Proceedings of the Language Technologies for All (LT4All) , pages 219–222
Paris, UNESCO Headquarters, 5-6 December, 2019.
© 2019 European Language Resources Association (ELRA), licenced under CC-BY-NC

Language technology for Indigenous Languages: Achievements and Challenges

Sjur N. Moshagen, Trond Trosterud, Lene Antonsen
UiT The Arctic University of Norway
Tromsø, Norway
{sjur.moshagen, trond.trosterud, lene.antonsen}@uit.no

Abstract

Fifteen years of indigenous language technology development by UiT/Saami Parliament has resulted in spelling and grammar checkers, desktop/mobile keyboards, morphological analysers, MT, speech synthesis, language learning tools and intelligent electronic dictionaries. This was facilitated by an open source language independent infrastructure, targeted at languages with rich and complex grammar, with integration for host operating systems and apps. The current primary challenge is integration with closed platforms where we cannot currently support user needs. Our proposed solution is a "Manifesto for Open Language Technology", where APIs, localisations and source code are open, while ensuring community intellectual property custodianship, engagement and commitment.

Keywords: language technology, working solutions, morphology-rich languages

Čoahkkáigeassu (in North Saami)

UiT/Sámedikki 15 jagi eamiálbmot giellateknologiija barggu bohtosat leat sátne- ja grammatihkkadivvunprográmmat, boallobeavdi dihtorii ja mobiltelefovdnii, morfologalaš analysáhtorat, dihtorjorgaleapmi, hállansyntesa, giellaoahppanreaiddut ja intelligeanta digitála sátnegirjjit. Dát lea huksejuvvon rabas gáldokoda infrastruktuvrras, mii lea heivehuvvon gielaide main lea rikkes ja kompleaksa grammatihkka – infrastruktuvra mii siskkilda geavahanlavttaid ja applikašuvnnaid. Dál váldohástalus lea integreret prográmmaid giddejuvvon geavahanvuogádagaide, maid siste mii dál eat beasa doarjut geavaheddjiid dárbbsuid. Min evttohus lea "Rabas giellateknologiija manifesta", mas API:t, lokaliseren ja gáldokoda leat rabas, muhto seammás giellaservodagat galget hálddašit gáldokoda intellektuealla rivttiid.

1. Introduction

All indigenous languages of the world, except the Polynesian ones, are morphologically very complex languages. This means that one and the same word may show up in tens, hundreds or even thousands of forms. At the same time, most indigenous languages have a short written tradition and possess very small text collections, where the number of words in available running text cannot even be numbered in the thousands, let alone millions. Also, the text material there may often represent inconsistent or conflicting literary norms, and be of little use to language technology.

In this paper we present our the language technology infrastructure used to build LT tools for indigenous languages of the High North, languages with a rich and complex morphology. We also present a model for cooperation on the huge work behind language technology solutions that overcomes the problems posed by the weak commercial potential in such work. We will use the Saami language family as an example, but our model can be–and as a matter of fact has been–scaled to other indigenous languages as well.

The article is organised as follows: Section 2 gives some background of the Saami languages. Then we present our language technology and the methods used, an overview of remaining challenges, concerning problems of integrating indigenous language tools in mainstream computer platforms and programs. Finally comes a conclusion and a view on further work.

2. Background

In our work on language technology solutions we have focused on the Saami languages. Counting 9 separate languages, 8 of which have an official orthography, the Saami languages constitute the westernmost branch of the Uralic languages. The languages are spoken in the Mid and Northern part of the Scandinavian peninsula, the northern part of Finland, and the Kola peninsula. The largest of the languages is North Saami, with more than 25,000 speakers. All the other languages have less than 1000 speakers. South, Lule, Inari, Skolt and Kildin Saami have several hundred speakers, whereas Pite and Ume Saami have less than one hundred.

Typologically, the languages are unmistakably Uralic. They are suffixing languages with a rich nominal and verbal inflection, including person/number inflection on both verbs and nouns, different verb modes, as well as numerous derivational processes within and between the main parts of speech. Contrary to most Uralic languages they also have a rich variety of stem-internal morphophonological processes accompanying the suffixation, resulting in each paradigm possessing several inflectional stems. These processes includes the whole lexicon, and affects both root vowels and consonants, as well as stem consonants and suffix classes. The net result is that neither word form based approaches nor a system of stemming (suffix removal) is going to work.

Orthographically, each language has its own orthographical convention. Four of them (South, Ume, Pite

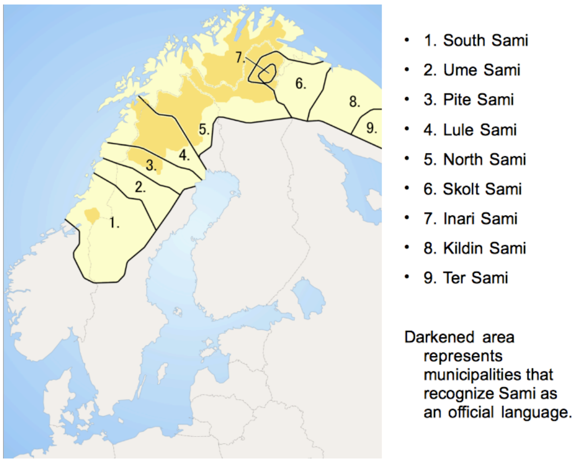


Figure 1: The Saami languages, and the municipalities where they have official status.

and Lule) build upon a tradition of writing the consonants according to the prevailing Scandinavian digraph tradition, and South Saami even has some vowel qualities similar to Scandinavian. The North, Inari and Skolt orthographies ultimately go back to a 200-year old tradition of one letter per phoneme, thus possessing a large repertoire of diacritical marks, each orthography still with its own conventions. Finally, Kildin Saami is written with the Cyrillic alphabet.

There are electronic text collections available for 5 Saami languages. North Saami is the largest, with 33M words, 3.5M parallel North Saami - Norwegian. The 4 languages all have less than 2M, from 200k Skolt Saami to 1.7M for Inari Saami. For the other Saami languages there are no online corpora available. (SIKOR, 2019). For an overview and a linguistic introduction to the language family, see (Sammallahti, 1998).

3. Our achievements

Fifteen years of indigenous language technology development by UiT/Saami Parliament has resulted in machine-readable grammars for most circumpolar literary languages, in the form of bidirectional models, capable of analysing and generating every word form of the language. These models in turn are used as key components in a wide array of tools, including spelling and grammar checkers[1], desktop and mobile keyboards[2], morphological and syntactic analysers[3], Machine translation[4], speech synthesis[5], language learning tools[6] and intelligent electronic dictionaries[7].

All the tools are in extensive use by the language communities. The spell checkers have been downloaded by approximately 2/3 of the language communities. On average, the e-dictionaries are used 12 times a week per speaker. The MT programs are in use in different contexts. Most notably, the Saami University College use our MT program to translate their web pages – being in North Saami only – into Norwegian.

The tools are discussed in several publications, a.o. on e-dictionaries (Johnson et al., 2013), spell checkers (Antonsen, 2018), grammarchecking (Wiechetek et al., 2019), Machine translation (Antonsen et al., 2017), and e-learning (Antonsen and Argese, 2018).

The grammar models we have made are made as bidirectional *finite state transducers*, as described in (Beesley and Karttunen, 2003). Grammatical ambiguities and syntax analysis we have resolved by means of Constraint Grammar (Karlsson, 1990).

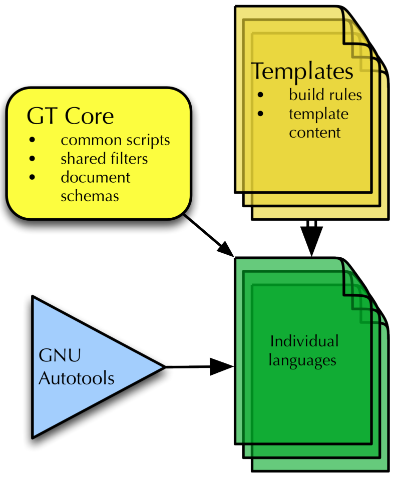


Figure 2: Our language independent infrastructure.

The challenge of scaling the work to several languages was handled by developing an open source, language independent infrastructure (see figure 2). It targets languages with rich and complex grammar, separating out the language specific work from language independent build and testing routines, and language agnostic integration into host operating systems and apps. In this way we have been able to reuse the work spent on integrating North and other Saami languages in office suites and other language processing software, thereby making such solutions available for language communities that simply do not possess the resources to achieve this by themselves.

A central part of our model is that it involves different stakeholders with different interests. For every language with an orthography, there is a community of university and field linguists devoting their career to it. For them, participating in making a machine readable model of the grammar is a way of empirically testing whether their hypotheses hold through. Philologists and lexicographers will find a way of formalising and testing their view on the vocabulary of the language. For language revivers and activists, the possibility of getting language technology tools is a central part of their strategy. The final part of this setup is the in-

[1] divvun.no/korrektur/korrektur.html
[2] divvun.no/keyboards.index.html
[3] giellalt.uit.no/lang/index.html
[4] gtweb.uit.no/mt/
[5] divvun.no/tale/tale.html
[6] https://oahpa.no/
[7] dicts.uit.no

frastructure for going from language model to practical program, and this infrastructure is provided by us as open source. The infrastructure is presented in (Moshagen et al., 2013).

4. Challenges

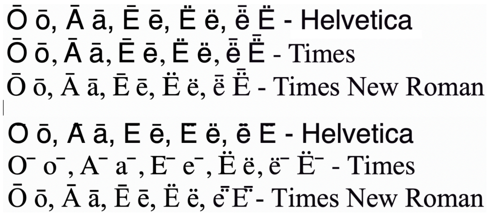
Currently, the primary challenge is integration with closed platforms where we cannot support user needs and meet their expectations. At present, language technology witnesses two conflicting trends. On one hand, more languages get some support or another by the major IT providers. On the other hand, the possibility of offering third-party solutions for language technology is restricted, often totally so.

In practice, **localisation of all major mobile operating systems is completely impossible**, and in practice also for all desktop operating systems and most apps. A language community has no way of defining or building their own digital presence, no tools to make their language visible and a natural part of the everyday language environment. This includes more things than the text on menus and buttons: it affects indexing and searching of text on computer systems, or hyphenation, needed for the long words resulting from complex morphology. For most of the world's languages it affects even such basic things as the name of the language: install a speller, and the name you get for that speller is not the actual language name, but a cryptic, three letter language code. How is an ordinary person meant to understand what that means?

Many operating systems, mobile and otherwise, provide a dictionary framework for adding dictionary content to the system. This is important for minority and indigenous languages. But often **those frameworks are not available to third parties**, or there is no way of adding lemmatisation or text analysis as part of the lookup process. For languages with complex morphology and phonology that is pretty much a blocker.

New web-based services and tools are a boon to many, but there is no support for languages outside the mainstream. A basic tool like a spell checker, which we have delivered for North Saami for 12 years now, is **suddenly locked out in new OS's like Chrome OS**, or web apps like Google Docs and Office 365. There is no way for us to provide it.

A great many indigenous and minority languages are using variants of the majority language alphabet, often with a lot of diacritics. The Unicode organisation has decided that they will not accept any new precomposed combinations of base characters plus diacritic, instead pointing to the mechanism for dynamic composition of diacritics. At the same time, this part of Unicode is not of great economical importance, since all majority languages are properly covered by precomposed letters in the standard. The end result is that **text written in minority and indigenous languages often becomes unreadable**, because the text rendering engines have bugs in them. The situation has been like this for more than 10 years (see figure 3).



Ō ō, Ā ā, Ē ē, Ë̄ ë̄, ē̈ Ë̄ - Helvetica
Ō ō, Ā ā, Ē ē, Ë̄ ë̄, ē̈ Ë̄ - Times
Ō ō, Ā ā, Ē ē, Ë̄ ë̄, ē̈ Ë̄ - Times New Roman

Ō ō, Ā ā, Ē ē, Ë̄ ë̄, ē̈ Ē - Helvetica
O¯ o¯, A¯ a¯, E¯ e¯, Ë ë, ë¯ Ë¯ - Times
Ō ō, Ā ā, Ē ē, Ë ë, ē̈Ē̈ - Times New Roman

Figure 3: Kildin Saami letters as they should look, and as they often look. Notice how the accents in Helvetica has been dragged down into the base letter.

All the rage these days are machine learning and artificial intelligence, mostly applied on MT and speech technologies. These are exciting new opportunities, and the enthusiasm is impossible to miss. But the enthusiasm is hard to share from a minority perspective, for two reasons. The first and obvious reason is the resource demands required. Although the newest technologies do require smaller corpora and less text than the previous generation of machine learning, the demands are still way out of reach for most languages. But even if it would be possible, there would be very little point in doing it, due to the second reason: **all major operating systems save macOS are closed for speech services**. Dialog systems and virtual assistants even more closed, and the languages are not even known to the operating systems. All in all: **speech technology tools can never be used in practice.** They are nice demonstrations, and perhaps add to the body of studied languages, but in terms of tools for the user community the possibilities are slim. Machine translation systems are more approachable, and there are programs for North Saami and a number of other languages. But often users wonder why Google Translate can't translate North Saami, and from a user perspective that is a legitimate question.

5. Our solution

How can we eliminate these technical hurdles? The issues being faced are not inherent to any language or writing system, but are **a consequence of technological considerations and economic incentives**. We propose a **"Manifesto for Open Language Technology"**, focusing on the following four points:

- **Open localisation:** all software should be localisable independent of the producer of the software
- **Open interfaces:** all language-related programming interfaces should be open by default
- **Open resources:** all language resources should be open and accessible for everyone, given the permission of the language community
- **Accessible standards:** all language-related international standards (ISO, etc) should be respected, fully implemented and implementations should be regularly updated

Immediate steps that can be taken by major vendors to **help us achieve indigenous self-determination in the digital realm** are as follows:

- **ISO 639 compliance** at the same rate as Unicode emoji compliance is supported in each major operating system
- Localisation packages for major operating systems should be installable from app stores and allow for **community-managed localisation**
- **Open up all language APIs currently held closed** on Windows, macOS, Android and iOS so that the community may integrate complex morphological tools with high quality user experience

The ultimate purpose it *not* that the major vendors implement language tools for us, but rather that there is a **guarantee of equal access to the APIs** that enable majority languages and allow them to be used for minority and indigenous languages.

We try to practice what we preach, by having all source code for the support infrastructure and tool integration on Github[8], building **tools in Rust to handle CLDR localisation data**, open source integration tooling for generating and maintaining **keyboards and locales for all major operating systems**, and developing continuous integration and continuous delivery infrastructure on top of the Azure platform.

The keyboard layout definitions and keyboard apps are also present on Github[9], with plans to migrate all remaining language technology source code from the currently used Subversion repository[10].

6. Conclusion

Indigenous languages need language technology made on their own terms. This may be implemented as a cooperation between university linguists and computational linguists, philologists and language activists, as well as programmers turning the language models into practical programs. In order for this to work the **software providers must open their software for third party providers**.

For minority and indigenous language communities, they need to have ownership over their own language, put their resources where they think it is most important, and **not be hindered by technical and economic decisions not related to their language at all**. It should be their decision whether they want their mobile phones and other devices to speak their language, not the decision of the vendor.

The ultimate goal is to achieve indigenous self-determination in the digital realm.

7. Acknowledgements

Thanks to our colleagues at Giellatekno and Divvun, as well as The Techno Creatives.

[8] https://github.com/divvun

[9] https://github.com/giellalt

[10] https://gtsvn.uit.no

8. Bibliographical References

Antonsen, L. and Argese, C. (2018). Using authentic texts for grammar exercises for a minority language. In *Proceedings of the 7th Workshop on NLP for Computer Assisted Language Learning (NLP4CALL 2018)*, pages 1–9. Linköping University Electronic Press.

Antonsen, L., Gerstenberger, C., Kappfjell, M., Rahka, S. N., Olthuis, M.-L., Trosterud, T., and Tyers, F. M. (2017). Machine translation with North Saami as a pivot language. In *Proceedings of the 21st Nordic Conference on Computational Linguistics, NoDaLiDa, 22–24 May 2017, Gothenburg, Sweden*, volume 29 of *NEALT Proceedings Series*, pages 123–131. Linköping University Electronic Press.

Antonsen, L. (2018). *Sámegielaid modelleren – huksen ja heiveheapmi duohta giellamáilbmái. [Modeling Saami languages. Construction and adaptation to real-world linguistic issues]*. Ph.D. thesis, UiT The Arctic University of Norway, Tromsø.

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology.* Studies in Computational Linguistics. CSLI Publications, Stanford, California.

Johnson, R., Antonsen, L., and Trosterud, T. (2013). Using finite state transducers for making efficient reading comprehension dictionaries. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013); May 22–24; 2013; Oslo University; Norway*, volume 16 of *NEALT Proceedings Series*, pages 59–71. Linköping University Electronic Press.

Karlsson, F. (1990). Constraint grammar as a framework for parsing running text. In *COLING '90 Proceedings of the 13th conference on Computational linguistics*, volume 3, pages 168–173, Helsinki.

Moshagen, S. N., Pirinen, T., and Trosterud, T. (2013). Building an open-source development infrastructure for language technology projects. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013); May 22–24; 2013; Oslo University; Norway*, number 16 in NEALT Proceedings Series, pages 343–352. Linköping University Electronic Press.

Sammallahti, P. (1998). Saamic. In Daniel Abondolo, editor, *The Uralic Languages*, pages 43—96. Routledge, London.

SIKOR. (2019). UiT The Arctic University of Norway and the Norwegian Saami Parliament's Saami text collection, with grammatical analysis. `http://gtweb.uit.no/korp/`.

Wiechetek, L., Moshagen, S., Gaup, B., and Omma, T. (2019). Many shades of grammar checking – launching a constraint grammar tool for north sámi. In Eckhard Bick et al., editors, *Proceedings of the NoDaLiDa 2019 Workshop on Constraint Grammar: Methods, Tools and Applications, Turku, Finland*, volume 33 of *NEALT Proceedings Series*, Linköping, Sweden. Linköping University Electronic Press.