On the development of NLP Tools for the Georgian Language

Irakli Tsikarishvili[1], Krzysztof Jassem[2], Ioseb Otskheli[3],
Urszula Boryczka[1]

[1]University of Silesia in Katowice
itsikarishvili@us.edu.pl
urszula.boryczka@us.edu.pl

[2]Adam Mickiewicz University
jassem@amu.edu.pl

[3]Poznan University of Technology
iosebi.otskheli@doctorate.put.poznan.pl

Abstract

The paper reports on the development of basic NLP tools for the Georgian language that resulted from the experiment on statistical translation between Georgian and Polish. The tokenizer and the sentence splitter that have been created for the sake of the experiment are now available for use and further development. These are the first publicly open NLP tools for the Georgian language.

1. Introduction

Georgian is one of the less-resourced languages. Spoken by four million people, the Georgian language is sparsely represented in the Internet. Structured bilingual corpora hardly exist for any language pair including Georgian. Even the simplest NLP tools need to be developed from scratch.

The research under description aimed at the creation of a Statistical Machine Translation (SMT) system between Georgian and Polish. The Moses translation toolkit [Koehn-2007] was selected for building the translation engine (http://www.statmt.org/moses/manual/manual.pdf). The major difficulties that needed to be overcome were the sparsity of bilingual corpora and the lack of Natural Language Processing (NLP) tools for Georgian.

2. Georgian language

Georgian is the native language for Georgians. The exact creation date of the Georgian writing is not known, but there is some evidence that the Georgian alphabet originated from the Semitic alphabet in the third century B.C. [Javakhishvili-1947]. The first version of the alphabet is called Asomtavruli (Eng. translation: rounded). The second graphic variant Nuskha-Khutsuri (Eng. translation: clerical) was used between 9th and 11th century [Dadiani-2007]. The third, contemporary version Mkhedruli (Eng. translation: military) is used since 12th century.

The Georgian alphabet consists of 33 characters, all of them being small letters. The list of the Georgian characters is shown in Fig 1.

N°.	letter	Sound in English	Button on keyboard	UNICODE
1	ა	a	A	U+10D0
2	ბ	b	B	U+10D1
3	გ	g	G	U+10D2
4	დ	d	D	U+10D3
5	ე	e	E	U+10D4
6	ვ	v	V	U+10D5
7	ზ	z	Z	U+10D6
8	თ	t	Shift+T	U+10D7
9	ი	i	I	U+10D8
10	კ	k'	K	U+10D9
11	ლ	l	L	U+10DA
12	მ	m	M	U+10DB
13	ნ	n	N	U+10DC
14	ო	o	O	U+10DD
15	პ	p'	P	U+10DE
16	ჟ	zh	Shift+J	U+10DF
17	რ	r	R	U+10E0
18	ს	s	S	U+10E1
19	ტ	t'	T	U+10E2
20	უ	u	U	U+10E3
21	ფ	f	F	U+10E4
22	ქ	k	Q	U+10E5
23	ღ	gh	Shift+R	U+10E6
24	ყ	q'	Y	U+10E7
25	შ	sh	Shift+S	U+10E8
26	ჩ	ch	Shift+C	U+10E9
27	ც	ts	C	U+10EA
28	ძ	dz	Shift+Z	U+10EB
29	წ	ts'	W	U+10EC
30	ჭ	ch'	Shift+W	U+10ED
31	ხ	kh	X	U+10EE
32	ჯ	j	J	U+10EF
33	ჰ	h	H	U+10F0

Fig 1. Georgian alphabet

The Georgian alphabet is listed in the UNESCO Intangible Cultural Heritage Lists. The direction of the Georgian writing is left-to-right.

According to the general morphological typology, Georgian (like other Caucasian laguages) is the agglutinative language [Melikishvili-2010]. The boundaries between lexical and morphological units are strictly defined. Any Georgian word may be easily segmented into explicit morphemes.

3. Existing resources and related work

A small Georgian-Polish parallel corpus turned out to be available from the Opus project [Tiedemann-2012]. The Opus web-page (http://opus.lingfil.uu.se/) gives access to the large database of parallel corpora, for hundreds of language pairs. The Georgian language is present in the project, although it is connected with just a few languages. The Georgian-Polish Opus corpus theoretically consists of 20 000 units. However, a significant part of the Polish side is actually represented by English texts. After cleaning the corpus contains 8000 units.

It is worth noting that a project on the creation of an English-Georgian corpus is currently run at Ivane Javakhishvili University under the supervision of Tinatin Margalitadze [Margalitadze-2015]. The domain of the corpus is science. All texts are translated by human specialists and manually inserted into the corpus.

In 2013, The Software for Composition of Some Georgian language words has been created in Sokhumi State University [Antidze-2013]. The tool uses the database of roots of Georgian words and morphological categories in order to derive base wordforms from inflected wordforms.

Because of its aglunative nature, the Georgian language is not easy for the computerized analysis. The first step on the way to creation robust NLP tools is to gather and annotate a large corpus of various types of texts. Such an initiative has been taken by Sofia Daraselia at the University of Leeds [Daraselia-2014]. Currently, the KaWaC corpus contains up to 150 million annotated words and word-phrases. We are of the opinion that using this resource in future research may improve the quality of tools reported here.

Compared to Georgian, the Polish language is rich in NLP tools and resources. An up-to-date list of processing tools for the Polish language may be found at http://clip.ipipan.waw.pl/LRT. There exist NLP web-service platforms for the processing of the Polish language (Multiservice [Ogrodniczuk-2012], clarin-pl [Piasecki-2014], PSI-Toolkit [Gralinski-2013]). PSI-Toolkit is a collection of NLP tools that work on the same data structure and may be called in one processing pipeline. This was one of the reasons for choosing the toolkit for our experiment.

4. Need of NLP tools for SMT

Our experiment consisted in the creation of an SMT system for less-resourced languages by means of the tools that have been widely used for popular language pairs. We trained the engine by means of the MOSES toolkit. The required data format for MOSES is a bilingual corpus that have been tokenized, segmented and aligned. Based on the statistical analysis of the corpus, the MOSES software builds the phrase translation table in the first stage (training). In the second stage (decoding) the system returns the most likely translation of each input sentence, according to the translation table.

In order to build the Georgian-Polish translation engine, it proved necessary to use pre-processing tools such as tokenizer and sentence-splitter for the two languages. No such tools existed for the Georgian language before the start of the project. Due to the fact that hardly any parallel corpora were available in the web, it was necessary to obtain bilingual data from various sources. For the sentence-alignment task we used the hunalign[Varga-2005] aligner. Although the tool does not require any lexicon, the presence of a prepared dictionary significantly increases the quality of alignment.

4.1. Segmentation and tokenization

Sentence is a word or a group of words expressing a complete thought. Segmentation is an automated process that divides text into sentences. The program that handles this task is called sentence-splitter or segmenter.

Similar to Indo-European languages, in the Georgian language the end of a sentence is designated by punctuation marks (.), (?), (!) or their combinations. Real-life texts pose various challenges to the task of sentence-splitting. These are the most common positions in the text that cause the tokenization problems:

- abbreviation ending with a dot
- small letter after a dot
- personal name after a dot
- ellipsis
- quotes
- urls
- non-standard sentence endings (e.g. parentheses)

[Milkowski-2011] addresses the problem of sentence-splitting of Polish texts.

Tokenization is an automated process of breaking a text into tokens, i.e. meaningful elements such as words or symbols. Similar to the Indo-European languages, in the Georgian language the tokens are divided from each other by spaces. The

challenges to overcome in the tokenization task are usually language-specific. For the English language the challenges may concern for example:

- apostrophes
- hyphenation
- unusual tokens
- multiword expressions

The tokenization issues for the Polish language have been discussed in [Radziszewski-2011].

5. Tools developed

5.1. Alignment

We used hunalign as a tool for text alignment. The tool assigns the quality scores for aligned texts. This assessment scale ranges from 0 (lowest) to 10 (highest). According to our observations the bilingual texts may be considered to be useful for statistical translation if their assessment exceeds 1.5. We have automated the aligning process with the script, which runs as follows:

- Reads a bilingual text from text file;
- Implements tokenization and segmentation for the two languages respectively;
- Aligns the text with Hunalign;
- Inserts the text into the corpus if its Hunalign assessment exceeds 1.5;

For the improvement of the alignment quality, hunalign may be supplemented by a bilingual lexicon. No such a resource existed for the Georgian-Polish pair at the beginning of the project. We decided to create a Georgian-Polish lexicon by compiling the existing English-Georgian and the English-Polish dictionaries. We automatically matched entries that shared the English equivalent. As a result we obtained the Georgian-Polish lexicon containing up to 45 000 words. Although the reliability of the resource is not sufficient for human translation, it proves helpful for NLP tasks. Table Fig 2. compares the quality of hunalign alignment with and without the use of the lexicon.

Number of sentences	Without the dictionary (hunalign assessment mark)	With dictionary (hunalign assessment mark)
19679	2,311	3,028
15434	2,456	3,981
12654	2,123	3,414
8495	2,735	3,945
6567	3,675	4,836
5479	1,564	3,176
5225	1,341	2,964
1456	0,876	2,167
1234	0,354	1,456
1199	0,292	1.765
average	1,772	3,218

Fig 2. Quality of hunalign with and without the lexicon.

The comparison shows that the compiled lexicon significantly improves the hunalign quality. The quality of hunalign's work depends mainly on the length of the text. Aligning is less effective for texts below 1500 sentences. The results show that without the lexicon, the quality of alignment does not reach the threshold (1.5) for use in parallel corpora. The lexicon is available at: (https://www.dropbox.com/s/j7ss723cj207es5/dict.ka-pl.dic.zip?dl=0).

5.2. Georgian segmenter

Assuming that orthography rules of the the Georgian language resemble those of the Indo-European languages, our first attempt consisted in adapting an existing segmenter (for Polish) to the Georgian language. For this purpose, we manually created a list of Georgian abbreviations. (Such a database did not exist at the beginning of Project). The current volume of the list is 250 items. Fig 3. shows the beginning of the list.

Georgian	English
კმ/წმ.	Km/s
კმ/სთ.	Km/h
ძვ. წ. ა .	BCE
ინგ.	English
ე.წ.	called
ტ.	tonne
მლნ.	mln
კგ.	kg
მაგ.	e.g.
ა.შ.	etc.

Fig 3. Example of Georgian abbreviations

We tried to apply the existing PSI-Toolkit Polish sentence-splitter by replacing the list of the abbreviations with that for the Georgian language. The attempt failed – only a part of the testing corpus was

segmented correctly. The lack of capital letters in the Georgian texts was responsible for not breaking the texts on sentence boundaries. On the other hand, the Georgian letters that are typed with the SHIFT key (Picture 1.) were responsible for incorrect breaks of texts before such characters. Non-existence of capital letters also caused a problem in contracted writings of a name and a surname (a full stop after the first letter of a name, and a surname). Fortunately, no one-letter words exist in the Georgian language, therefore one-letter token before a full stop cannot mean the end of a sentence.

These peculiarities of the Georgian language persuaded us to create the Georgian segmenter. The tool consults two text files. The first file contains all punctuation marks which may end the sentence. The second file contains Georgian abbreviations and other figures, written as regular expressions, which can be interpreted by the program as the end of sentence.

The segmenter works as follows:

1. Place the whole text in one line;
2. Search for texts that match regular expressions (from shortest to longest).
3. Replace each fragment that matches a given regular expression with a unique identifier of the changed string.
4. Segment the text according to the list of punctuation marks;
5. Replace the identifiers with the corresponding string;

The method is simple and the segmenter is easy to adjust. The correction of splitting mechanism merely consists in a modification of a text file. The code of the segmenter is available at (https://bitbucket.org/irakli8888/georgiansegmenter).

5.3. Georgian tokenizer

Non-existence of capital letters causes similar problems to the tokenizer as to the segmenter. Therefore, we assumed the similar approach to the task of tokenization to that used for sentence-splitting. We decided to create the tool from scratch. The tokenizer consults two text files, analogous to those of the segmenter. The main file contains the list of abbreviations enriched by other regular expressions that may represent tokens of the Georgian language. Currently the file contains more than 100 non-abrreviation token representations. Fig 4. shows the beginning of that file.

- tel:[phone number]:
  ტელ:^\+?[0-9]{3}-?[0-9]{6,12}$
- email:
  b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}\b
- period from 'x' to 'y' year (1999-2001 წწ.):
  [0-9]{1,4}-[0-9]{1,4} წწ\.
- date dd.mm.yyyy:
  \d{2}.\d{2}.\d{4}
- century before common era:
  ძვ\. წ\. ახ\. (XC|XL|L?X{0,3}) ს\.

Fig 4. Excerpt from the file representing Georgian tokens

The other file, containing punctuation marks, is also different from that prepared for the segmenter. Besides the marks denoting the end of a sentence, the list contains all punctuation symbols. The code of the tokenizer is available at (https://bitbucket.org/irakli8888/georgiantokenizer).

6. Conclusions and future work

The paper reports on the research that aimed at the creation of the Georgian-Polish statistical MT system. The experiment revealed the need for the development of NLP tools for the Georgian language, such as tokenizer and sentence-splitter. Our attempts to obtain such tools by adapting existing tools to the Georgian language did not succeed due to several peculiarities of the Georgian language. We created the tools from scratch. The source code is open for use and further development. We have also created online tool with free access to it (http://geonlp.us.edu.pl/).

We believe that the result of our project will contribute to the progress in the natural language processing of Georgian, one of the less-resourced languages. In future we hope to improve our tools and, in co-operation with other centers that deal with the Georgian language develop new ones, such as:

- Tagger;
- Lemmatizer;
- Dictionary of flexible forms of the Georgian words;

We have high hopes that the resources compiled in the Kawac project [Daraselia-2014] might prove particularly helpful.

7. References

Koehn, P. Federico, M. Cowan, B. Zens, R. Dyer, C. Bojar, O. Constantin, A. Herbst, E. - (2007).: Moses: Open Source Toolkit for Statistical Machine Translation. Prague: ACL '07 Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, Pages 177-180;

Javakhishvili, I. - (1949).: Kartuli Paleontologia (Georgian Paleography). Tbilisi: The Tbilisi University press;

Dadiani, E. Putkaradze, E. - (2007).: Kartvelian Language Space: Kartvelian Heritage XI. Kutaisi: Kutaisi State University Press;

Melikishvili, D. Daniel Humphries, J. Kupunia, M. - (2010).: The Georgian Verb: A Morphosyntactic Analysis. Georgia: Dunwoody Press;

Tiedemann, J. - (2012).: Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). Istanbul, Turkey. European Language Resources Association (ELRA);

Margalitadze, T. Ormotsadze, I. - (2015).: The Platform of English-Georgian Parallel Corpus of Scientific Texts and Specialized Lexicography: International Conference 'Language and Modern Technologies' organized by Ivane Javakhishvili Tbilisi State University, Goethe University Frankfurt am Main and Arn. Chikobava Institute of Linguistics. Tbilisi;

Antidze, J. Gulua, N. Kardava I. - (2013).: "The Software for Composition of Some Natural Languages' Words": Lecture Notes on Software Engineering, Volume 1 Number 3;

Daraselia, S. Sharoff, S. - (2014). Morphosyntactic Specifications for KaWaC, a Web Corpus for Georgian. In Proceedings of Humanities in the Information Society II Conference, 24-26 October. Batumi Shota Rustaveli State University, Georgia;

Ogrodniczuk, M. Lenart M. - (2012).: Web Service integration platform for Polish linguistic resources. Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012. ELRA, Istanbul, Turkey;

Piasecki, M. - (2014).: User-driven Language Technology Infrastructure – the Case of CLARIN-PL. Ljubljana. 9th Language Technologies Conference Information Society - IS;

Gralinski, F. Jassem, K. Junczys-Dowmunt, M. - (2013).: PSI-Toolkit: A Natural Language Processing Pipeline: In Proceedings of the 6th Language and Technology Conference. Springer Berlin Heidelberg.;

D. Varga, L. Németh, P. Halácsy, A. Kornai, V. Trón, V. Nagy - (2005). Parallel corpora for medium density languages. Recent Advances in Natural Language Processing IV RANLP. pages 590-596. Borovets, Bulgaria. John Benjamins Publishing Company.

Miłkowski M., Lipski J. - (2011).: Using SRX Standard for Sentence Segmentation. In: Vetulani Z. (eds) Human Language Technology. Challenges for Computer Science and Linguistics. LTC 2009. Lecture Notes in Computer Science, vol 6562. Springer, Berlin, Heidelberg;

Radziszewski, A. Sniatowski, T. - (2011).: Maca, a configurable tool to integrate Polish morphological data. Proceedings of the Second International Workshop on Free/Open-Source Rule-Based Machine Translation. Open University of Catalonia (Barcelona);