Making the tools to connect isiXhosa and isiZulu to the digital age

March 13 2018, by Maria Keet

Credit: cottonbro studio from Pexels

We live in a world where around 7000 languages are spoken, and one where information and communication technologies are becoming increasingly ubiquitous. This puts increasing demands on more, and

more advanced, Human Language Technologies (HLTs).

These technologies comprise computational methods, computer programmes and electronic devices that are specialised for analysing, producing or modifying texts and speech.

Engaging with a language like English is made easier thanks to the many tools to support you, such as spellcheckers in browsers and autocomplete for text messages. This is mainly because English has a relatively simple and well investigated grammar, more data that software can learn from, and substantial funding to develop tools. The situation is somewhat to very different for most language in the world.

This is beginning to change. Profit driven multinationals such as Google, Facebook and Microsoft, for instance, have invested in the development of HLTs also for African languages.

Researchers and scientists, myself included are also investigating and creating these technologies. It has a direct relevance for society: languages, and the identities and cultures intertwined with them, are a national resource for any country. In a country like South Africa, learning different languages can foster cohesion and inclusion.

Just learning a language, however, is not enough if there's no infrastructure to support it. For instance, what's the point of searching the Web in, say, isiXhosa when the search engine algorithms can't process the words properly anyway and so won't return the results you're looking for? Where are the spellcheckers to assist you in writing emails, school essays, or news articles?

That's why we have been laying both theoretical foundations and creating proof-of-concept tools for several South African languages. This includes spellcheckers for isiZulu and isiXhosa and the generation

of text in mainly these languages from structured input.

Using rules of the language to develop tools

Tool development for the Nguni group of languages – and isiZulu and isiXhosa in particular – wasn't simply a case of copy-and-pasting tools from English. I had to develop novel algorithms that can handle the quite different grammar. I have also collaborated with linguists to figure out the details of each language.

For instance, even just automatically generating the plural noun in isiZulu from a noun in the singular required a new approach that combined syntax – how it is written – with semantics (the meaning) of the nouns by using its characteristic noun class system. In English, merely syntax-based rules can do the job.

Rule-based approaches are also preferred for morphological analysers, which split each word into its constituent parts, and for natural language generation. Natural language generation involves taking structured data, information or knowledge, such as the numbers in the columns in a spreadsheet, and creating readable text from them.

A simple way of realising that is to use templates where the software slots in the values given by the data or the logical theory. This is not possible for isiZulu, because the sentence constituents are context-dependent.

A grammar engine is needed to generate even the most basic sentences correctly. We have worked out the core aspects of the workflow in the engine. This is being extended with more details of the verbs.

Using lots of text to develop tools

The rules-based approach is resource intensive. This, in combination with global hype around "Big Data", has brought data-driven approaches to the fore.

The hope is that better quality tools may now be developed with less effort and that it will be easier to reuse those tools for related languages. This can work, provided one has a lot of good quality text, referred to as a corpus.

Such corpora are being developed, and the recently established South African Centre for Digital Language Resources (SADiLaR) aims to pool computational resources. We investigated the effects of a corpus on the quality of an isiZulu spellchecker, which showed that learning the statistics-driven language model on old texts like the bible does not transfer well to modern day texts such as news items from the Isolezwe newspaper, nor vice versa.

The spellchecker has about 90% accuracy in single-word error detection and it seems to contribute to the intellectualisation of isiZulu.

Its algorithms use trigrams and probabilities of their occurrence in the corpus to compute the probability that a word is spelled correctly, rather than a dictionary based approach that is impractical for agglutinating languages. The algorithms were reused for isiXhosa simply by feeding it a small isiXhosa corpus: it achieved about 80% accuracy already even without optimisations.

Data-driven approaches are also pursued in tools for finding information online, i.e., to develop search engines alike a 'Google for isiZulu'. Algorithms for data-driven machine translation, on the other hand, can easily be misled by out-of-domain training data from which it has to learn the patterns.

Relevance for South Africa

This sort of natural language generation could be incredibly useful in South Africa. The country has 11 official languages, with English as the language of business. That has resulted in the other 10 being sidelined, and in particular those that were already under resourced.

This trend runs counter to citizens' rights and the state's obligations as outlined in the Constitution. These obligations go beyond just promoting language. Take, for instance, the right to have access to the public health system. One study showed that only 6% of patient-doctor consultations was held in the patient's home language. The other 94% essentially didn't receive the quality care they deserved because of language barriers.

The sort of research I'm working on with my team can help. It could contribute to, among others, realising technologies such as automatically generating patient discharge notes in one's own language, text-based weather forecasts, and online language learning exercises.

This article was originally published on The Conversation. Read the original article.

Provided by The Conversation