Figure 1: Current coverage of the concept list (on December 15th, 2014).
5 Future Work

One of the long-term goals of the author is to further improve the coverage of the Uralic language family by collecting data on some additional languages. At least six languages are planned for future inclusion: Enets, Hill Mari, Lule Sami, Inari Sami, Ingrian, and Võro. Work on producing the datasets for major contact languages is already under way, with the current focus of data collection being on the languages of Siberia. The release of this part of the NorthEuraLex database is currently scheduled for the end of 2015, with the remaining parts following until the end of 2017.

To turn the database into an attractive benchmark for cognate detection, we are also starting to enhance the dataset by cognacy judgments. This subproject will be pursued further in collaboration with other researchers, based on written sources, or on existing databases such as the one developed by the Etymon project [6].

6 Conclusion

This paper describes the design decisions behind and the data collection process for a large lexicostatistical database that spans about a thousand concepts in twenty Uralic languages. The envisioned primary use case of the database is as a benchmark for different tasks in computational historical linguistics. Once it is enhanced by cognacy judgments, the database will become one of the largest available testsets for automated cognate detection. The unusually high coverage of the database will also allow markup of quite a few cross-semantic cognates, providing a first test case for advanced cognate detection methods that also attempt to model semantic change. By including variants of Proto-Uralic as reconstructed by Károly Rédei [7] or Pekka Sammallahti [8], the database could also become a very interesting test case for the challenging task of automated proto-language reconstruction (see e.g. [9]).

Beyond its relevance for the field of computational historical linguistics, an open and readily available lexical database is likely to increase the attractiveness of interdisciplinary work on Uralic languages, generating more interest in the field and hopefully leading to new discoveries.

Acknowledgments

This work has been supported by the ERC Advanced Grant 324246 'EVOLAEMP', which is gratefully acknowledged.
References


Can Morphological Analyzers Improve the Quality of Optical Character Recognition?

Miikka Silfverberg  
University of Helsinki  
Dept. of Modern Languages

Jack Rueter  
University of Helsinki  
Dept. of Modern Languages

mpsilfve@iki.fi  
jack.rueter@helsinki.fi

December 16, 2014

Abstract

Optical Character Recognition (OCR) can substantially improve the usability of digitized documents. Language modeling using word lists is known to improve OCR quality for English. For morphologically rich languages, however, even large word lists do not reach high coverage on unseen text. Morphological analyzers offer a more sophisticated approach, which is useful in many language processing applications. This paper investigates language modeling in the open-source OCR engine Tesseract using morphological analyzers. We present experiments on two Uralic languages Finnish and Erzya. According to our experiments, word lists may still be superior to morphological analyzers in OCR even for languages with rich morphology. Our error analysis indicates that morphological analyzers can cause a large amount of real word OCR errors.

1 Introduction

Digital media is an integral part of modern society. Thus digitization of printed matter is crucial for the viability of minority languages. It also serves the linguistic community by making printed media widely available. Simply scanning documents, however, is not enough because few applications can deal with images directly. Optical Character Recognition (OCR) can substantially improve the usability of digitized documents for example by allowing search engines to index them. In this paper, we
investigate improving the quality of OCR for languages with rich morphology, that is languages with extensive inflection, derivation and compounding.

OCR engines can benefit from language modeling, which is a field encompassing a variety of techniques that aim at improving the function of language processing applications by capturing key properties of the target language. For example, translation software and speech recognizers benefit greatly from sophisticated statistical language models. In OCR, however, simple language models such as word lists are commonly used.

Word lists are adequate in applications designed for languages with limited morphology such as English. Nevertheless, morphologically rich languages, including the Uralic languages, require more elaborate approaches. For these languages, even extensive word lists are unlikely to reach high coverage on previously unseen text [1]. In contrast to word lists, **morphological analyzers** [2], which encode the derivational and inflectional morphology of a language, can achieve substantially higher coverage. Thus it is conceivable that language models utilizing morphological analyzers could improve the quality of OCR for morphologically rich languages.

In this paper, we present experiments on OCR for two Uralic languages with rich morphology, Finnish and Erzya. We performed the experiments using the open-source OCR engine Tesseract [3] and open-source morphological analyzers for both languages. As baselines, we use both OCR systems without language modeling and systems using word lists.

In light of our experiments, it seems that morphological analyzers do help in OCR of morphologically rich languages compared to a baseline without language modeling. We were, however, unable to get improvements over using word lists harvested from the Wikipedia databases for Erzya and Finnish. This result is somewhat surprising, as the morphological analyzers have higher coverage on the test material than the word lists do. Error analysis revealed that the high coverage of the morphological analyzers may in fact present a problem for the OCR process, as it leads to a substantial number of real word errors.

Although we did not get improvements over word lists, it is worth pointing out that for some under-resourced minority languages morphological analyzers created by linguists represent the best readily available lexical resources in machine readable format. The reason for this is that digital content on the Internet can be scarce and the orthography of the material may be non-standard. Therefore, using a morphological analyzer as part of an OCR engine can still be motivated.

The paper is structured as follows. We describe related work in Section 2. In Section 3, we describe the Tesseract OCR engine, morphological analyzers and their integration. Section 4 details the experimental setup. In Section 5, we present the results of the experiments and a brief error analysis for the experiment on Finnish.
We conclude the paper in Section 6.

2 Related Work

Although, morphological analyzers have been used in OCR *post-processing*, this is, to our knowledge, the first investigation of utilizing morphological analyzers as language models *during* the OCR process. There are, however, other approaches to language modeling for OCR of morphologically rich languages, which have been investigated.

Smith et. al [4] add a module, which expands the vocabulary by generating additional word forms from stem suffix pairs. In contrast to our approach, their method requires no additional linguistic resources, since the sets of stems and affixes are harvested from word lists. We believe, however, that this approach is unlikely to work well with languages that have extensive compounding such as Finnish. Data sparseness will be a grave problem.

There is a large body of literature on spelling correction for morphologically rich languages, for example [5] and [6], and similar approaches have been successfully applied to OCR post-processing, for example [7, 8, 9]. In our work, we wanted to improve the language model instead of using post-processing to correct errors, because post-processing cannot in principle give as good results as improved language modeling. The reason is that knowledge about the reliability of predictions of the individual characters has already been lost before the post-processing stage.¹

Finally, character based statistical language models have been investigated, but the results of this approach are mixed [10]. It seems that statistical language models do improve performance when the baseline is low, but they may in fact degrade the performance of high accuracy OCR systems. Statistical language modeling, however, has given good results in the related field of handwritten text recognition [11], where the overall performance is much lower.

3 Methods

In this section, we describe the Tesseract OCR engine, the HFST finite-state library, HFST morphological analyzers, and the process of combining these utilities.

¹If this knowledge were available at the post-processing stage, post-processing could probably be used to the same effect as language modeling.
3.1 Tesseract

The Tesseract² OCR engine [3] was originally developed at HP Labs between 1984 and 1995 for high quality OCR of English. In 2005 it was released as an open-source project and has since been applied to several languages and alphabets, for example Finnish. Tesseract was therefore a natural starting point for exploring improvements for OCR of the Uralic languages.

The recognition process of Tesseract can be seen as a pipeline consisting of four stages [3]: (1) identification of character boundaries, (2) grouping of characters into words and lines, (3) word level recognition, and (4) resolution of ambiguous word spacing.

Our work focuses on the third stage of the pipeline, namely word level recognition. Word level recognition encompasses two sub-tasks: character recognition using a character classifier and word recognition using a combination of an additional word level classifier and various language models. During word level recognition, the word level classifier and language models give competing suggestions based on the output of the character recognizer. The highest scoring suggestion becomes the OCR output.

The existing language models in Tesseract are word lists, which are compiled into directed acyclic graphs (DAG) for fast processing. Tesseract incorporates a number of different language models³, for example: A short list of frequent word forms, a more extensive dictionary, punctuation patterns and a list of word forms containing digits.

Each language model and the adaptive classifier have associated weights which determine their relative importance. For example, the frequent word model has a greater weight than the dictionary model reflecting the higher prior for seeing frequent words.

When the character model returns a scored set of possible word forms, each of the language models and the word level classifier return the highest scoring word form known to the model. These suggestions are further re-scored using the model specific weights. Finally, the highest scoring suggestion is selected.

We modify this system by replacing the word lists with a morphological analyzer. The associated weight for the analyzer is the same as for the dictionary model in the original system.

3.2 Helsinki Finite-State Technology

Helsinki Finite-State Technology (HFST) [12] is an open-source C++ library and collection of tools for constructing finite-state transducers and morphological analyzers

⁴https://code.google.com/p/tesseract-ocr/
⁵https://code.google.com/p/tesseract-ocr/wiki/TrainingTesseract3

3.3 Can Morphological Analyzers Improve the Quality of Optical Character Recognition? [page 48 of 131]
based on finite-state technology. Morphological analyzers compatible with the HFST library exist for several languages. We know of at least fifteen Uralic languages with HFST morphological analyzers, for example Erzya and Finnish.

3.3 Morphological Analyzers as OCR Language Models

As mentioned above, Tesseract internally represents language models as directed acyclic graphs or DAGs. HFST morphological analyzers are finite-state transducers (FST), which are closely related to DAGs. The main difference is that finite-state transducers transform strings instead of simply accepting or discarding them. Additionally, finite-state transducers can be cyclic unlike DAGs. By modifying both Tesseract and the analyzers, we were able integrate morphological analyzers into Tesseract.

There exists a straightforward conversion (projection) from FSTs to finite-state automata, which are identical to DAGs in other respects, but may be cyclic like FSTs.

It turned out, that Tesseract is not in principle incompatible with cyclic graphs. The existing implementation simply did not offer a way to produce cyclic graphs. Fortunately, it was not difficult to implement a sub-class for the Tesseract language model class, Dawg, which does support cyclic graphs. Additionally, we implemented a driver for HFST automata in optimized lookup format, which supports lookup speeds of up to 100 000 words/s [13].

HFST morphological analyzers can contain so called flag diacritics [14], which are used to compress the finite-state machine by introducing non-determinism in a controlled way. Tesseract employs a search algorithm for finding word suggestions that requires that the language model be deterministic. Hence, it cannot handle flag diacritics. Fortunately, HFST includes utilities which can be used to eliminate flag diacritics from a finite-state machine without changing its behavior.

All the necessary steps to transform an HFST morphological analyzer into a Tesseract language model have been incorporated into the HFST interface as the tool hfst-fst2tesseract.

4 Experiments

In this section we describe the Finnish and Erzya data sets used in the experiments, the evaluation procedure and the experiments.

4http://giellatekno.uit.no/all-lang.eng.html
4.1 Data

We evaluate the impact of morphological analyzers in OCR for two Uralic languages, Erzya and Finnish. The Erzya language has a relatively rich morphological system of regular inflection, most extensive in the verbs and nouns. The verbs attest to object and subject conjugation in 7 moods, whereas there are 9 declensions for 9-15 regular case forms in nouns, with additional conjugation possibilities in two tenses for 3-6 of those. Erzya is written using the Cyrillic alphabet. Finnish is similar to Erzya in that it has a extensive noun and verb inflection. Additionally, Finnish has a productive compounding mechanism, which gives rise to an extensive vocabulary. Unlike Erzya, Finnish is written using the Latin alphabet.

We perform experiments on excerpts from novels. For Finnish, we use pages 5 - 21 of the novel Elokuu (August) by F.E. Sillanpää [15] (3219 tokens, 24096 characters) and for Erzya, we use pages 3 - 21 from the translation of the, originally Russian, novel Ава (Mother) by Maksim Gorky [16] (4539 tokens, 58548 characters). In order to estimate the effect of different language models on scanned material of varying quality, the data were scanned in different resolutions: 100, 200 and 300 dpi.

Even without language modeling, Tesseract performs quite well on scanned images of quality 300 dpi. The result requires relatively little manual correction. In contrast, 100 dpi images usually result in quite poor performance. In fact, manual correction may take longer than simply writing the text from scratch.

4.2 Resources

For constructing Tesseract systems with word lists as language models, we used the XML dumps of the Erzya\(^5\) and Finnish\(^6\) Wikipedias. We used the utility wp2text \(^7\) for extracting the text contents from the XML files.

We formed lists containing the N most frequent word forms for various N in the range 1000 up to 1 million for Finnish and 1000 up to 68 000 for Erzya (there were no more unique word forms in the Erzya Wikipedia).

In addition to Wikipedia text, we used freely available morphological analyzers for Finnish and Erzya. OMorFi \(^8\) is a broad coverage Finnish morphological analyzer available online\(^8\). For Erzya, we used the Erzya analyzer distributed by the Giellatekno project \([\text{18}]\).

\(^6\)ftp://wikipedia.c3sl.ufpr.br/wikipedia/fiwiki/20141018/fiwiki-20141018-pages-meta-current.xml.bz2
\(^7\)https://github.com/yohasebe/wp2text
\(^8\)https://code.google.com/p/omorfii/
The coverages of different linguistic resources on test data are shown in Table 1. For both languages, the coverage is best using the morphological analyzer. However, for Finnish, the coverage of the one million word list comes very close.

<table>
<thead>
<tr>
<th>Erzya</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1K word list</td>
<td>28.0%</td>
</tr>
<tr>
<td>10K word list</td>
<td>49.5%</td>
</tr>
<tr>
<td>68K word list</td>
<td>58.6%</td>
</tr>
<tr>
<td>Morphological analyser</td>
<td>80.6%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Finnish</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1K word list</td>
<td>32.2%</td>
</tr>
<tr>
<td>10K word list</td>
<td>52.8%</td>
</tr>
<tr>
<td>100K word list</td>
<td>71.4%</td>
</tr>
<tr>
<td>1000K word list</td>
<td>84.5%</td>
</tr>
<tr>
<td>Morphological analyser</td>
<td>86.7%</td>
</tr>
</tbody>
</table>

Table 1: Coverages of linguistic resources on the text tokens of the Erzya and Finnish test material.

4.3 Experiments

We trained five different OCR systems for Finnish and four systems for Erzya:

- A system without a language model (the baseline).
- Systems using 1000 and 10 000 word vocabularies both for Finnish and Erzya, a 68 000 word system for Erzya and 100 000 and 1 million word systems for Finnish.
- A system using a morphological analyzer as language model.

For Finnish, we constructed the baseline system simply by deleting the vocabularies (freq-dawg and word-dawg) from the existing Tesseract OCR system for Finnish. For Erzya, we trained our own baseline system.

In order to compile systems with vocabularies ranging from 1000 words to 1 million words, we extracted the most common N words from the Wikipedia, compiled them into a directed acyclic graph using the Tesseract utility wordlist2dawg and used the graphs as word models (word-dawg) in a system which otherwise was identical to the baseline system.

The morphological analyzers, were first processed using the HFST utility hfst-fst2tesseract. We then combined them with a system identical to the baseline systems.

---

*see: https://code.google.com/p/tesseract-ocr/downloads/list
4.4 Evaluation

It is tempting to view OCR as a special case of sequence labeling, since an OCR engine essentially labels the characters in a digitized text using alphabetical symbols. This suggest evaluation based on character error rate in relation to a gold standard text.

Unfortunately, simple metrics such as character error rate cannot be used, since OCR frequently changes the length of the underlying text, because spurious characters may be inserted and characters in the text may be deleted. Therefore, we evaluated by measuring the edit distance [19] of the OCR result and the gold standard.

In practice, we first aligned the texts on character level using the Unix utility diff. We then computed the number of edits required to transform the OCR result into the gold standard text. We call this figure the edit count (EC). For each experiment, we report both raw edit counts and the reduction in edit count (ER) compared to the baseline OCR system without language modeling.

The edit reduction, ER, from a baseline $B$ to an improved edit count $C$ is

$$ER = \frac{B - C}{B}$$

If, the baseline $B$ is in fact better than the count $C$, ER will be negative.

We divided the test material into pages, and performed paired one sided Wilcoxon tests to assess the statistical significance of our results with confidence level 95%. We compared all systems to the baseline model. We additionally compared the best word-list system to the system using a morphological analyzer.

5 Results

In this section we show the results for Finnish in Table 2 and for Erzya in Table 3.

For the Finnish novel, all systems utilizing some kind of language modeling fared better than the baseline system without any kind of vocabulary information. The morphological analyzer performed better than the other systems on the lowest image quality 100 dpi. Otherwise, it in fact performed worse than the other systems utilizing language modeling.

For resolutions 300 and 200 dpi, all language models gave statistically significant improvements over the baseline in the 95% confidence interval. The best word list system was better than the morphological analyzer. For 100 dpi, only the morphological analyzer performed significantly better than the baseline, but not significantly better than the best word list model.

The results for Erzya paralleled those of Finnish. The morphological analyzer improves over the word lists only for the lowest resolution 100 dpi. For resolution 200
Table 2: ER (and edit counts) for the Finnish novel Elokuu using different systems and resolutions.

<table>
<thead>
<tr>
<th></th>
<th>300 dpi</th>
<th>200 dpi</th>
<th>100 dpi</th>
</tr>
</thead>
<tbody>
<tr>
<td>No language model</td>
<td>0.0% (794)</td>
<td>0.0% (1265)</td>
<td>0.0% (15504)</td>
</tr>
<tr>
<td>1000 words</td>
<td>32.1% (539)</td>
<td>36.8% (799)</td>
<td>2.1% (15172)</td>
</tr>
<tr>
<td>10 000 words</td>
<td>35.3% (514)</td>
<td>44.7% (699)</td>
<td>4.0% (14891)</td>
</tr>
<tr>
<td>100 000 words</td>
<td>31.5% (544)</td>
<td>44.0% (708)</td>
<td>3.2% (15014)</td>
</tr>
<tr>
<td>1 million words</td>
<td>33.5% (528)</td>
<td>45.4% (691)</td>
<td>2.4% (15131)</td>
</tr>
<tr>
<td>Morph. analyzer</td>
<td>25.3% (593)</td>
<td>30.0% (885)</td>
<td>5.7% (14621)</td>
</tr>
</tbody>
</table>

dpi, the morphological analyzer does not seem to have any effect. For 200 and 300 dpi, the morphological analyzer was significantly worse than the best word list model. For 100 dpi, the model with 1000 word vocabulary was significantly worse than the baseline, but the other results were not statistically significant.

Table 3: ER (and edit counts) for the Erzya novel Ава using different systems and resolutions.

<table>
<thead>
<tr>
<th></th>
<th>300 dpi</th>
<th>200 dpi</th>
<th>100 dpi</th>
</tr>
</thead>
<tbody>
<tr>
<td>No language model</td>
<td>0.0% (3257)</td>
<td>0.0% (3224)</td>
<td>0.0% (15788)</td>
</tr>
<tr>
<td>1000 words</td>
<td>20.9% (2576)</td>
<td>11.7% (2846)</td>
<td>-10.7% (17473)</td>
</tr>
<tr>
<td>10 000 words</td>
<td>29.5% (2295)</td>
<td>22.7% (2492)</td>
<td>1.8% (15498)</td>
</tr>
<tr>
<td>68K words</td>
<td>30.9% (2249)</td>
<td>21.9% (2517)</td>
<td>0.5% (15702)</td>
</tr>
<tr>
<td>Morph. analyzer</td>
<td>8.0% (2996)</td>
<td>-0.1% (3230)</td>
<td>2.8% (15353)</td>
</tr>
</tbody>
</table>

5.1 Error Analysis

We examined the errors of the Finnish OCR system using a morphological analyzer and the best performing word list system for the highest image quality 300 dpi.

We classified errors into two types: real word errors and others. Real word errors are errors, where the resulting incorrect word is known by the language model, for example a genitive of ’his/her’ “Hänen” was recognized erroneously as the genitive of ’pike’ “Hauen”. Other errors simply encompass all other error types, common examples include insertion and deletion of punctuation and casing errors such as lower case “v” being recognized as an upper case “V” and vice versa.
A total of 18% of the errors produced by the morphological analyzer were real word errors. In contrast the word list only gave 2% real word errors.

6 Discussion and Conclusions

In light of our experiments, it seems that morphological analyzers may do more harm than good in OCR. For higher resolutions, 200 and 300 dpi, the morphological analyzers fared worse than even the smallest vocabulary of 1000 words. This happens both for Finnish and Erzya. We believe, the large amount of real word errors is to blame. However, for the lowest image quality 100 dpi, the morphological analyzers do improve performance. It is interesting to compare these results to statistical language modeling for OCR, which also improves results when performance is low, but can degrade it otherwise [10].

Interestingly, vocabulary size does not seem to be a very good predictor of performance. For Finnish, the 10 000 word vocabulary performs best on the 300 dpi material with. Similarly, the system with 10 000 word vocabulary performs best for Erzya material in 200 dpi resolution. Overall, the results for all but the smallest vocabularies lie very close to each other. All OCR results are better for Finnish, which probably reflects the quality of the baseline models.

It would seem that the effect of language modeling is already exhausted at 10 000 words. Therefore, it is not horribly surprising that the morphological analyzer does not achieve better results than the systems using word lists. The fact that its performance is so low, however, was mildly surprising.

In order to limit the number of real word errors, we tried excluding all compounds that had not been attested in real text from the morphological analyzers. Unfortunately, this did not improve the results.

It might be worth while trying to include word frequency information into the language model. However, this remains future work, as it would require extensive changes to Tesseract.

Acknowledgments

We, the authors, want to thank Jussi-Pekka Hakkarainen, Timo Honkela and Kristter Lindén for helpful ideas and Tommi Pirinen for creating OMorFi. Additionally, we want to thank the Kone Foundation for funding. Finally, all three anonymous reviewers made useful suggestions. We are in your debt!

¹⁰This may be a consequence of Tesseract’s approach to language modeling described in Section 3.1.
References


Corpus.mari-language.com: A Rudimentary Corpus Searchable by Syntactic and Morphological Patterns

Jeremy Bradley
Ludwig Maximilian University of Munich
Institute for Finno-Ugric and Uralic Studies

&

Koneen Säätiö
J.Bradley@lmu.de

December 16, 2014

Abstract

This paper introduces a rudimentary infrastructure for a searchable corpus of Mari, a highly agglutinative Uralic language spoken in the Volga and Ural regions of the Russian Federation. This infrastructure allows users to search the corpus by syntactic and morphological patterns. It makes use of the University of Vienna’s digital Mari-English dictionary, published under a Creative Commons License in 2014, and a morphological analyser following a simple item-and-arrangement approach. Texts fed into the corpus are subjected to a morphological analysis, the results of which are saved into the application’s database with the corpus materials and are accessed by the search algorithm. A demonstration of this open-source tool, covering 994,097 tokens taken from works not subject to copyright, can be found at corpus.mari-language.com, the source code at source.mari-language.com. While a non-representative text collection of this scope can only serve demonstrative purposes, the infrastructure could enable quantitative diachronic or sociolinguistic comparisons, if fed with a sufficiently wide text collection annotated with adequate metadata.

This work is licensed under a Creative Commons Attribution–NoDerivatives 4.0 International Licence. Licence details: creativecommons.org/licenses/by-nd/4.0/
1 Introduction: Structure of the Paper

Section 2 introduces the language (or languages) dealt with in the paper. Section 3 gives a brief overview of the language data available and obtainable to those interested in a corpus-linguistic approach towards Mari. Section 4 introduces the manner in which data is stored and manipulated in the corpus (a demonstration of which can be found at corpus.mari-language.com), Section 5 briefly explains how data of this kind can be searched in non-trivial manners. Finally, Section 6 outlines the technical framework upon which this tool is based. Given practical constraints, only a rough overview of these tools can be given. However, the source code can be accessed in its entirety by anyone who is interested (source.mari-language.com) and extensive documentation is in preparation.

2 What is Mari, who are the Maris?

The Mari language, referred to by the exonym Cheremis in older sources, is a highly agglutinative () Uralic language native to the Volga and Ural Regions of the Russian Federation. It shares official status with Russian in the Mari El Republic, a subject of the Russian Federation slightly smaller in area than Macedonia that is located some 500–800 kilometres east of Moscow, near the confluence of the Volga and Kama rivers. It is a pluricentric language with two distinct literary norms, the dominant Meadow Mari and the critically endangered Hill Mari. In the 2010 All-Russia population census [1], 365,127 people claimed to be Mari speakers, and 23,062 of them identified themselves as speakers of Hill Mari. Linguists generally divide Mari into four dialect groups: Meadow Mari, Hill Mari, Northwestern Mari, Eastern Mari [2, p. 15]. The two aforementioned literary norms are based on the dialect groups of the same name. Speakers of dialects belonging to the other two groups use the Hill Mari and Meadow Mari literary norms in writing. The UNESCO Atlas of the World’s Languages in Danger [3] classifies Meadow Mari as “definitely endangered” and Hill Mari as “severely endangered”.

Mari uses a variant of the Cyrillic alphabet slightly different from the Russian alphabet, and Mari data is stored using the Cyrillic alphabet in the corpus infrastructure at hand. Using the Vienna project’s transcription and transliteration toolkit (found at transcribe.mari-language.com [4]), however, on-the-fly transcriptions into UPA and IPA are possible. All examples used in this paper are given in IPA transcription.
3 How much data do/can we have?

In spite of the socio-political hardships facing the Mari language and indigenous languages in Russia in general, sufficient amounts of language data are obtainable to make Mari attractive for serious corpus-linguistic research. Both literary norms are still comparatively widely used: Novels, daily newspapers, magazines, textbooks and scientific theses are still published today in both language norms, in Russia and abroad (thanks in large part to funding from the Finnish M.A. Castrén Society [“M. A. Castrénin seura”] and the Estonian Kindred Peoples’ Programme [“Hõimurahvaste Programm”]). A corpus containing millions of tokens of modern-language texts written by native speakers of different speech variants would be viable in principle, were it not for practical and legal constraints. It would not be possible to adhere to guidelines followed when creating corpora on large European languages, because, for example, texts on medicine (as the Oxford Guide to Practical Lexicography [5, p. 222] suggests as a building block of a corpus) simply do not exist in Mari. But a representative corpus covering actual usage domains of literary Mari appears to be a valid goal.

Historical texts are available as well. Mari literacy traces its roots back to the first grammar of Mari, published in Saint Petersburg in 1775 and widely accessible today thanks to the publication of an extensively commented facsimile edition in 1956 [6]. It did not, however, take off in a serious fashion until the 20th century. The Mari elementary school teacher Timofey Yevseyev [Тимофей Евсеев] (1887-1937) provided the Helsinki-based Finno-Ugrian Society [“Suomalais-Ugrilainen Seura”] with a wide range of Mari-language texts between 1908 and 1929; these have since been published with German translations [7]. The Hungarian linguist Ödön Beke was able to collect a large body of texts working with Mari-speaking prisoners of war during the First World War [8]. More recently, a substantial body of Mari-language newspapers and textbooks from the early twentieth century, covering a wide geographic range, has been digitized, and made available on the National Library of Finland’s website [9].

If these historical materials were integrated into a joint infrastructure with texts in modern Mari, a wide range of analyses would become possible: diachronic (Hill Mari today vs. Hill Mari around 1920), dialectological (Mari in Mari El around 1920 vs. Mari in Bashkortostan around 1920), genre-based (newspapers vs. schoolbooks), sociolinguistic (articles written by men vs. articles written by women), etc. As this is currently beyond the scope of my capacities, I will restrict myself here to presenting the infrastructure that would make such an analysis possible, if it was fed with the correct texts. For the time being, a non-representative body of texts was fed into the infrastructure. This is discussed in Section 6.3.
Semi-automatic annotation of texts

For a corpus to be maximally useful, it has to be searchable in non-trivial manners. The amount of annotation needed to make this possible differs greatly from language to language. Corpora of morphology-poor English, for example, rely heavily on part-of-speech tagging, where individual words of English strings are classified by their word class: “The house is on fire.” could be tagged as “The[article] house[noun] is[verb] on[preposition] fire[noun].” For English, this annotation already suffices to allow users to search for a wide range of grammatical structures. For example, a linguist researching the proliferation of the split infinitive (“to boldly go”) could simply search for the lexeme “to”, followed by an adverb, followed by a verb, to uncover examples of the structure of interest. In languages where words have more internal structure, however, a morphological analysis is indispensable.

Using the mechanism detailed below, the resources in this demonstration infrastructure were run through an automated morphological analyser when imported into the infrastructure. The result of the analysis of a simple string, a rudimentary interlinearization following the Leipzig Glossing Rules [10] as best possible, can be seen in Figure 1.

![Figure 1: A glossed sentence, not disambiguated](image)

4 Semi-automatic annotation of texts

In a word, *(it's)* a beautiful winter day. [edit]
The morphological analysis is not deterministic, i.e. is no morphological disambiguation is performed. When morphological or lexical ambiguity is encountered and several interpretations of a word are found, the analyser yields and saves all possible interpretations. No attempts to cut down on ambiguity using, for example, collocation data or syntactic models have been made so far. Thus, when the analyser encounters the Mari word form kefće it can either be an uninflected noun meaning “sun” or “day”, or the imperative (second person singular) or a connegative form of a verb meaning “to hang”. Authorized users (the screenshots in this paper show the interface as seen by an authorized user) can pick the correct glossing by pushing the button titled “[pick]” beside the correct gloss. Their choice is then stored in the database; any subsequent users who encounter this string will see the disambiguated glossing seen in Figure 2. Authorized users can reset the glossing by pressing the button titled “[(re)analyse sentence]”.

Note that the disambiguation of morphological or lexical ambiguity does not disambiguate polysemy. The noun kefće has several aspects of meaning, the most important of which are “sun” and “day”. The analyser puts the very first translation given in the lexical base as a glossing for the stem by default. If users move their mouse over the English gloss, all translations contained in the lexical base show up as a tooltip, as seen in Figure 3. Users can alter the gloss of individual words by clicking the button titled “[edit]” beside a gloss.

Resources fed into the corpus can be sanitized manually as shown here, or they can be left in the corpus in a raw form. Obviously, sanitized data is preferable, but sanitizing data in this manner is time-consuming. Especially if an infrastructure of this sort was realized as a monitor corpus automatically updated on a regular basis, it
might not be realistic for a human user to disambiguate all the glosses in this manner. However, this data sanitization is not a requirement for the strings to be searchable by syntactic or morphological patterns. The presence of multiple interpretations of a string simply means that the false positive rate of search algorithms will be somewhat higher: grammatical structures will be found in places where they are not present. The false negative rate is not affected: where grammatical structures do occur, the search algorithms do not miss them.

5 Searching the corpus

Users can browse the text collection, or search for specific grammatical constructions within it. The “[Search]”-button on the main page of the corpus infrastructure (corpus.mari-language.com) and on pages of indidividual resources¹ and chapters². Depending on where the button is pressed, the entire inventory or one specific resource is searched.

The interlinearization created by the morphological analyser has several layers or tiers. Table 1 gives another example of an interlinearization, with the layers marked. Users can search for full or partial matches on all layers. For example, they can search for all occurrences of the base (lemma) form ɤʃte “to do”, and still find these string even if the allomorph ɤʃt occurs in this particular example.

To search for more complicated structures, users can specify additional features that must - or may not - co-occur with the first search specified. It must be specified where addition features must - or may not - occur: in the same word, in the next word, in the previous word, later in the sentence, earlier in the sentence, anywhere in the sentence.

Some examples of possible queries, and the structures they would return:

- “base form” “equals” “ida” - “next word” - “gloss” “equals” “-CNG” “negated’ (see Figure 4): This input would search for occurrences of the word “ida”, followed by anything but the so-called connegative form. Mari uses a negation

---

¹e.g. www.univie.ac.at/maridict/site-2014/corp_chapters.php?book=1
²e.g. www.univie.ac.at/maridict/site-2014/corp_content.php?book=1&chapter=16
<table>
<thead>
<tr>
<th>(string)</th>
<th>tudo mom ɤʃta?</th>
</tr>
</thead>
<tbody>
<tr>
<td>unglossed</td>
<td>tudo mom ɤʃta?</td>
</tr>
<tr>
<td>morpheme</td>
<td>tudo mo -m ɤʃt -a</td>
</tr>
<tr>
<td>base form</td>
<td>tudo mo -m ɤʃte -a</td>
</tr>
<tr>
<td>gloss</td>
<td>(s)he what -ACC do -3SG</td>
</tr>
<tr>
<td>part of speech</td>
<td>pr pr -case vb2 -pers</td>
</tr>
<tr>
<td>(free translation)</td>
<td>What does (s)he do?</td>
</tr>
</tbody>
</table>

Table 1: The layers of interlinearization, with an example

![Search interface](http://dx.doi.org/10.7557/5.3467)

Figure 4: The search interface with a sample query
verb [11, p. 115] typically followed by the connegative form, in the same manner that Finnish does. The form “ida” is the second person plural imperative of the connegative verb, and this query would find all occurrences of it where it is atypically not followed by the negation verb, but rather.

- “gloss” “equals” “-PTCP.FUT” - “next word” - “part of speech” “equals” “po”:
  This query would return postpositional constructions using the future future participle, which are quite rare compared to postpositional constructions using the passive participle.

Users can use the morphological analyser at morph.mari-language.com to determine how exactly the structure they are interested in is glossed by the software. A complete overview of the suffixes processed by the analyser will be included in the documentation.

6 The architecture behind the infrastructure

Three fundamental building blocks were necessary for the creation of this demonstration: a lexical base, a morphological analyser, and a text collection.

6.1 The lexicon

The Mari-English Dictionary [12] created by my project team at the University of Vienna is one of the ingredients needed for this resource. It currently includes 42,560 lexemes. The entries covering these are saved in a systematic format (XML) and are annotated as needed by the analyser - the word class of all lemmas is defined, etc.

6.2 The morphological analyser

The morphological analyser is based on a morphological analyser of Mari I wrote using Java several years ago [13]. Due to repeated difficulties with Java related to security updates, and more general problems resulting from having such a program operate on the client side (i.e., the user’s computer), I recently reimplemented the same infrastructure using server-side PHP. The source code can be found in its entirety at source.mari-language.com. Individual strings can be interlinearized at analyser.mari-language.com.

A detailed overview of the workings of the analyser will be included in its documentation, which is currently still work in progress. Roughly speaking, the analyser
Table 2: Excerpt of the analyser’s inflectional morpheme inventory

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Gloss</th>
<th>PoS</th>
<th>Type</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>f</td>
<td>X</td>
<td>«case-g2»</td>
<td>LAT</td>
<td>case</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>na</td>
<td>N</td>
<td>«poss»</td>
<td>1PL</td>
<td>poss</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

follows a naïve item-and-arrangement architecture. The analyser has access to an inventory of inflectional suffixes. Three illustrative entries on this list - which contains over a hundred entries in its entirety - can be seen in table 2.

The field **Suffix** indicates the suffix itself, the fields **Gloss** and **PoS** contain the glosses used for the morphemes in question in the gloss and part of speech layers respectively (see Table 1).

The field **Type** indicates how a suffix is connected to a stem. The value *E* indicates that this suffix is in some cases preceded by an epenthetic *e*, the value *N* indicates that the suffix is not preceded by an epenthetic vowel. Every suffix is assigned to one type and all suffixes of a type behave in the same way morphologically. The analyser has a separate extraction mechanism for every suffix type that it uses to derive possible base forms when extracting a potential suffix.

The field **Class** assigns every suffix to a grouping. There are complex constraints governing suffix arrangement in Mari [11, p. 75]. For example, possessive suffixes (the class «poss») follow locative, illative, lative, and inessive case suffixes (the class «case-g2»), but precede the genitive, accusative, and comitative case suffixes (the class «case-g1»), whereas both arrangements are theoretically possible with the dative and comparative case suffixes (the class «case-g3»). The frequency of different suffix arrangements has been studied extensively [14], but as the morphological analyser is intended to be possibilistic, not probabilistic, all hypothetically possible arrangements were allowed.

The analyser was equipped with a list of possible suffix arrangements, an excerpt of which can be seen in Figure 5. Every arrangement shows which suffix classes can be connected to which stems (*n* for nominal stems, *v* for verbal stems, etc.) in which order. Suffix classes given in «guillemets» can occur optionally; suffix classes in {braces} must occur exactly once. (The class *tmp* represents tense/mode suffixes; one suffix of this type must occur in a finite verbal form.) The analyser only accepts interpretations of words that are compatible with one or more of the valid arrangements known to the computer.

The morphological analyser is, moreover, capable of extracting productive deriva-
n + «comp»«gen»«poss»«plur»«case-g1»«p3»«enc»
n + «comp»«gen»«plur»«case-g2»«poss»«p3»«enc»
...
v + {tmp}&«comp»«p3»«enc»
...

Figure 5: Valid suffix arrangements

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Date</th>
<th>Tokens</th>
<th>Eng. Trans.?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oŋaj marij jɤlme [16]</td>
<td>textbook</td>
<td>2010</td>
<td>2,508</td>
<td>yes</td>
</tr>
<tr>
<td>Elnet [17]</td>
<td>novel</td>
<td>1937</td>
<td>63,918</td>
<td>no</td>
</tr>
<tr>
<td>Sum</td>
<td>-</td>
<td>-</td>
<td>994,097</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3: Contents of illustrative corpus

...tional suffixes from words. Mari morphology generally adheres to the universal principle that derivational suffixes are closer to the base than inflectional suffixes [15, p. 95], though it is possible for plural suffixes to precede derivational suffixes: *verlase “local” < ver “place” + -la “-PL” + -se “-ADJ”*. When looking up prospective stems in the lexicon after the inflectional morphology has been extracted, the analyser also attempts to extract any of a number of productive derivational suffixes from the stem that produce words of a valid part of speech (e.g. nominal derivational suffixes when looking up a word that according to the arrangement patterns must be a nominal). Note that for practical reasons, participles are treated as deverbal nominal derivational suffixes by the analyser.

6.3 The texts

Table 3 shows the range of texts included in the demonstration, with some basic data. The New Testament suggested itself as an open-source English counterpart to freely available Mari strings and was included here as well. The content of my work group’s textbook *Oŋaj marij jɤlme* [16] was sanitized, but the contents of the other resources were not.
Acknowledgments

I would like to thank Paul Trilsbeek of the Max Planck Institute for Psycholinguistics for providing me with a digital version of the New Testament for non-commercial purposes. Thanks are due to the Kone Foundation for funding the research project that is giving me the opportunity to work on these resources (“The Mari Web Project: Phase 2.”). Furthermore, I owe gratitude to my former employer, the University of Vienna, for continuing to host my work even after I left its active duty roster. Finally, I am thankful to Timothy Riese and Elaine Bradley for proofreading this paper, and to Nele Lond for help related to hardware matters.

References


Infinite Monkeys of Babel – Crowdsourcing for the betterment of OCR language material

Wouter Van Hemel
National Library of Finland
Library Network Services
PL 26 (Teollisuuskatu 23)
FI-00014 University of Helsinki
wouter.van.hemel@helsinki.fi

Jussi-Pekka Hakkarainen
National Library of Finland
Research Library
P.O. Box 15 (Unioninkatu 36)
FI-00014 University of Helsinki
jussi-pekka.hakkarainen@helsinki.fi

November 7, 2014

Abstract

The OCR editor is the National Library of Finland’s most recent foray into the budding phenomenon of crowd-sourcing. Under the motto of many hands make light work, users can swiftly correct the typical mistakes in OCR scanned text of source materials – often of challenging visual quality – using nothing more than their browser. Improving the quality and availability of the digital text would make it easier to directly study the original sources, and indirectly contribute to other tools depending on accuracy such as word list generators and dictionaries.

This work is licensed under a Creative Commons Attribution–NoDerivatives 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by-nd/4.0/
1 Introduction

The OCR editor project of the National Library of Finland hopes to be a useful tool to improve the quality of OCR digitisation and thus to bring works of diverse physical quality and sources into the digital age. The editor is an interactive web application, enabling many people to contribute simultaneously and revise the OCR text of source materials in the system.

The project has multiple goals:

- make sure the transfer of source material into the digital age does not in any way take away any of the quality of the original;
- make it easier to study the material by availability and dissemination (i.e. internet); “editor as reader” or distributor;
- make automated corpora or word lists to improve the editor itself and other tools.

2 Software architecture

Let’s start with a brief technical overview of the tool’s architecture to put any technical terms in perspective.

The software consists of two major components: a front-end and a back-end.

The front-end – or editor itself – is a JavaScript application that maps ALTO XML attributes such as language and word coordinates to words in a web page. Through its two-pane graphical interface, users can correct wrongly recognised words side by side with the actual scanned imagery, or improve meta-attributes such as the language of individual words.

The more visual editor part is supported by a server back-end application in Python. This component serves the actual data to the editor and handles revisions, users, collection listing and other tasks. It also takes care of ”administrative” duties such as locking documents whose editing process has been considered finished so they can be exported – either as text or XML.

The more technically inclined readers can now feast their eyes on the graphical representation of the editor’s technical architecture to get a broad overview of its inner workings.
3 Process overview

How do we get from faulty OCR text to a usable corpus?

- The editor’s team receives – or produces themselves – ALTO XML from scanned images.
- These XML ALTO files and images are imported into the editor
- The text of the works is edited
- Optionally, corrections are checked by an authority
- The resulting text can be exported as XML or plain text, or can be used to create word lists and other derivatives

3.1 Manual labour

Manual labour here refers to the actual making of corrections in the online editor, marked in the figure by ❶.

Users log in to the editor and after selecting the desired document are greeted by a two-paned page. On the left side a high-resolution image of the original work is shown; on the right-hand side the OCR text. Hidden behind the words in the editor...
are the ALTO XML metadata attributes such as coordinates and language. On the top of the screen users have access to a toolbar that enables viewer functionality such as zoom and panel positioning. On the bottom an optional virtual keyboard layout can be selected to aid in the input of characters that might not be present on the physical keyboard. After the user has edited the text, they can save the version which then writes a revised ALTO XML file for the relevant pages and updates system metadata.

The editor aspect is already well developed and has been tested rather successfully in classroom settings – such as literature classes of Tampere university. The students mentioned the interactive component of working together and seeing immediate results as a more pleasant experience than individually working with paper copies. Experts benefit from the ability to off-load some of the burden and having to spend less time merging multiple contributed improvements manually.

### 3.2 Automated correction

This facet of the editor, marked by ❷, is still under development. Automatic correction is challenging because of the lack of support (stemmers, dictionaries) for a lot of the languages. Also, there is an inherent danger in automated mass-correction: changes ought to be reviewed by somebody intimately familiar with the language in question to avoid doing more harm than good. Faulty corrections could form a risk for corpora and word lists, especially because of the statistical relevance in languages with a small amount of source material.

For now the project’s aim is to let researchers identify the n most common OCR mistakes and only replace those literal words specifically listed as problematic. The actual corrections happen as a batch process in the back-end only after being approved by a researcher in the site’s interface. Replacing the n most commonly wrongly recognised OCR mistakes would in many cases lower the amount of often tediously repetitive manual labour significantly.

Other approaches are under consideration; this area is open for experimentation, as we can empirically determine which approaches are feasible and which ones do more than good. We hope to receive more input from researchers.
4 Crowd-sourcing and policy

Another challenge the editor tries – or needs – to solve is one of policy. When we think of a single researcher, located in their proverbial dungeon with nothing more than the soothing glow of their computer monitor, the answer is clear. But not so in an interactive multi-user environment. Who can edit texts? Does the corpus benefit from a fully open, wikipedia-style approach where anyone can make changes – and hence affect the outcome of corpus quality and by-products such as word lists? Or should the editing be limited to students of relevant languages? Perhaps only experts should be allowed to make direct changes to the corpus?

While the editor does not attempt to formulate the actual policy a collection or language community should apply – it is a mere technical tool – the granularity of access rights and restrictions can configured for collections according to the wishes of administrators of the system’s collections. It is possible and advised that collection ”owners” set their own policies and guidelines for their respective collections. The OCRUI editor offers a hierarchical permission system that allows designated ”administrators” to manage the users and permissions of the collections they hold the sceptre of. These users and permissions then apply for all documents and collections below that point in the tree.

Documents can also be ‘locked’ by administrators, disabling further editing. This makes it possible to export a completely corrected (or at least “known-good”) version for use in other tools.

5 Conclusion

This concludes our brief overview of some of the technical aspects and design of the National Library’s OCR editor. With this software, the team hopes to make a minor contribution to the field of linguistics and the preservation of Uralic languages – especially those endangered in our ever shrinking world.

To quote Frank Zappa: “Writing about music is like dancing about architecture”. If you are interested in the software or if this paper left you wanting, or perhaps the actual experience of using the editor would help to get a better overview of its use, its strengths and weaknesses – don’t hesitate to contact the team for more information.

http://ocrui.lib.helsinki.fi/
Acknowledgments

Acknowledgement – in alphabetical order – goes to the following people connected with the project, either currently or in the past: Jussi-Pekka Hakkarainen, Esa-Pekka Keskitalo, Victoria Kurkina, Anis Moubari and Juho Vuori.
Multilingual Semantic MediaWiki for Finno-Ugric dictionaries

Niklas Laxström
University of Helsinki
Department of Modern Languages
niklas.laxstrom@helsinki.fi

Antti Kanner
University of Helsinki
Department of Finnish, Finno-Ugrian and Scandinavian Studies
antti.kanner@helsinki.fi

December 16, 2014

Abstract

This paper introduces the concept of Multilingual Semantic MediaWiki, which can be used to build collaborative on-line projects for certain types of multilingual content. Namely, dictionaries whose users are multilingual or have different native languages. We describe two multilingual on-line dictionary projects built using the Multilingual Semantic MediaWiki framework. These projects cover Finnish, Swedish, the Sámi languages, Estonian and Ludic among others. We describe the benefits of using semi-structured data and the limitations of this particular semantic software based on the case study offered by the aforementioned projects. We evaluate these projects in terms of development and maintenance effort, number of visitors and contributors. We conclude that this is a low cost approach to increase openness and collaboration and to create more value for this kind of data.

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: https://creativecommons.org/licenses/by/4.0/
1 Introduction

Following the trends of opening up data for the public under free licenses and of letting the public contribute, this paper describes two such projects coming from the academic sector: the Bank of Finnish Terminology in Arts and Sciences (tiheetentermipankki.fi), hereafter BFT, and Sanat (sanat.csc.fi).

We used open source software to build platforms for dictionary data and imported a number of existing sources into them under a free license. We let the interested public contribute and we abolished a separate publishing step.

Building upon existing MediaWiki software, we developed the Multilingual Semantic MediaWiki (MSMW) framework to support such multilingual content and multilingual user base. A multilingual user base consists of polyglots and users with different linguistic backgrounds. By multilingual content we mean content which can be meaningfully accessed through multiple languages.

In the next section we will define our objectives and describe our projects. In the third section we give summary of previous work which we built upon. In the fourth section we describe what MSMW is. In the remaining sections we will present the results, draw conclusions and outline future work.

2 Scope of the research

Our objective is to create an on-line platform for users to collaboratively build specialized dictionaries: in this case terminologies for different fields of research or a multilingual dictionary for purposes of education in a lesser used language. In this section, we describe the general characteristics of two such dictionary projects.

The Bank of Finnish Terminology in Arts and Sciences (started in 2011) is a multi-disciplinary project for gathering a permanent terminological database covering all fields of research in Finland. BFT receives funding from the Academy of Finland and University of Helsinki; it is coordinated at the Department of Finnish, Finno-Ugrian and Scandinavian Studies at the University of Helsinki. The project aims to strengthen the parallel use of languages in the academic sector by providing a reliable, easily accessible and up-to-date terminological resource. BFT is split into multiple sub projects or expert groups, each coordinating terminology work in their own field of research. Each group comprises experts of its field, while terminological consultation and guidance are offered by the BFT staff.

In BFT, each term can contain the following information, compliant with ISO standards concerning terminological work: definition, explanation, images, references, additional information, categories, expressions and related terms. There are also some
rarely used fields or topic-specific fields, like the scientific name. Expression, related terms and images are special in a sense that they have an inner structure. Expressions contain the language of the expression, the expression itself, the characterization of equivalence and whether it is recommended, to be avoided, obsolete, etc. Related terms also contain the type of relation, e.g. hyponymy or meronymy, in addition to the term itself. Images hold the name of an image file and a description.

The primary language of most fields is Finnish, but there are also a handful of fields which use Swedish, English, or Finnish Romani. Primary language means the language of term descriptions and explanations: it is the starting point for concept systems. For example, in the topic of jurisprudence, the term *tuomari* (judge) has a very different meaning if we look at it from the view point of the Finnish legal system compared with what it means in common law systems.

*Sanat* is an editing and publishing platform under development (since 2014) to host multiple monolingual and multilingual dictionaries from the Institute for the Languages of Finland. A prototype was developed in 2014 in collaboration with the Institute for the Languages of Finland, Lyydiläinen seura, CSC – IT Center for Science and Niklas Laxström.

Sanat is composed of more independent dictionaries and hence differs in structure from BFT. While BFT is a concept-based terminology, Sanat is a lexicographical dictionary, where the starting point is a word, not a concept as it is in the BFT. A Ludic dictionary built by the Lyydiläinen seura was chosen as a pilot dictionary to be converted and imported to the Sanat prototype. The Ludic dictionary was a word processing document which used text formatting and special symbols to denote the structure of the data.

Each term in the Ludic dictionary can contain the following linguistic information: basic form; word class; variants and their inflection in different dialects; example sentences translated in different languages, usually Finnish and Russian. There are more than 1 500 Ludic terms in Sanat.

3 Past work: Semantic MediaWiki

We chose Semantic MediaWiki (or SMW; semantic-mediawiki.org) as our base. SMW is explained in this section. Section 4 will explain our extension of it, MSMW.

A wiki approach has multiple properties which have proven to be useful for our projects. All changes are immediately visible. Wikis are cheap to host and it is easy to keep the software up to date; occasionally spam fighting might become a problem. Wikis enable collaboration, as many users are already familiar with them through Wikipedia and it is possible to build an intuitive and predictable interface. People
work on content areas according to their interests and expertise, so some very specialized areas can thrive, while there are naturally gaps in the data.

Semantic MediaWiki is used by over 1600 websites as of 2014, for very different purposes [1]. Semantic MediaWikis include:

- the Finnish Järviwiki (jarviwiki.fi), which contains information about all the lakes in Finland and where users contribute their observations;
- translatewiki.net, created by Niklas Laxström, which is the second biggest SMW site in number of pages (4 millions as of October 2014) [2] and uses SMW for auxiliary functions;
- WikiApiary.com, which is the second biggest SMW site in number of semantic property values (30 millions as of October 2014) [1].

SMW has already been described extensively, in particular by Krötzsch and Vrandečić [3]: here we will only summarise why we chose it as a platform for our projects. SMW satisfied the following requirements:

- store and preserve all the information contained in the pre-existing thesaurus structure;
- offer a user-friendly interface to add and edit data, namely semantic forms which most users can use without knowing any markup¹;
- expose the data in meaningful and attractive ways, for instance on the main page, topic-based portals and other query interfaces to the data; allow researchers to search expressions with a certain form.

Moreover, with SMW the structure is not defined in the programming code, but instead the semantic relations and forms are defined in the wiki, with wiki markup. It is also easy to modify this structure on the fly, unless manual updates of the existing data are implied, and even then modifications are retrospectively possible to some extent. This lowers the barrier to build the dictionary structure. Practically all available resources (in terms of funding and time) can be used to customise the wiki platform for the intended purpose.

4 Multilingual Semantic MediaWiki

For both dictionaries, given their goals, we defined a set of requirements for multilingual support. We call a wiki which satisfies these requirements a Multilingual Semantic MediaWiki.

¹In fact, previous research has stressed how SMW is suitable for inexperienced users as well [4], so that the usability of web 2.0 and richness of the semantic web are not in opposition [5].
tic MediaWiki (MSMW). In this section we explain how we satisfied the requirements by using existing solutions or developing one ourselves.

Semantic wikis have been attempted in the past, often based on Controlled Natural Language \[6,7\] and with a multilingual approach \[8\], showing users reach a high level of consensus for content \[9\]. However, we find that all previous approaches failed to fully internationalize the user experience as we aim; and produced wikis which rarely are still on-line and in use. Therefore, the need of a more systematic and robust approach arises: the properties which we define next.

We need a semantic wiki provided with

1. automatic guesses of the user’s preferred language,
2. manual language switching,
3. input methods,
4. web fonts for language support,
5. translatable documentation,
6. translatable forms,
7. translatable content interface,
8. structured multilingual content.

Firstly, we consider multilingual something that strives to equally support all languages: hundreds rather than few. MediaWiki aspires to be "internationalized, with equal support for all languages," \[10\] and is being localised in over 350 languages, including right-to-left languages, with full support for any language specificity \[11\]. MediaWiki is the only existing platform satisfying our multilingualism requirement; no existing CMS, semantic platform or wiki engine can satisfy all the 8 properties, other than MediaWiki. This made us choose SMW as our base platform and call MSMW a system which satisfies all the 8 properties.

The MSMW framework is meant to leverage this extensive language support, by making sure that it extends to all semantic features of the wiki without degradations. As for features and interfaces shared between wikis, any defect or lack of translations should be fixed in the upstream MediaWiki code and in translatewiki.net respectively, to benefit all installs. As for content and interfaces specific to one wiki, they should be translatable on the wiki itself.

Parts 1.–5. are readily available to any MediaWiki instance by installing the MediaWiki Language Extension Bundle (MLEB)\(^2\). Parts 1.–4. constitute basic language support and are provided by the Universal Language Selector extension (included in MLEB), which uses information given by the user’s browser, geo-location of the user’s

\(^2\)https://www.mediawiki.org/wiki/MLEB
IP address and Unicode Common Locale Data Repository (CLDR) to choose and suggest most likely languages. The user can easily choose language manually when the Universal Language Selector fails to infer correctly.

Input methods and web fonts complement the support provided by browsers and operating systems, using JavaScript and web standards for font delivery. Lack of fonts is a common problem for many Indian languages. Lack of input methods is also common for many Indian languages, as well as small languages not included in computer standards and people who live abroad or are traveling.

Part 5, translatable documentation, can be achieved with the Translate extension (included in MLEB). When pages are prepared for translation according to the documentation of the Translate extension, they can easily be translated by translators using a dedicated translation interface inside the wiki. The interface provides common translation tools like translation memory, machine translation service integration, translation notes and most importantly change tracking. Change tracking ensures that translated versions are never out of date by integrating missing and outdated translations with the source language.

Content interface means that some elements of the interface are defined on the wiki, but follow the user’s interface language. This feature was used, for example, with headings and labels. The Translate extension also provides a way to tag those elements so that they can be translated. In Translate’s documentation, this method is called unstructured element translation. Parts 6. and 7. are an application of unstructured element translation to SMW, which to our knowledge has not been done before.

Structured multilingual content means the wiki has data input forms which can accept multilingual content and is able to store and display such multilingual content. For example, in BFT, the list of expressions in different languages for each term are multilingual content. To design structured multilingual content, one has to understand what parts of the data can be multilingual. Multilingual content does not necessarily follow the user’s interface language as content interface does.

In practice this means providing, for fields which accept content in different languages, an additional field where to set the language of linguistic content. Correct language tagging in HTML output is important for search engines and application of web fonts. Language annotations of the data are used in semantic queries and by third party users of your data.

³https://www.mediawiki.org/wiki/Universal_Language_Selector/FAQ#How_does_Universal_Language_Selector_determine_which_languages_I_may_understand
5 Other semantic structure characteristics

In MediaWiki, all content is split across pages. A page is like a web page in that it has no fixed length, unlike printed pages. A page has content on a specific topic, usually defined by the page title, which is also used in the unique address of the page. Pages are sometimes also known as articles if it fits the type of content, like in Wikipedia. Furthermore, namespaces are used to separate different types of content, e.g. help pages are in a separate namespace. Namespace appears in page title as prefix separated by colon, for example Help:Editing.

For BFT we created multiple additional namespaces. Each sub group, which we call a terminology, has its own namespace. Thanks to this Kielitiede:kieli (language in linguistics) and Eläintiede:kieli (tongue in zoology) are two separate terms. When links are created to other pages, the namespace is not usually visible in the link text. We gave each terminology a separate color, which is shown in page titles and links across the interface. This allows users to know in which terminology they are.

In addition to each terminology having its own namespace, we also created a namespace for all the expressions. The pages in the expressions namespace contain information which relates to the surface forms like word class and language. The pages link back to all terms in any terminology which contain that expression. In the case that multiple expression have same form, they will share the same page with information for both.

What caused most headaches for us were the limitations of the semantic structure. The basic tool we have is a subject-property-value triplet, where the subject is always implicitly a page in the wiki. Properties we can define freely, but the values cannot have inner structure; in other terms, SMW can only store 2D data, not 3D or N-dimensional data.

For example, we have a word talo in Finnish and we want to give multilingual examples of sentences where it is used. We were unable to say that the subject talo had a property example-sentence with value {fi: talo paloi; en: house burned}. We tried to store the data without semantic relations, but that did not work either, because semantic forms have limitations with so-called multiple instance templates⁵. In the end this problem was solved in different ways in BFT and Sanat. In BFT we do not have such complex embedded structure. In Sanat we moved that kind of data into separate pages (subpages in MediaWiki), which forced us to provide editing controls directly on the page itself, hence mixing up view mode and edit mode.

⁵https://www.mediawiki.org/wiki/Extension:Semantic_Forms/Defining_forms#Multiple-instance_templates
From Sanat’s characteristics follows the main structural difference from BFT: there is no shared global namespace for expressions. Information related to expressions is included in the term pages. Multilingual examples of terms are stored in separate pages due to reasons described above. The section "Related terms" is replaced by a generic section "See also" on the same page. Examples are given as sentences with translations.

Using semantic queries, we also automatically created two reverse dictionaries: Finnish to Ludic and Russian to Ludic. We expect that in Sanat the fields will be more customised for each dictionary, as opposed to BFT where all sectors of research contain the same fields to a great extent.

6 Outcomes

We found out that we could come up with working prototypes in just few hours, including the time to set up MediaWiki with many extensions. After the initial launch of the BFT, we were also able to quickly satisfy user feedback thanks to the flexibility of the platform.

We made an extension to MediaWiki, MixedNamespaceSearchSuggestions⁶, to show more suggestions when the user types something in the search box and to often eliminate the need for a full text search. First, suggestions are shown from all namespaces (dictionaries) at once, with no need for the keyword to match the namespace name. Second, the namespace is shown next to each suggested title. The extension is released with an open source license.

While developing BFT we found out that some form elements, like certain types of buttons, did not allow translation with the approach used in MSMW. We submitted patches to fix some of these, but not all cases have been fixed yet.

In 2014, BFT has reached about 30 000 concept article pages, corresponding to about 70 000 terms in 35 languages. The contents of the concept articles vary from terminology to terminology, as different work groups present different choices in workflow and differing stages of progress. For example terminology of Jurisprudence contains nearly 2 000 articles with extensive contents, while Epidemiology contains not much more than Finnish-English term lists. Even still, according to user survey conducted in the spring of 2014, 85 percent of the respondents said that they had found information they were looking for either completely or partly.

BFT’s constant activity of 20 to 40 monthly active editors has ensured a constant growth of the dictionary⁷. Editors are mostly academics from across Finland. For

⁷https://wikiapiary.com/wiki/Tieteentermipankki.fi
comparison, the Finnish Wiktionary has around 30 editors active in a given month\(^8\). Some active contributors helped the wiki’s development beyond their edits, for example by negotiating a licence for an existing terminology with hundreds of terms to be imported into the wiki. Expert participation on the platform is not thus limited to only writing new terminological records, but encompasses also selecting, revising and updating existing terminological records for import. The added value of bringing resources to BFT is the possibility to integrate separate terminologies in to one easily accessible resource. According to BFT’s user survey of 2014, its resources are widely used by undergraduate students in those fields, where the contents have reached sufficient extension.

Since Sanat is still not publicly launched, we cannot qualify its success in terms of users and contributors. We can say it took less than 40 hours to develop it, including conversion of an existing dictionary in a format suitable for import in the wiki.

7 Conclusion

BFT is a successful project which has benefited from the support for multilinguality. It does not compete with general purpose dictionary projects like Wiktionaries or OmegaWiki due to its specific scope and customisations to support terminology work.

MSMW enhances a regular SMW instance by making it suitable for multilingual users and multilingual content. This is different from for example Wikipedia and Wiktionary, where each language version is a separate instance with a separate community. Wikipedia and Wiktionary also do not use SMW to structure their data. The novel part of MSMW is the idea of combining MediaWiki, MLEB and SMW and the ways how to best use them together to provide additional value. We have contributed to all of the components to make them work better together.

By using MSMW on a projects like BFT we know that MSMW works in practice. The main issue is the manual work needed to set up unstructured element translation. Also the issues with untranslatable elements in forms, until fixed, make MSMW a less compelling approach. Even with these issues, MSMW is already useful because of the benefits in rapid prototyping and the superior language support.

Our code changes have been integrated in MediaWiki and extensions or released as new open-source extensions to MediaWiki where applicable.

\(^8\)https://stats.wikimedia.org/wiktionary/EN/TablesWikipediaFI.htm#editor_activity_levels

---

3.6 Multilingual Semantic MediaWiki for Finno-Ugric dictionaries [page 83 of 131] http://dx.doi.org/10.7557/sec.2015.3
8 Future work

MSMW extensions can be further developed to reduce the manual work needed for a new MSMW wiki to be translatable using the unstructured element translation feature. Currently, the Universal Language Selector is not employed for language selection in forms, but this should be relatively easy to add.

The idea of separate view and edit modes may be considered outdated by the supporters of in-place editing [12]. This might become an issue in the future if people start seeing our platform as outdated and hard to use. Replacing the form paradigm with in-place editing would be a huge undertaking in SMW.

Integration of our data with external data sources is still to be solved. SMW provides machine-readable application programming interfaces (API). They are not, however, tied to any general vocabulary, which means that a developer would have to map the properties and values manually, and each wiki can have different structure. A simple standardized format should be developed for our dictionaries to be used as data providers for tools such as Content translation⁹, currently in development by the Wikimedia Foundation.

Other researchers working with SMW stress the importance of alternative modes of accessing lexicographical data such as maps, timelines and charts [13]. We have not yet explored how to apply MSMW in such a context.

Finally, MSMW is defined strictly by the actual software we have used. A more general, software independent approach can be developed for creating multilingual collaborative content management systems by starting from the important concepts such as the separation between the software interface, the user created content interface and the user created content.

Acknowledgements

Niklas Laxström, co-author of this paper, is a MediaWiki consultant and has received funding from the University of Helsinki to develop BFT and from Lyydiläinen seura to develop Sanat. He also works with Wikimedia Foundation which is a contributor to MediaWiki software [14].

We thank Federico Leva for detailed comments on the manuscript and helpful discussions regarding this work and previous research, aided by his extensive knowledge of wikis.

⁹https://www.mediawiki.org/wiki/Content_translation
References

[1] Jamie Thingelstad. Semantic statistics. https://wikiapiary.com/wiki/Semantic_statistics, October 2014. WikiApiary, as of this writing, monitors 24,997 MediaWiki websites, of which over 21,000 were provided by WikiTeam and Federico Leva. Some well known Semantic MediaWikis are not included in the statistics because they don’t publicly expose the required data.


The Finno-Ugric Languages and The Internet project

Heidi Jauhiainen
University of Helsinki
Department of Modern Languages
heidi.jauhiainen@helsinki.fi

Tommi Jauhiainen
University of Helsinki
Department of Modern Languages
tommi.jauhiainen@helsinki.fi

Krister Lindén
University of Helsinki
Department of Modern Languages
krister.linden@helsinki.fi

December 12, 2014

Abstract

This paper describes a Kone Foundation funded project called "The Finno-Ugric Languages and The Internet" together with some of the achieved results. The main activity of the project is to crawl the internet and gather texts written in small Uralic languages. The sentences and words of the found texts will be assembled into a freely available corpus. Crawling is done using the open source crawler Heritrix, which is developed by the Internet Archive. Heritrix crawls through the pages and passes the found texts to a language identifier. We are using a state of the art language identifier, which has been further developed within the project and has been evaluated using 285 languages. We describe the language
identification evaluation results concerning the 34 Uralic languages known by the language identifier. We also describe the initial observations and results from the first five large crawls which were done in the national internet domains of Finland, Sweden, Norway, Russia, and Estonia.

1 Introduction

"The Finno-Ugric Languages and The Internet"-project\(^1\) started at the beginning of 2013 as part of the Kone Foundation Language Programme [1]. The project is located at the Department of Modern Languages\(^2\) at the University of Helsinki and is part of the international CLARIN\(^3\) cooperation. The main goal of the project is to build a prototype of a system that will crawl the internet and gather texts written in small Uralic languages. Web crawling has been used to collect text corpora for a variety of languages [2, 3], but to our knowledge, this project is the first collecting the texts in small Uralic languages. The largest Uralic languages Hungarian, Finnish, and Estonian are outside the scope of the project. We are using language identification software to detect the language of the crawled web-pages. The gathered texts will be collected into sentence and word corpora for each language and the links to the associated web-pages into link collections. The corpora will act as a source for linguists and the link collections will hopefully spread the knowledge of the existence of relevant pages to interested parties. Due to the copyrights connected with longer texts, we are only publishing corpora of up to sentence-length text snippets with links to the original text on the internet. The negotiations for free use of copyrighted texts would be far beyond the resources of this project. We aim at making the complete work-flow from the crawling to the creation of corpora as automated as possible.

In Section 2, we talk about the language identifier used in the project with respect to Uralic languages. Section 3 deals with the five large crawls done so far, i.e. in the national domains of Finland, Sweden, Norway, Russia and Estonia. In Sections 4 and 5, we describe the link collection and the sentence corpora respectively.

2 Language Identification for Uralic Languages

We are using an extended version of the language identifier described in [4]. The extended version of the language identification method will be described in a forthcoming journal article [5], where it is evaluated together with the methods presented

\(^1\)http://suki.ling.helsinki.fi
\(^2\)http://www.helsinki.fi/modernlanguages/
\(^3\)http://clarin.eu
in [6], [7], [8], [9], [10], and [11]. The language identifier uses relative frequencies of n-grams of characters together with tokens and token-based backoff. The evaluated language identifier recognizes 285 languages from all around the world. The definition of a language is taken from Ethnologue [12] and the division of the languages is as in the ISO 639-3 standard*. The 285 languages include 34 Uralic languages: Hungarian, Khanty, Mansi, Estonian, Finnish, Kven Finnish, Tornedalen Finnish, Ingrian, Karelian, Liv, Livvi-Karelian, Ludian, Veps, Votic, Võro, Hill Mari, Meadow Mari, Erzya, Moksha, Udmurt, Komi-Permyak, Komi-Zyrian, Inari Sami, Kildin Sami, Skolt Sami, Ume Sami, Lule Sami, North Sami, South Sami, Nenets, Nganasan, Forest Enets, Tundra Enets and Selkup. The current version of the language identifier only knows one orthography per language, but this will be corrected in future versions.

The evaluation of the language identifier was done in tests using sequences from 5 to 150 characters in length and the recall figures for Uralic languages can be seen in Table 1. The character sequences are random parts of the test corpus, always beginning from the beginning of a word. For most of the languages the test set consists of the texts of the Universal Declaration of Human Rights in that particular language and the training set is the text from Wikipedia. The aim was that the test set would always be a text from a different domain than the training text. This was easily possible for most of the larger languages, but quite difficult for some of the smaller Uralic languages. In some cases, such as Forest and Tundra Enets, the test set is from a different section of the same document as the training text. The amount of training material differs considerably between languages ranging from 19 000 characters in Ume Sami to over 400 million characters in the Hungarian material.

The average identification accuracy for Uralic languages is generally slightly lower than for all languages. This is due to some of the languages being very close varieties of each other, especially within the Finnic languages. The language identifier has not been optimized to perform better with Uralic languages or even with closely related languages. In the test length of 20 characters, the overall average is 93.9% whereas the average is 90.5% for the Uralic languages. Almost all languages attain 100.0% recall at 150 characters. Table 1 also includes the recall figures attained by the widely used method described in [6].

Table 2 is a confusion matrix showing the kind of mistakes that were made in the language identifications between most of the Finnic languages in the 20 character sized tests. Notable problem pairs are those of Finnish (fin) and Tornedalen Finnish (fit), Tornedalen Finnish (fit) and Kven Finnish (fkv), as well as Ludian (lud) and Livvi-Karelian (olo). The Samic languages are not as easily confused and no table is presented for them.

*http://www.sil.org/iso639-3/
Table 1: Recalls of Uralic languages obtained by the two language identifiers for test lengths between 5 and 150 characters. Percentages are averages over 1000 sample sequences of each length. The figures on the left are for the identifier developed within the project and the figures on the right are for an identifier using the well-known method of Cavnar & Trenkle.
Table 2: Confusion matrix of Finnic languages. The Finnic languages were mistaken also as other languages, and if the figures of the other languages would be added to the table, the rows would add to 100.0%.

Table 3 shows the confusions between most of the Uralic languages written in Cyrillic script. Forest Enets seems to dominate over Tundra Enets as does Komi-Permjak over Komi-Zyrian. The language model for Komi-Permjak is based primarily on Wikipedia and the one for Komi-Zyrian on a bible translation. The test material for Komi-Zyrian is also from the bible, which should in fact make it easier to identify but, nevertheless, 6.9% of the 20 character extracts are identified as Komi-Permjak.

Table 3: Confusion matrix of some of the Uralic languages written in Cyrillic script. As with the Table 2, the rows in this table would add to 100.0% if the figures for all the 285 languages would be shown.

3 Crawling the National Domains

In order to crawl for pages written in small Uralic languages, we use Heritrix [13], a web archiving system developed by the Internet Archive\(^3\). We chose to use Heritrix
after considering several available crawlers. Heritrix is the outcome of many years of development by the Internet Archive and it is still being maintained. In the beginning it was a product of cooperation between Internet Archive and the Nordic national libraries and it is still used by several national libraries around the world to collect national web archives. It has also been successfully used for collecting similar corpora by [2].

The goal of the Internet Archive is to archive the sites as usable collections for future generations. In this project, we are only interested in collecting the textual material in the small Uralic languages. The version we are currently using downloads all text files as well as pdf files it finds from within the domain in question. We have made some custom changes to the code of the crawler so that, when a file has been downloaded, the running text is extracted from it. The crawler sends an excerpt of 300 characters from the middle of text to the language identifier which responds with the ISO-639-3 code of the language of the text. If the language is one of the Uralic languages we are interested in, the crawler sends the whole text to be re-identified. If this identification still points to a small Uralic language, the whole text of the page is archived. The address and the identification results of all crawled pages, including the ones rejected, are stored.

We have chosen to start collecting the material by crawling the national domains most likely to contain material written in small Uralic languages, i.e. .ee, .fi, .no, .ru, and .se. Table 4 shows the statistics for each of the five national domain crawls. The first column "URLs" indicates the total number of downloaded files during the crawl and the fourth column "domains" indicates how many subdomains were crawled. For the Russian crawl "domains" numbers only the top level domains as our crawling tactic had changed when the crawl started. The second column "LI-1 URLs" gives the number of pages identified to contain small Uralic languages during the crawl. The third column "LI-2 URLs" is the number of pages still identified as Uralic after a more
precise language identification which was done after the crawl. The fifth and sixth columns indicate the number of subdomains in Uralic languages before and after the more precise identification.

<table>
<thead>
<tr>
<th></th>
<th>URLs</th>
<th>LI-1 URLs</th>
<th>LI-2 URLs</th>
<th>domains</th>
<th>LI-1 domains</th>
<th>LI-2 domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>.fi</td>
<td>354 000 000</td>
<td>89 166</td>
<td>39 056</td>
<td>450 000</td>
<td>2 824</td>
<td>1 465</td>
</tr>
<tr>
<td>.se</td>
<td>308 000 000</td>
<td>16 687</td>
<td>14 979</td>
<td>1 500 000</td>
<td>676</td>
<td>439</td>
</tr>
<tr>
<td>.no</td>
<td>358 000 000</td>
<td>137 059</td>
<td>133 513</td>
<td>800 000</td>
<td>636</td>
<td>586</td>
</tr>
<tr>
<td>.ru</td>
<td>172 000 000</td>
<td>18 122</td>
<td>8 585</td>
<td>1 400 000</td>
<td>3 243</td>
<td>909</td>
</tr>
<tr>
<td>.ee</td>
<td>108 000 000</td>
<td>22 785</td>
<td>13 496</td>
<td>100 000</td>
<td>500</td>
<td>232</td>
</tr>
</tbody>
</table>

Table 4: Statistics for the crawls of the five national domains.

In the following paragraphs we make a few notes of the individual national domain crawls. We will be doing new crawls for them all as most of the crawls ended before the domains were really exhausted. It is actually far from trivial to define when we have exhausted a national domain. There are many sites that dynamically generate an infinite number of web-pages and even sub-domains, which makes each of the national domains infinite in size if we are calculating the number of pages or sub-domains. Currently we have set the crawler to accept only up to 100 000 pages per top-domain. Even this does not allow us the luxury to just wait for the exhaustion of the queued URLs, as some of the sites are very slow to serve the pages and waiting for them to reach the page limit could take months or even years. We are now trying to determine if the speed of the crawl could be used as an additional indication of domain exhaustion. We could consider, for example, that if the hourly average speed drops below 10% of the average speed of the first week of the crawl, the domain is exhausted. As we have not yet stabilized our criteria for exhaustion, the current figures can not really be compared with each other and do not give a realistic picture of the size of the national domains.

**Finnish .fi domain**  In the crawl of the Finnish internet we downloaded around 354 million files. The Finnish crawl was terminated as the crawler was running out of disk space and the speed had slowed down to around 25 pages per second. The average speed for the first week of the crawl was 282 pages per second, so we could consider the crawl exhausted. The Finnish language model used in the language identifier is derived from the Finnish Wikipedia, which is mostly written in the official form of written Finnish. However, many people do write texts using the forms of their respective dialects. The written forms of Tornedalen Finnish, Kven Finnish, and Ingrian are much closer to these written Finnish dialects than the official written Finnish. This creates a problem as a great number of texts in dialectal Finnish are identified as these three languages. This could be corrected by creating separate language models...
for written dialects of Finnish.

**Swedish .se domain**  In the crawl of the Swedish internet we downloaded around 308 million files and it was terminated after the speed of the crawl had slowed to around 20 pages per second, which is well below 10% of the 238 pages per second average for the first week of the crawl. In this crawl, the library systems which were localized for Northern Sami turned out to be a problem. Over 100 domains dedicated to library systems were found by the crawler, the largest of them, bibliotek.nora.se, with 2 322 pages in Northern Sami. Not only do the library catalogue links expire quickly, they usually include the same text over and over again. We will have to incorporate a double-checking mechanism before creating the link collections and corpora in order to avoid collecting the same text many times. Some methods for removing doubles are introduced in [2] and [3].

**Norwegian .no domain, Russian .ru domain, and Estonian .ee domain**  All of the three crawls ended in problems with either software, hardware or the crawl strategy used. The national domains were far from exhausted by any criteria we have considered. As most of the pages written in Uralic languages have been found in our crawl of the Norwegian internet, we are including the statistics for these crawls in the Tables 4 and 5.

## 4 The link collection

The link collection that is available at the time this was written has been curated by hand from the pages of the .fi crawl. It contains links to 266 sites from which text was found in 19 of the 31 small Uralic languages searched. The links have not been verified by experts or native speakers. We are planning to incorporate a simple crowdsourcing platform to be able to get feedback from those who are more familiar with the languages. The links lead to the actual pages currently found on the internet, so it is certain that some of the links will break while time passes. We will not remove the broken links completely from the database, but move them elsewhere and, if possible, make links to corresponding pages in the Internet Archive.

Our goal is to make the creation of the link collection as automated as possible, avoiding manual link curation. The list of sites from the Finnish crawl available at the moment includes only the front page of some sites although more pages were found during the crawl. In the future all the links found when crawling will be in the list of

---

*http://suki.ling.helsinki.fi/sites*
links. The greatest problems will arise from the pages, which are written in a correctly identified language, but are near-doubles of other pages as in the case of the Swedish library systems mentioned above.

5 Sentence corpora

When we are creating a sentence corpora, one of the greatest problems we have at the moment is that many of the downloaded pages are multilingual. We are currently making a survey of the methods for language identification in multilingual documents and in future we will incorporate a multilingual detection method in the system. We did a separate language identification for all the lines of all the files containing small Uralic languages in order to see which ones are, indeed, written in the language indicated by the identification of the file as a whole. The first column of Table 5 shows the number of unique lines identified as written in the respective language. The columns 2 and 3 show the total number of words and characters in these lines.

Even though the language identification used is state of the art, it is far from perfect. The collections have not been checked by experts in the corresponding languages, but some things are clear even for a layman. The three smallest collections Selkup, Nganasan, and Tundra Enets do not actually contain the intended language at all, but are mostly some sort of lists of model numbers in Cyrillic for Nganasan and Tundra Enets. The Selkup collection consists of pages frequented with the word "Статья", "article", which is a very frequent word in the training text used for Selkup’. The Ingrian collection contains mostly hyphenated or otherwise broken Finnish or Finnish dialects written as spoken. Especially south-western Finnish dialects seem to be identified as Ingrian. Northern Finnish dialects are identified as either Kven or Tornealen Finnish. In order to fix these problems with dialectal Finnish, we will try to include separate language models for dialectal Finnish in the future. Some long lists of names from the Swedish crawl have been identified as Ludian and are now polluting the collection of the language. The Khanty collection is polluted by long lists of model numbers written in Cyrillic script.

Future work

Language identification methods will be further developed in order to improve the robustness of the language identifier we use. We will also try to enhance the language models in order to more efficiently distinguish small languages from various dialects
Table 5: The number of lines, words, and characters in small Uralic languages after language identifying each individual line.

<table>
<thead>
<tr>
<th>Language</th>
<th>#unique lines</th>
<th>#words</th>
<th>#characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Northern Sami (sme)</td>
<td>312 150</td>
<td>3 209 570</td>
<td>30 314 461</td>
</tr>
<tr>
<td>Vořo (vro)</td>
<td>167 997</td>
<td>3 239 365</td>
<td>22 940 862</td>
</tr>
<tr>
<td>Ingrian (izh)</td>
<td>98 743</td>
<td>2 960 322</td>
<td>22 054 552</td>
</tr>
<tr>
<td>Eastern Mari (mhr)</td>
<td>132 692</td>
<td>1 626 001</td>
<td>20 586 975</td>
</tr>
<tr>
<td>Western Mari (mrj)</td>
<td>137 739</td>
<td>882 884</td>
<td>10 115 581</td>
</tr>
<tr>
<td>Southern Sami (sma)</td>
<td>86 856</td>
<td>814 187</td>
<td>9 264 654</td>
</tr>
<tr>
<td>Udmut (udm)</td>
<td>41 133</td>
<td>570 633</td>
<td>7 554 055</td>
</tr>
<tr>
<td>Erzya (myv)</td>
<td>29 742</td>
<td>503 107</td>
<td>6 911 773</td>
</tr>
<tr>
<td>Lule Sami (smj)</td>
<td>53 734</td>
<td>376 067</td>
<td>3 436 123</td>
</tr>
<tr>
<td>Inari Sami (smn)</td>
<td>35 740</td>
<td>352 319</td>
<td>3 428 425</td>
</tr>
<tr>
<td>Torneåalen Finnish (fit)</td>
<td>21 133</td>
<td>384 037</td>
<td>3 137 644</td>
</tr>
<tr>
<td>Moksha (mdf)</td>
<td>15 931</td>
<td>202 740</td>
<td>2 853 814</td>
</tr>
<tr>
<td>Komi-Zyrian (kpv)</td>
<td>13 139</td>
<td>205 243</td>
<td>2 374 729</td>
</tr>
<tr>
<td>Skolt Sami (sms)</td>
<td>23 354</td>
<td>188 873</td>
<td>2 010 098</td>
</tr>
<tr>
<td>Livvi (olo)</td>
<td>6 622</td>
<td>112 560</td>
<td>940 632</td>
</tr>
<tr>
<td>Liv (liv)</td>
<td>12 194</td>
<td>85 171</td>
<td>602 979</td>
</tr>
<tr>
<td>Kven Finnish (fkv)</td>
<td>3 414</td>
<td>57 199</td>
<td>500 600</td>
</tr>
<tr>
<td>Ludian (lud)</td>
<td>2 078</td>
<td>53 094</td>
<td>485 457</td>
</tr>
<tr>
<td>Khanty (kca)</td>
<td>7 244</td>
<td>38 704</td>
<td>378 562</td>
</tr>
<tr>
<td>Veps (vep)</td>
<td>5 480</td>
<td>29 691</td>
<td>324 504</td>
</tr>
<tr>
<td>Komi-Permyak (koj)</td>
<td>4 370</td>
<td>19 982</td>
<td>186 543</td>
</tr>
<tr>
<td>Karelian (krj)</td>
<td>950</td>
<td>11 550</td>
<td>103 498</td>
</tr>
<tr>
<td>Mansi (msn)</td>
<td>319</td>
<td>4 997</td>
<td>60 811</td>
</tr>
<tr>
<td>Votic (vot)</td>
<td>702</td>
<td>5 895</td>
<td>42 657</td>
</tr>
<tr>
<td>Kildin Sami (sdj)</td>
<td>332</td>
<td>2 751</td>
<td>32 409</td>
</tr>
<tr>
<td>Ume Sami (sju)</td>
<td>194</td>
<td>2 636</td>
<td>21 703</td>
</tr>
<tr>
<td>Nenets (yrc)</td>
<td>209</td>
<td>1 165</td>
<td>14 370</td>
</tr>
<tr>
<td>Selkup (sel)</td>
<td>639</td>
<td>1 486</td>
<td>12 849</td>
</tr>
<tr>
<td>Nganasan (nio)</td>
<td>195</td>
<td>428</td>
<td>6 950</td>
</tr>
</tbody>
</table>
| Tundra Enets (enh)        | 6             | 14       | 112         

and to identify languages in multilingual documents. The material found during the already performed crawls will be of assistance for this.

We will, furthermore, try to increase the speed of the crawler in order to crawl more widely and more often. The most important national domains in regard to the Uralic language speakers will be re-crawled with more depth and more frequency. We also intend to look into crawling the .com and .org domains. We would also like to extract text from other binary files than pdfs.

Acknowledgments

We are thankful for the support of the Kone Foundation and to Jack Rueter for sharing his invaluable resources in Finno-Ugric languages. We also thank the anonymous reviewers for their suggestions and references.
References


On the Road to a Dialect Dictionary of Khanty Postpositions

Zsófia Schön
Ludwig Maximilian University of Munich
Institute for Finno-Ugric Studies
zsophia.schoen@gmail.com

December 17, 2014

Abstract

This paper aims to present the first steps of a corpus based dialect dictionary of postpositions in several Khanty dialects and subdialects. Based primarily on specifically elicitated data from more than fifty informants, this ongoing project focuses not only on the semantic properties of this part of speech in Khanty, but also on the morphology and combinatorics as exhibited by (sub)dialectal microvariation. Special attention is paid to two of the Northern dialects – Kazym and Shuryshkary Khanty – and to one of the Eastern dialects – Surgut Khanty.

The lexicon entries have been compiled according to TEI P5 guidelines in XML format, while the corpus data is stored in a MySQL database. A web application combining the lexicon with the corpus data, sound files, annotations and metadata is currently under construction.

As a multilingual dialect dictionary of Khanty postpositions, this project hopes to fill a gap in current research on Khanty: namely the lack of easily accessible digital dictionaries. It is designed to be a pilot project for forthcoming digital Khanty dictionaries.
1 Introduction

Khanty (Ostyak) language is spoken in North-West Siberia along the river Ob and its tributaries. According to the 2010 Russian census, some 30,000 Khanty are living there, but only about 20% of them are native speakers.

Belonging to the Ob-Ugric branch of the Finno-Ugric language family, Khanty demonstrates the highest dialectal variability within this language family. Most of these dialects are highly endangered.

I hold the opinion that despite the relative spatial proximity of the dialects and subdialects of Khanty, even the lexicon of a grammatic category, like the one of postpositions, shows a level of variation so high, that the Khanty language cannot be considered a homogeneous entity.

Postpositions are considered a part of speech which marks grammatical function [1]: 224. They can be categorised either as a homogeneous closed word class or as a heterogeneous open word class – depending on the grade of grammaticalisation, e.g. nouns with case suffixes used as postpositions. In Khanty dialects, postpositions occur only with an antecedent at the left which can be a substantive (always in nominative), a pronoun, an adverb, an adjective, a numeral or a participle. The antecedent and the postposition are inseparable and form an adverbial construction. The postpositions themselves can be variable or invariable, i.e. if they take possessive suffixes or not. (For a detailed description and categorisation of Khanty postpositions, see [2].)

1.1 About the Project

This ongoing lexicological project originates from my doctoral thesis about postpositional constructions in several Khanty dialects [2] which was started in October 2009 and completed in July 2014. It contains, as an appendix, a Dialect Dictionary of Khanty Postpositions, formatted from XML into LATEX using XSLT, as well as selected sample data from the partly tagged and analysed corpus.

The dictionary entries have been compiled in XML according to the TEI P5 guidelines, the corpus data is stored in a MySQL database. The corpus consists of approximately 7,900 sentences containing postpositions, where the postpositional constructions have been glossed and tagged.

The goal of the current project is to present the material and lexicon in a digital and freely accessible format on the web. The project is supported by the IT Group for the Humanities at the Ludwig Maximilian University of Munich, particularly by Christian Riepl and Stephan Lücke as database designers, Gerhard Schön as developer of the web application, and myself working as lexicographer.
1.2 About the Khanty Dialect System

Although there is a large number of studies on Khanty dialectology (see e.g. [3], [4], [5], [6]), no consensus has been established so far about terminology or structure of the Khanty dialect system. Different regroupings and dialect names can be found in the literature, where either phonetic or morphologic aspects have been taken to classify the Khanty dialects (cf. e.g. [7], [8], [9], [10], [11]).

In this paper, the term dialect is used according to the definition of Chambers and Trudgill: a dialect “refers to varieties which are grammatically (and perhaps lexically), as well as phonologically different from other varieties” [12]:5.

Subdialects, however, form smaller entities, are even more concrete than dialects and always mutually intelligible. The minor differences between subdialects appear only in the lexicon or on a phonetic level.

Existing heterogenous approaches, diachronic tendencies and endangerment of dialects necessitate the establishment of a coherent classification of Khanty dialects not only on a dialectal, but also on a more concrete, subdialectal level.

Therefore, I propose a dialect system with a set of subdialects; see Table 1, described in my doctoral thesis ([2]: 9–45), although the number of subdialects could increase as a result of further research. The compilations were made on the basis of all accessible Khanty dialect dictionaries, especially [13] and [14], of secondary literature, as well as of own field notes and field observations.

2 The Digital Dialect Dictionary of Khanty Postpositions

In print, there exist excellent dictionaries of Khanty dialects, like the DWS [13] or the SVH [14]. In digital form, however, easily usable and accessible dictionaries are still missing.

The current project aims at changing this by creating an easily searchable multilingual online dictionary with a user-friendly web interface.

The lexicon entries were compiled on the one hand from a synchronic point of view, which is guaranteed by the corpus data collected between 2010 and 2012, and on the other hand from a diachronic point of view, which is ensured by information provided by the two aforementioned lexical resources. Basically, the lexicon entries are sorted alphabetically, where the vowels are judged one by one, unlike the Finno-Ugric tradition which would treat all vowels as one entity.

The digital lexicon entries will be enriched by sound files, annotations, as well as metadata about informants, dialects and subdialects, circumstances of the collection etc.
<table>
<thead>
<tr>
<th>Dialect</th>
<th>Subdialect</th>
<th>Southern dialect group</th>
<th>Northern dialect group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eastern dialect group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vakh-Vasyugan dialect</td>
<td></td>
<td><em>Irtysh dialect</em>¹</td>
<td><em>Middle-Ob dialect</em></td>
</tr>
<tr>
<td>Surgut dialect</td>
<td>Pim subdialect</td>
<td></td>
<td>Keushi subdialect</td>
</tr>
<tr>
<td></td>
<td>Tremyugan subdialect</td>
<td></td>
<td>Muligort subdialect</td>
</tr>
<tr>
<td></td>
<td>Tromagan subdialect</td>
<td></td>
<td>Nizyam subdialect</td>
</tr>
<tr>
<td></td>
<td>Agan subdialect</td>
<td></td>
<td>Sherkaly subdialect</td>
</tr>
<tr>
<td></td>
<td>Yugan subdialect</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Likrisovskoe subdialect</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Southern dialect group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Salym dialect</td>
<td></td>
<td></td>
<td><em>Kazym dialect</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Lower Kazym subdialect</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Middle Kazym subdialect</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Upper Kazym subdialect</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Northern dialect group</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shuryshkar dialect</td>
<td></td>
<td></td>
<td><em>Beryozov dialect</em></td>
</tr>
<tr>
<td></td>
<td>Poslov subdialect</td>
<td></td>
<td>Tegi subdialect</td>
</tr>
<tr>
<td></td>
<td>Shuryshkar subdialect</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Muzhi subdialect</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Synya subdialect</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Khanty dialect and subdialect system compiled from [2]: 45.

¹ Since the extent of divergation of the around twenty-six subdialects of the Irtysh dialect is unclear, and as it is impossible to say how many of them can be really called subdialects, because they are all extinct, I refrain from naming them here.
The headwords of the lexicon entries are organised with the help of abstract lemmata, which were elaborated by referencing to the phoneme systems of the dialects from a synchronic point of view. This was necessary because of the strong divergence between the phoneme systems, and makes it possible to organize the very same postposition from different dialects into one entry without favouring one form from one dialect over another, and without relying on the knowledge of the dictionary user about the historical phonology of Khanty.

At present, the languages of the dictionary are seven Khanty subdialects as well as German, but it is intended to add more Khanty (sub)dialects from mostly written sources and a translation into Russian.

2.1 The Corpus

The corpus data stored in a MySQL database was collected by myself in Siberia during four field trips, all together six months, between 2010 and 2012 from over fifty native speakers of three dialects and seven subdialects, in particular of the Tromagan and Yugan subdialects of the Surgut dialect, all three subdialects of the Kazym dialect and all two abovementioned subdialects of the Shuryshkary dialect.

The material consists of a total of over 7,900 sentences containing postpositions, which were elicited mainly with the help of Russian stimuli. Even if this method has its limitations (e.g. the data is not suitable for analysing word order), it allowed me to assemble a heterogenous collection of postpositions in a short time from a high number of informants with subdialectal microvariation. The free translations show a large range of usage of the different structures in contemporary spoken language.

The postpositional constructions have been glossed and tagged according to the Leipzig glossing rules.

Concerning the transcription of the data, a transcription in IPA was elaborated – for Kazym and Surgut Khanty (sub)dialects, this was already done in the EUROBABEL project “Ob-Ugric languages: conceptual structures, lexicon, constructions, categories”, followed by the subdialects of Shuryshkary Khanty during my work on the doctoral thesis. The transformation of the data into cyrillic orthographies for the different (sub)dialects will be planned at a later stage.

2.2 Structure of the Lexicon Entries

As mentioned before, the lexicon entries have been compiled according to TEI P5 guidelines in XML format. Special consideration has been given to the morphology of the postposition, its combinatorics and the highlighting of the (sub)dialectal level
of information, with dialect abbreviations from the two aforementioned dictionaries and my collection.

The structure of the lexicon entries looks as follows, e.g. in the case of the headword JUP-:

The headword is immediately followed by the etymology of the postposition as given in the dictionary DWS [13]:

<entry>
  <form type="lemma">JUP-</form>
  <etym>Grammatikalisiert aus Substantiv mit der Bedeutung 'Hinterseite (eines lebenden Wesens)'.</etym>
  <bibl>DWS 328</bibl>
</entry>

Then the forms of the postposition itself are given along with specified usage and details about the subdialect in which the forms of the postpositions occur:

<gramGrp>
  <case n="I">In Lokativ</case>
</gramGrp>

<form type="inflected">jupiʃən
  <lang>KAZ_KAM</lang>
  <lang>KAZ_KAK</lang>
  <lang>KAZ_KAO</lang>
  <lang>SHU_POS</lang>
  <lang>SHU_SYN</lang>
</form>

<form type="inflected">jupe-
  <gram xml:lang="KAZ-KAM">vor Possessivsuffix (3SG)</gram>
</form>

3.8. On the Road to a Dialect Dictionary of Khanty Postpositions [page 104 of 131] [http://dx.doi.org/10.7557/5.3472]
Once the forms of the postposition are enumerated, the meanings can be worked out. Each meaning is provided not only with details about the subdialect in which it occurs, but also with information about the combinatorics of the meaning, its sources in the two aforementioned dictionaries or my collection. If documented in the corpus, the meaning is illustrated by sample sentences and their metadata:

```xml
<sense n="1">
  <gramGrp>Mit Nomen</gramGrp>
  <lang>Š</lang>
  <lang>KazSt.</lang>
  <lang>Sy.</lang>
  <lang>PB</lang>
  <def>nach, hinter</def>
  <ref>DWS 328</ref>
</sense>

<sense n="1a">
  <gramGrp>Mit Personalpronomen</gramGrp>
  <lang>Kaz.</lang>
  <lang>Sy.</lang>
  <lang>KAZ_KAM</lang>
  <lang>KAZ_KAO</lang>
  <def>hinter jemandem</def>
  <eg xml:lang="KAZ-KAO">לג געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן געפ=ן גع
  <eg xml:lang="GER">Hinter ihm, wenn er jagen geht, geht sein Hund.</eg>
  <bibl>ETM 202/028</bibl>
</def>
  <ref>DWS 328</ref>
  <ref>SZS</ref>
</sense>
```
The entry is concluded either with the last meaning or with possibly already grammaticalised phrases:

```xml
<dictScrap>
  sʲi ~
  <desc>danach, dann</desc>
  <lang>Š</lang>
  <lang>PB</lang>
  <lang>KAZ_KAM</lang>
  <lang>KAZ_KAO</lang>
  <lang>SHU_POS</lang>
  <lang>SHU_SYN</lang>
  <ref>DWS 328</ref>
  <ref>SZS</ref>
</dictScrap>
</entry>

3 Further steps

The material is now being transferred into a web application combining the XML TEI P5 documents of the lexicon and the MySQL data of the corpus. Each headword will come with the full lexicon entries as described above, with hyperlinks from each (sub)dialect, category of form, usage and meaning to all corresponding occurrences in the corpus, along with sound files, annotations and metadata. Aside from a simple search over the most common attributes, an advanced search mode will allow for a variety of combinations of search terms and clauses. A contact form will be provided for feedback and suggestions.

References


[3] Ф.М. Лельхова. Шурышкарский диалект (сынский говор) в системе диалектов хантыйского языка. In Актуальные проблемы разработки учебно- metodических комплексов по хантыйскому и мансийскому языкам;


FinUgRevita: Developing Language Technology Tools for Udmurt and Mansi

Veronika Vincze¹, Ágoston Nagy², Csilla Horváth², Norbert Szilágyi³, István Kozmács³, Edit Bogár², Anna Fenyvesi²

¹University of Szeged, Department of Informatics
²University of Szeged, Institute of English-American Studies
³University of Szeged, Department of Finno-Ugric Studies

finugrevita@gmail.com

December 16, 2014

Abstract

Nowadays, digital language use such as reading and writing e-mails, chats, messages, weblogs and comments on websites and social media platforms such as Facebook and Twitter has increased the amount of written language production for most of the users. Thus, it is primarily important for speakers of minority languages to have the possibility of using their own languages in the digital world too. The FinUgRevita project aims at providing computational language tools for endangered indigenous Finno-Ugric languages in Russia, assisting the speakers of these languages in using the indigenous languages in the digital space. Currently, we are working on two Finno-Ugric minority languages, namely, Udmurt and Mansi. In the project, we have been developing electronic dictionaries for both languages, besides, we have been creating corpora with a substantial number of texts collected, among other sources like literature, newspaper articles and social media. We have been also implementing morphological analyzers for both languages, exploiting the lexical entries of our dictionaries. We believe that the results achieved by the FinUgRevita project will contribute to the revitalization of Udmurt and Mansi and the tools to be developed will help these languages establish their existence in the digital space as well.

This work is licensed under a Creative Commons Attribution–NoDerivatives 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by-nd/4.0/
1 Introduction

In the age of modern technology, the constant development and widespread usage of technical tools such as the internet and smartphones enable people to communicate in real time throughout the world. Human-human interaction and machine-human interaction is supported by several language technology tools and applications such as spellcheckers, machine translation websites and search engines, besides, online resources and databases are exploited in communication in the digital world. However, the fact that while there are effective language technology tools available for languages with millions or billions of speakers, for minority languages even the most basic digital language processing tools are often missing. Hence, it is of utmost importance to develop language technology tools for users of minority languages, in order to facilitate communication in their mother tongue in the digital world as well.

Minority languages differ from other languages not only with respect to the numbers of their speakers but with respect to the fact that they are usually not recognized as official languages in their respective countries, where there is an official language and one or more minority languages. Thus, it is often the case that the speakers of minority languages are bilingual, and usually use the official or majority language at school and at work, and the language of administration is also the majority language. On the other hand, the use of the minority language is typically restricted to the private sphere, i.e. among family and friends, and thus it is mostly used in oral communication, with only rare examples of writing in the minority language.

Nowadays, digital language use such as reading and writing e-mails, chats, messages, weblogs and comments on websites and social media platforms such as Facebook and Twitter has increased the amount of written language production for most of the users [1]. Thus, it is of primary importance for bilingual speakers to be able to use their mother tongues in the digital space as well (cf. [2]).

In order to implement user-friendly language technology applications such as the above-mentioned spellcheckers or machine translation systems, basic linguistic pre-processing technologies are a must for the given language. In the case of minority languages, natural language processing might encounter problems even at the level of character encoding, provided that there are no standardized or well-known character sets in use. For higher-level language technology applications, it is further necessary to have a sentence splitter and tokenizer, a morphological analyzer and part-of-speech tagger, moreover, to get a deeper understanding of the content of texts, syntactic and semantic parsers are indispensable. These tools are often used in a chain: for instance, the output of the tokenizer is the input of the morphological analyzer, and the syntactic parser usually makes use of the output of the POS-tagger when parsing sentences.

In this paper, we discuss work within our project, FinUgRevita, which seeks to...
create language technology tools for minority Finno-Ugric languages. We first describe the project, then we provide some basic background to the languages we are currently working on: Udmurt and Mansi. Later, we present the main tasks of the project, i.e. corpus building, developing electronic dictionaries and morphological analyzers. Lastly, we offer some possible directions for future work that we intend to do in the next phases of the project.

2 The FinUgRevita Project

The FinUgRevita project¹ aims at providing computational language tools for endangered indigenous Finno-Ugric languages in Russia, assisting the speakers of these languages in using the indigenous languages in the digital space, and assessing, with the tools of sociolinguistics, the success of these computational language tools. The project is supported by the Hungarian National Research Fund and the Finnish Academy of Sciences, and is carried out by researchers working at the University of Szeged and the University of Helsinki.

In the computational linguistic component of this project we plan to use existing language resources in endangered minority Finno-Ugric languages to develop computational tools (learning tools and authoring tools) that would enable speakers to use their minority language in modernized popular discourse required in common everyday functions of written language use. Another key goal of the project is to provide these tools free of charge to anyone who is interested in learning and practising these languages. The tools, we believe, will increase speakers’ proficiency in their minority language, positively change speakers’ attitudes to their minority language, and, in the end, aid the revitalization process.

3 The Languages: Udmurt and Mansi

Here we provide some background on Udmurt and Mansi and basic demographic data on their speakers.

3.1 Udmurt

The Udmurt language (or, by an earlier exonym, Votyak) is a member of the Uralic language family, a somewhat endangered indigenous language in Russia. It is spoken in the area between the Vyatka, Cheptsa and Kama rivers, about 1,200 kilometers

¹http://www.ieas-szeged.hu/finugrevita/index.html
(about 750 miles) east of Moscow but west of the Ural mountains, in the Udmurt Republic (or, informally, Udmurtia). Additionally, Udmurts also live in greater numbers in Kazakhstan, and dispersed in many cities and towns of Russia. According to the latest, 2010, Russian census, 552,299 people profess to be of Udmurt ethnicity and 324,338 to be speakers of the Udmurt language. (Both figures have been decreasing from census to census in recent decades.)

Today, the Udmurt language is used mostly within the family and among friends, and even though it is an official language in Udmurtia, it has limited power and rights. It is not used in the legislature or political life. However, it is present in the media, education, and the cultural sphere, as well as enjoying a growing presence on the internet.

3.2 Mansi

The Mansi language (or, by an earlier exonym, Vogul) is a member of the Uralic language family, a severely endangered indigenous language in Russia. It is spoken primarily in the Khanti-Mansi Autonomous Okrug of Western Siberia. According to the latest, 2010, Russian census, 12,269 people profess to be of Mansi ethnicity and 938 to be speakers of the Mansi language. (The former figure has been increasing from census to census in recent decades, while the latter decreasing.)

Today, the Mansi language is used mostly within the family and among friends. It has no official status or economic value associated with it. It is not used in the legislature or political life. However, it is present in the media, education, and the cultural sphere, as well as enjoying a growing presence on the internet.

4 A Survey of User Data: The Case of Saami

At the beginning of our project, we contacted the maintainers of the website Giellatekno², which offers many important CL resources and tools for several minority languages including various dialects of Saami, Circumpolar and Uralic languages. They kindly provided us their access logs, on the basis of which we were able to carry out some quantitative data analysis in order to gain some insight into what user preferences are when using CL resources and tools for minority languages.

First, we analyzed dictionary searches made in Giellatekno’s database. It was revealed that the most frequently searched language pairs are Northern Saami – Norwegian and vice versa, Northern Saami – Finnish and vice versa, Finnish Kven – Norwegian, Nenets – Finnish and Western Mari – Finnish. The users usually seek to
translate words from Northern or Southern Saami, Finnish Kven or Nenets, on the other hand, the languages they would like to translate into are usually Norwegian, Finnish or English. All this suggests that most users translate from a minority language to a majority language (or a widely known second language like English), with the exception of Saami dialects, where both translation directions are widely attested. The number of page visits also demonstrates that online dictionaries play an essential role in learning minority languages. With this in mind, we felt it necessary to set ourselves the goal of creating online dictionaries for both languages we are working with (see Section 5.1 for details).

Second, we also analyzed the demographic data of the users of the page. We were also given access to the Google Analytics of the Giellatekno sites. Most of the users of the GT site still use Norwegian (Bokmål) on their computers. In the last month (Oct 2014), 10,000 people connected to the site, and more than 6,000 of them use Bokmål, while the second most important language is English with 1,300 users, and the third is Finnish with 1,000 users.

Google Analytics also provide data about the location of the access. These are in line with the language data: most of the users connect to the site from Norway (8,000), the second one is Finland with 1,400 users and the third is Sweden with nearly 600 users. All this proves that existing online resources for Finno-Ugric languages raise the interest of users across linguistic and geographic boundaries, which tendency we would also like to exploit in our project, that is, we intend to make our resources freely available on the web.

5 FinUgRevita’s Contributions

In this section, we present the FinUgRevita project’s most important contributions to the computational linguistic field, which cover the digitization of existing resources and the implementation of new tools and resources as well.

5.1 Creating online dictionaries

The creation of online electronic dictionaries is in progress for the two main languages of the project, Mansi and Udmurt.

The original paper-based Udmurt–Hungarian dictionary we are using as a starting point was compiled and edited by István Kozmács ([3]). In the project, the electronic version (Microsoft Word document) of this book is used and is transformed for our needs semi-automatically. First, the document is transformed into a simplified HTML containing the main text style character markers (like **bold** or *italics*). On the basis of
this formatting, the whole document is converted into a CSV file (comma-separated values) automatically, but this has to be reviewed manually since a paper-based dictionary contains some shortcuts which do not enable its automatic processing, for instance, it contains coordinations that can only be interpreted by humans. At this stage, the automatic conversion has been already carried out, and the manual correction phase is in progress. The dictionary contains approximately 13,000 entries.

The project’s online Mansi dictionary is going to be based primarily on the already existing Mansi–Russian and Russian–Mansi dictionaries, compiled by Mansi scholars. The online dictionary covers the lexical material of Rombandeeva’s and Kuzakova’s dictionary [4], and Rombandeeva’s Russian–Mansi dictionary [5], collated with the data of Munkácsi’s enormous Mansi–Hungarian dictionary [6] and also expanded with the Northern Mansi material of Balandin’s and Vakhrusheva’s Mansi–Russian dictionary [7], as well as with dozens of the most necessary neologisms describing different features of contemporary lifestyle (such as the urban environment, oil mining or judicial terms), created and used first and foremost by the journalists of the Mansi newspaper *Luima Seripo*.

The beta version of the online Mansi dictionary will contain approximately 10,000 entries. The Mansi lexemes will be supplemented with English, Russian and Hungarian translations, parts of speech and annotation of the sources, i.e. the dictionaries that are contained within. The Mansi forms are retrieved from the PDF versions of the dictionaries by means of optical character recognition, while the English and Hungarian translations are provided by linguists. Figure 1 presents the process of dictionary building: the automatic optical character recognition is followed by manual correction and translation of the entries, and then this database is turned into a searchable, digitized dictionary [8].

The online Mansi dictionary being a key resource for creating a morphological analyzer, the project also aims to make it available for public use as well, thus meeting a long-felt need for a sufficient Mansi–English–Mansi and a suitable online Mansi dictionary.

### 5.2 The Development of Morphological Analyzers

One of the most important tasks of this project is to create morphological analyzers. First, morphological analyzers for the Finno-Ugric languages we are working on were searched for and their usability was evaluated.

For Mansi, we were able to find a morphological analyzer [9] developed by MorphoLogic Ltd.³. However, it was not applicable to our purposes for several reasons.

First, it employs Latin-based transcription but the current Mansi orthography is Cyrillic-based (see Section 5.3). Second, its vocabulary completely lacks the contemporary lexicicon of the 20th and 21st centuries since it is based on Munkácsi’s Mansi dictionary [6] and it was optimized for the texts covered in Kálmán’s *Chrestomathia Vogulica* [10] and *Wogulische Texte* [11], mostly collected at the end of the 19th century. Third, it is not open-source. For all these reasons, we decided to create a new morphological analyzer for Mansi from scratch. The dictionary mentioned in Section 5.1 will serve as a basis for the morphological analyzer as well, and lexical entries of Mansi are now being grouped into different morphological categories depending on the conjugational/inflectional paradigm they belong to. For this, we rely on the descriptions found in several Mansi grammars [12, 13], as well as on the linguistic intuitions of native speakers of Mansi.

In the case of Udmurt, we contacted the developers of the already existing Udmurt analyzer available at http://giellatekno.uit.no/cgi/d-udm.eng.html. We collaborate now with them and our task is mainly to correct and to create the lexical database and the grammatical rules behind the analyzer. The lexical material
of our Udmurt dictionary mentioned in Section 5.1 is also being integrated into the database of the morphological analyzer.

### 5.3 Corpus Building

In order to create and test the applications to be made in the project, corpora of Mansi and Udmurt are being created. The corpora contain mainly newspaper articles and literature, but other types of texts are also planned to be integrated. Now, raw texts are collected, and later these texts will be transformed into a uniform structure and annotated.

Table 1 summarizes the number of words and characters in each discourse type of the Udmurt corpus. As can be seen, the biggest represented text type is the newspaper section with the published available volumes of the Udmurt language periodical *Udmurt Dunne*, but material from some children’s journals like *Kizili* and *Zechbur* and other newspapers are also included here. Topics vary from interviews to sports and cultural news, reports on events etc.

We were also able to collect material from the web, i.e. Wikipedia pages and weblogs, due to the growing presence of the Udmurt language in the social media as well. We also included some academic essays in the corpus, together with texts on education. Most of these texts were already digitized, which made it easier for us to collect and process them. The corpus now contains approximately 70,000 tokens.

The core of the Mansi corpus consists of the articles published in the Mansi newspaper *Luima Seripos*. The editorial staff of *Luima Seripos* (Mansi for “Northern dawn”) separated from the regional minority newspaper and started the Mansi monolingual newspaper on 11 February 1989. The length of the newspaper started from two pages, appearing twice a month, then increased to eight pages per week, and it has recently been published on sixteen pages every two weeks. The online archive of *Luima Seripos*, consisting of 46 issues, is available on the homepage of the joint editorial board.
of *Luima Seripos* and regional Khanty newspaper *Khanty Yasang*. This database, together with several former issues, increases the project’s Mansi corpus up to 260 exemplars, that is, to approximately 5,200 articles. The corpus now contains more than 1 million tokens.

The Mansi texts published in *Luima Seripos* cover various topics, most importantly not only those introducing traditional lifestyle, folklore and short biographies, but domains of urban life as well, thus they provide the project with a multilayered and diverse corpus. Since the Mansi newspaper is the only stable and complex source of Mansi texts, of all the possible sources it has the greatest impact on the language use of the Mansi population.

Using *Luima Seripos* as the primary source of the Mansi corpus also defines the project’s choice for Mansi orthography. The first researchers visiting the Mansi used different Latin-based transcriptions to write down Mansi texts, and the first attempts to create the standard variety and orthography for the Mansi language at the beginning of the Soviet era were based the Latin alphabet as well. Cyrillic transcription came into use in 1937 when all the nationalities living in the Soviet Union were ordered to switch over to the use of Cyrillic-based alphabets. The change caused several problems and the unsuitability of the Cyrillic alphabet and orthographical system to represent the morpho-phonological features of the Mansi language was not the smallest among them. The newspapers, schoolbooks and other works published in Mansi were inconsistent in marking special phonemes (such as the grapheme ӈ denoting the phoneme ŋ), or vowel length (despite of its role in differentiating the meaning of words, e.g. *оc* ‘surface’ and *о̄c* ‘sheep’). Nowadays the Mansi writing system is almost completely unified [14], the only minor difference between the two currently used orthographies is marking the palatal fricative: while scientific works use a combination of letters c and palatalizing vowels, in non-scientific publications, such as the *Luima Seripos* newspaper, and, for instance, schoolbooks in alternative educational institutions the authors replace c with щ.

### 6 Summary

In this paper, we have discussed the FinUgRevita project, which seeks to provide language technology tools for two Finno-Ugric minority languages, namely, Udmurt and Mansi. Currently, we have been developing electronic dictionaries for both languages, besides, we have been creating corpora with a substantial number of texts collected, among other sources like literature, newspaper articles and social media. We have
been also implementing morphological analyzers for both languages, exploiting the lexical entries of our dictionaries.

Our future plans involve several tasks. First, we intend to make our dictionaries and morphological analyzers freely available for the speakers of Udmurt and Mansi and for anyone else interested in them. Second, we want to annotate our corpora with morphological and possibly syntactic information, which might serve as training data for statistical POS-taggers and syntactic parsers. Third, we also want to create online linguistic games that might help the process of language learning. We believe that the results achieved by the FinUgRevita project will contribute to the revitalization of Udmurt and Mansi and the tools to be developed will help these languages establish their existence in the digital space as well.

Acknowledgments

This work was supported in part by the Finnish Academy of Sciences and the Hungarian National Research Fund, within the framework of the project Computational tools for the revitalization of endangered Finno-Ugric minority languages (FinUgRevita). Project number: OTKA FNN 107883; AKA 267097.

References


Automatic creation of bilingual dictionaries for Finno-Ugric languages

Eszter Simon
simon.eszter@nytud.mta.hu

Ivette Zs. Benyeda
benyeda.ivett@nytud.mta.hu

Péter Koczka
koczka.peter@nytud.mta.hu

Zsófia Ludányi
ludanyi.zsofia@nytud.mta.hu

Research Institute for Linguistics, Hungarian Academy of Sciences

December 15, 2014

Abstract

We introduce an ongoing project whose objective is to provide linguistically based support for several small Finno-Ugric digital communities in generating online content. To achieve our goals, we collect parallel, comparable and monolingual text material for the following Finno-Ugric (FU) languages: Komi-Zyrian and Permyak, Udmurt, Meadow and Hill Mari and Northern Sami, as well as for major languages that are of interest to the FU community: English, Russian, Finnish and Hungarian. Our goal is to generate proto-dictionaries for the mentioned language pairs and deploy the enriched lexical material on the web in the framework of the collaborative dictionary project Wiktionary. In addition, we will make all of the project’s products (corpora, models, dictionaries) freely available supporting further research.

1 Introduction

In his survey on language death, Kornai [1] states that language has become a function that is performed digitally, and that a language is digitally viable only to the extent it produces new, publicly available digital material. Language death implies loss of function, entailing the loss of prestige, and ultimately the loss of competence. To avoid such deterioration, our project aims to support Finno-Ugric language communities so...
that they would be able to cope with some of the digitally performed functions of their native languages.

In this context, language technology aspires to become an enabler technology that helps people collaborate, conduct business, share knowledge and participate in social debate regardless of language barriers and computer skills [2]. However, cutting-edge technologies are typically available only for widely-spoken languages which are in the class of digitally thriving languages according to Kornai’s classification [1].

In this paper, we introduce an ongoing project whose objective is to provide linguistically based support for several small Finno-Ugric digital communities in generating online content and help revitalize the digital functions of some endangered Finno-Ugric languages. The project is based on comparable corpora collected from the web. We generate proto-dictionaries for several endangered Finno-Ugric and major language pairs and deploy the enriched lexical material on the web in the framework of the collaborative dictionary project Wiktionary.

The first major component of the research project is the compilation and development of parallel and comparable corpora. We collect text material for the following Finno-Ugric (FU) languages: Komi-Zyrian and Permyak, Meadow and Hill Mari, Udmurt and Northern Sami, as well as for major languages that are of interest to the FU community: English, Russian, Finnish and Hungarian, see Section 3.1.

The parallel and comparable corpora will be automatically pre-processed. Since these small FU languages have weak language technology support, we experiment with language-independent tools applying machine learning methods, see Section 3.2.

Having the data pre-processed, we conduct experiments with automatic dictionary generation both from parallel and comparable texts. As a result, we have bilingual proto-dictionaries containing more than one thousand translation candidates for each language pair, see Section 4.

Dictionary entries will be automatically enriched with linguistic information and manually corrected by native speakers then uploaded to Wiktionary, see Section 5.

2 Related work

Bilingual dictionaries play a critical role not only in machine translation [3] and cross-language information retrieval [4], but also in other NLP applications, like language learning [5], computational semantics and several tasks requiring reliable lexical semantic information [6]. Since manual dictionary building is time-consuming and takes a significant amount of skilled work, it is not affordable in the case of lesser used languages. However, completely automatic generation of clean bilingual resources is not possible according to the state of the art. As a middle course, rough equivalence
at the conceptual level is already a useful notion, and filtering out candidate translation pairs produced by standard bilingual dictionary building methods can support lexicographic work.

The standard dictionary building methods are based on parallel corpora. However, as foreseen by Rapp [7], “the availability of a large enough parallel corpus in a specific field and for a given pair of languages will always be the exception, not the rule”, such corpora are still available only for the best-resourced language pairs. This is the reason of the increased interest in compiling comparable (non-parallel) corpora.

The standard approach of bilingual lexicon extraction from comparable corpora is based on context similarity methods (e.g. [7, 8]), which consist of the following steps: building context vectors, translation of context vectors, and comparison of source and target vectors. These methods need a seed lexicon which is then used to acquire additional translations of the context words. One of the shortcomings of this approach is that it is sensitive to the choice of parameters such as the size of the context, the size of the corpus, the size of the seed lexicon, and the choice of the association and similarity measures. Since there are no sufficiently large corpora and lexicons for these FU languages, conducting experiments with alternative methods is needed. There are several newer approaches to extracting translation pairs from non-parallel corpora, e.g. independent component analysis [9], label propagation [10], and topic model based methods [11]. One of the hot topics in NLP is using deep learning algorithms for obtaining vector representations for words, which are applied for a wide range of NLP tasks, as well as for extracting translation candidates from large amounts of unstructured text data (e.g. [12]). Yet another method for lexicon building is extracting the real parallel sentences from comparable corpora (e.g. [13]), which are then used as standard parallel texts for generating proto-dictionaries.

3 Creating parallel and comparable corpora

As a first step, we collected parallel and comparable texts for the language pairs in question. Linguistic processing of the collected data is inevitable before the next steps of dictionary building. Since all lexicon building methods require sentence-level aligned text, the running text must be split into sentences. Dictionary entries usually are words, thus word-level pre-processing, i.e. tokenization is also needed. Providing part-of-speech tag and lemma for each token is a very important step, since Wiktionary entries cannot be a finite word form without a lemma. In this section, we describe the subtasks of the corpus building workflow, including text collection and text processing steps.
3.1 Text collection

To build parallel corpora, we collected source texts and translations in parallel. In a strict sense, only Bible translations, novel translations, software documentation, and official documents, such as the Universal Declaration of Human Rights, can be treated as real parallel texts. For building comparable corpora, multilingual text collections have been created by applying several approaches.

Parallel corpora. Using the Bible as a parallel text in dictionary building has a long tradition [14]. To the extent feasible we tried to use modern Bible translations to avoid extracting archaic or extinct words. We downloaded the New Testament in the investigated FU and major languages from the Parallel Bible Corpus [15], Bible.is and The Unbound Bible. The translations are provided in a verse-aligned plain text format, thus they can be easily used for further processing.

Additionally, we found Northern Sami software documentation aligned with all major languages in question in the OPUS corpus [16] and some parallel texts on the websites of officially bilingual regions of Norway, Finland, and Russia. We did not find any parallel texts, not even Bible translations, for the language pairs where L1 is Komi-Permyak, Hill Mari or Udmurt.

Comparable corpora. For creating comparable corpora, the most commonly used source is Wikipedia. First of all, we downloaded the Wikipedia dumps for the languages we are dealing with and extracted each interlanguage-linked article pair. We used a slightly modified version of Wikipedia Extractor¹ for extracting the plain text, some metadata and in-text inter-wiki links with Wikidata IDs. Wikidata is a sister project of Wikipedia: it is a free collaborative multilingual knowledge base where inter-linked Wikipedia titles are instances of one and the same entity with one Wikidata ID. Using these IDs will help us to find and anchor named entities in the text of articles – regardless of language.

The length of the corresponding Wikipedia articles can be highly different: articles that are being maintained by a large, active digital community are typically fully-fledged, whereas articles from the language domain of a small community can be very brief. To improve the comparability measure, we consider only the first \(x\) sentences of each article in the major languages, where \(x\) is the number of sentences in the corresponding FU article, supposing that they are roughly each other’s translation (corresponding to the first, defining paragraph).

Another approach to building a comparable corpus is downloading domain-specific monolingual texts by specifying a keyword [8]. We collected documents about the Sami culture, education and society in English and Northern Sami from several websites.

¹http://medialab.di.unipi.it/wiki/Wikipedia_Extractor
Yet another way of collecting comparable text material is downloading multilingual daily newspaper materials from the same time interval and, if feasible, from the same country or region. Since these articles are about the most important local and global events, even if they are not translations of each other, they can be treated as comparable corpora [8]. Based on this hypothesis, we collected articles from online newspapers in Finland for Northern Sami–{Finnish, English, Russian} language pairs.

**Monolingual texts.** We also collected monolingual texts for all FU languages. While parallel and comparable corpora are used to create dictionaries, monolingual texts serve as training data for the tokenizer and sentence splitter. We collected monolingual data from several websites in various domains, e.g. literature, news, personal blogs, official texts.

Table 1 shows the number of tokens for parallel, comparable and monolingual corpora. Numbers for the latter comprise all data, even the texts included in parallel and comparable corpora. Token numbers for comparable corpora contain only the restricted size Wikipedia articles, i.e. only the first *x* sentences of each article pair (see above). The time interval-based texts have been compared on a yearly basis, i.e. news files which do not have a corresponding pair from the same time period were not considered. The numbers in the table show the current state of the text processing task; they will indeed change at later phases of the project.

### 3.2 Text processing

The pre-processing and segmentation is a particularly important step, largely because any error made at this stage is likely to cause complications at later phases of text processing. However, these small FU languages have weak language technology support. An outstanding contributor is Giellatekno², which provides NLP tools for several FU languages. Northern Sami by far has the largest available online resources, but, as far as we know, tools for the other small FU languages are still under development.

**Pre-processing steps.** As mentioned in Section 3.1, we gathered a relatively large amount of text, but there are a number of issues with the collected material, since an easily observable portion can be classified as “dirty text”.

First, all texts must undergo character normalization before any further processing steps. All sources have been converted to plain text files using standard Unicode characters in UTF-8 encoding.

Second, closely related languages (Komi-Zyrian and Permyak, Meadow and Hill Mari) are often mixed even within a single document, thus they have to be separated. For language discrimination, we used the Blacklist Classifier³, which resulted in a

²[http://giellatekno.uit.no](http://giellatekno.uit.no)
<table>
<thead>
<tr>
<th>lang</th>
<th>mono</th>
<th>lang pairs</th>
<th>parallel L1</th>
<th>parallel L2</th>
<th>comparable L1</th>
<th>comparable L2</th>
</tr>
</thead>
<tbody>
<tr>
<td>sme</td>
<td>1,364,254</td>
<td>sme–eng</td>
<td>691,260</td>
<td>724,750</td>
<td>253,930</td>
<td>1,754,968</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sme–fin</td>
<td>245,440</td>
<td>273,973</td>
<td>239,651</td>
<td>5,259,591</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sme–rus</td>
<td>173,179</td>
<td>220,790</td>
<td>212,332</td>
<td>233,748</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sme–hun</td>
<td>171,668</td>
<td>224,014</td>
<td>86,244</td>
<td>106,391</td>
</tr>
<tr>
<td>kvp</td>
<td>480,609</td>
<td>kvp–eng</td>
<td>121,108</td>
<td>174,742</td>
<td>89,580</td>
<td>183,602</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kvp–fin</td>
<td>121,120</td>
<td>133,715</td>
<td>88,507</td>
<td>80,797</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kvp–rus</td>
<td>117,903</td>
<td>125,085</td>
<td>108,013</td>
<td>141,369</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kvp–hun</td>
<td>121,319</td>
<td>134,344</td>
<td>68,179</td>
<td>74,274</td>
</tr>
<tr>
<td>koi</td>
<td>719,325</td>
<td>koi–eng</td>
<td>0</td>
<td>0</td>
<td>257,871</td>
<td>194,784</td>
</tr>
<tr>
<td></td>
<td></td>
<td>koi–fin</td>
<td>0</td>
<td>0</td>
<td>137,578</td>
<td>77,696</td>
</tr>
<tr>
<td></td>
<td></td>
<td>koi–rus</td>
<td>0</td>
<td>0</td>
<td>188,334</td>
<td>139,976</td>
</tr>
<tr>
<td></td>
<td></td>
<td>koi–hun</td>
<td>0</td>
<td>0</td>
<td>95,120</td>
<td>64,794</td>
</tr>
<tr>
<td>mhr</td>
<td>1,335,457</td>
<td>mhr–eng</td>
<td>128,316</td>
<td>175,075</td>
<td>121,588</td>
<td>250,583</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mhr–fin</td>
<td>128,328</td>
<td>133,965</td>
<td>118,120</td>
<td>115,028</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mhr–rus</td>
<td>109,449</td>
<td>109,818</td>
<td>158,977</td>
<td>215,724</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mhr–hun</td>
<td>128,565</td>
<td>134,618</td>
<td>106,813</td>
<td>121,453</td>
</tr>
<tr>
<td>mrj</td>
<td>366,964</td>
<td>mrj–eng</td>
<td>0</td>
<td>0</td>
<td>137,088</td>
<td>306,465</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mrj–fin</td>
<td>0</td>
<td>0</td>
<td>85,134</td>
<td>93,622</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mrj–rus</td>
<td>0</td>
<td>0</td>
<td>124,289</td>
<td>187,687</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mrj–hun</td>
<td>0</td>
<td>0</td>
<td>77,855</td>
<td>90,168</td>
</tr>
<tr>
<td>udm</td>
<td>584,113</td>
<td>udm–eng</td>
<td>0</td>
<td>0</td>
<td>67,306</td>
<td>135,450</td>
</tr>
<tr>
<td></td>
<td></td>
<td>udm–fin</td>
<td>0</td>
<td>0</td>
<td>56,222</td>
<td>49,961</td>
</tr>
<tr>
<td></td>
<td></td>
<td>udm–rus</td>
<td>0</td>
<td>0</td>
<td>80,800</td>
<td>129,293</td>
</tr>
<tr>
<td></td>
<td></td>
<td>udm–hun</td>
<td>0</td>
<td>0</td>
<td>41,883</td>
<td>48,736</td>
</tr>
</tbody>
</table>

Table 1: Number of tokens for monolingual, parallel and comparable corpora. We use the ISO 639-3 language codes: sme – Northern Sami, kvp – Komi-Zyrian, koi – Komi-Permyak, mhr – Meadow Mari, mrj – Hill Mari, udm – Udmurt.
97.47% accuracy for Komi-Zyrian and Permyak and a 96.77% accuracy for Meadow and Hill Mari languages.

Third, majority of FU language bloggers use blog publishing services that only support Russian or English, therefore these texts are mixed, thus dates and some elements of websites are not in the desired FU language but in one of the languages supported by the service provider. To filter out foreign parts, we use Langid⁴, a language identifier using trigram statistics with Katz’s back-off smoothing. Models were created using manually selected text samples. Since dates are valuable information in order to build time frame-based comparable corpora, we preserved them.

**Sentence segmentation and tokenization.** For sentence segmentation and tokenization, we use the sentence detection and tokenizer tools of Apache OpenNLP⁵. Since Northern Sami is quite well-supported with NLP tools, we built models only for the FU languages using Cyrillic script. We created gold standard data for training and testing the tokenizer and sentence splitter modules of Apache OpenNLP. Over ten thousand sentences were randomly selected for each language, and, after manual correction, the data was divided into training and test sets (90%-10%). Both modules performed over 98% F-measure, which is partly due to the abbreviation dictionary support of Apache OpenNLP, blocking the false sentence segmentation at abbreviations. Using such lists is a common practice in sentence segmentation, however, building an exhaustive list is beyond the bounds of possibility, especially for the FU languages in the scope of our research. For this reason, our abbreviation dictionary is mainly based on the Russian abbreviation list of Wiktionary, but we plan to extend it with more abbreviations and acronyms found at later stages of our work. Nevertheless, using only Russian abbreviations would be sufficient for our needs as the FU languages written in Cyrillic script tend to use a number of Russian abbreviations (consider the abbreviations for units of measurement, place names and internationalisms).

**Morphological analysis and disambiguation.** Morphological analysers are available as online applications for Udmurt and Komi-Zyrian⁶ and Hill Mari⁷. The website of Giellatekno contains source files for a finite state transducer-based morphological analyser for almost all FU languages we deal with. However, as far as we know, morphological analyser for Komi-Permyak do not exist.

For languages that lack any morphological analysers, there are two possibilities we can choose from. Once, we can use semi-supervised or unsupervised morphological segmentation tools such as Morfessor⁸ and develop additional tools to extend

---

⁴https://github.com/juditacs/langid
⁵https://opennlp.apache.org/
⁷http://www.univie.ac.at/maridict/site-2014/morph.php
⁸http://morfessor.readthedocs.org/en/latest/general.html#techrep
its functionality to meet our needs. We made some experiments with Morfessor: we trained it on an Udmurt word list and compared its segmented output to the output of an Udmurt morphological analyser [17]. The results are convincing, but it would still take great effort to develop additional utilities that could produce lemmas and POS tags from Morfessor’s output.

The other option is to use existing tools developed for closely related languages. The most simple solution is the direct application of tools developed for the related language, thus the models built for Komi-Zyrian could be applied on the Komi-Permyak data directly. Moreover, we expect that morphological tags in Komi-Zyrian can be transferred to the Komi-Permyak version of the same text. Since large amounts of data for training do not exist for the majority of languages, experimenting with several methods of the annotation transfer between closely related languages is a hot topic in NLP (e.g. [18, 19]). We plan to investigate the approach for transferring POS annotations from a resourced language towards a closely related non-resourced language by Scherrer and Sagot [18].

4 Creating proto-dictionaries

Completely automatic generation of clean bilingual resources is not possible according to the state of the art, but it is possible to create certain lexical resources, termed proto-dictionaries, that can support lexicographic work. Proto-dictionaries are expected to provide greater coverage but comprise more incorrect translation candidates; their right size depends on the specific needs.

We made experiments with several lexicon building methods, which are detailed below. Applying each method resulted in bilingual resources containing translation candidates for almost all language pairs. These dictionary files will then be used as the starting point to create the final dictionaries, where only the most likely translation candidates are kept on the basis of some heuristics, developed in a later phase of the project by manually evaluating the results. At this stage of the project, we have raw, i.e. still un-cleaned proto-dictionaries for all language pairs each containing more than one thousand translation candidates. The most under-resourced language pair is Komi-Permyak–Hungarian with ca. 1300 word pairs, while the Northern Sami–Finnish proto-dictionaries contain more than 20,000 word pairs.

**Wikipedia titles.** Wikipedia is not only the largest publicly available database of comparable documents, but it also can be used for bilingual lexicon extraction in several ways. Erdmann et al. [20] used pairs of article titles for creating bilingual dictionaries, which were later expanded with translation pairs extracted from the article texts. Mohammadi and Ghasem-Aghaee [13] extracted parallel sentences from the
English and Persian Wikipedia using a bilingual dictionary generated from Wikipedia titles as a seed lexicon. Following this approach, we also created bilingual dictionaries from Wikipedia title pairs using the interwiki links.

**Wiktionary-based methods.** Besides Wikipedia, Wiktionary is also considered as a crowdsourced language resource which can serve as a source of bilingual dictionary extraction. Although Wiktionary is primarily for human audience, the extraction of underlying data can be automated to a certain degree. Ács et al. [21] extracted translations from the so-called translation tables. Since their tool Wikt2dict is freely available⁹, we could apply it for our language pairs. We parsed the English, Finnish, Russian and Hungarian editions of Wiktionary looking for translations in the small FU languages we deal with.

Ács [22] expanded the collection of translation pairs, discovering previously non-existent links between translations with a triangulation method. It is based on the assumption that two expressions are likely to be translations, if they are translations of the same word in a third language. With the triangulation mode of Wikt2dict, we could further expand our dictionaries.

**Hundict.** Hundict¹⁰ is an experimental project for bilingual lexicon extraction from parallel corpora. It extracts word pairs based on high co-occurrence in corresponding text segments, using the Sørensen-Dice co-efficient. The tool’s performance can be improved by adding a gold standard dictionary and a list of stopwords. We made some experiments with Bible translations for Northern Sami–Finnish and Komi-Zyrian–English language pairs, which resulted in word pairs along with their confidence measures. The system has more parameters which can be fine-tuned, and we plan to test it with more options and on more language pairs. However, the tool needs lemmatized text as an input, thus we have to lemmatize our parallel corpora before further experiments.

5 Conclusion and future work

We introduced an ongoing project which is based on parallel and comparable corpora collected from the web. The project’s main objective is to generate dictionaries for language pairs where the source language is one of the following small FU languages: Komi-Zyrian and Permyak, Meadow and Hill Mari, Udmurt and Northern Sami, while the target language is one of the following major languages: English, Finnish, Russian and Hungarian. However, collecting text for these under-resourced FU languages and processing them poses several problems. Most of these FU languages are digitally not

---

⁹https://github.com/juditacs/wikt2dict
¹⁰https://github.com/zseder/hundict
really viable, since they produce very few digital text material. For this reason, finding large amounts of text in these languages is challenging. Since there are language pairs for which we did not find any parallel text material, standard dictionary building methods cannot be used. Moreover, these small FU languages have weak or no language technology support, thus language-independent supervised tools are needed to be used. On the level of morphology we faced other kind of problems: some analysers are available only as an online application, some are still under development.

In spite of the difficulties, we collected some text material for these languages and built proto-dictionaries by applying several methods. The final dictionaries will be uploaded to Wiktionary, where lexical entries contain morphological, etymological and lexico-semantic information, and translation equivalents across languages. We will generate the dictionaries and the linguistic information as automatically as possible. The manual validation and correction of Wiktionary input files will be conducted only in the last phase of the project by native speakers.

Using the Wiktionary infrastructure, lexical entries across the language versions of Wiktionary can be interlinked. This will enable user communities to access rich, networked lexical material that can be used for translation purposes. Content in Wiktionary is formatted in a lightweight markup system, but we will recast the data into an XML format suitable for further processing. After cleaning the copyright issues, we will make all of the generated resources (corpora, dictionaries, models) freely available.

Acknowledgments

The research reported in the paper was conducted with the support of the Hungarian Scientific Research Fund (OTKA) grant #107885. We thank the anonymous reviewers for their constructive comments.

References


