Reports of Enlarged Session of the
Seminar of I. Vekua Institute
of Applied Mathematics
Volume 23, 2009

A GEORGIAN LANGUAGE MORPHOLOGICAL PARSER

Kapanadze O.

Abstract. In the paper application of the Finite State Tools to Georgian is discussed. The FST has been very popular in computational morphology and other lower-level applications in natural-language engineering. The basic claim of finite-state morphology is that a morphological analyzer for a natural language can be implemented as a data structure called a Finite State Transducer. In the Georgian language, as in many non-Indo-European agglutinative languages, concatenative morphotactics is impressively productive within its rich morphology. The presented Georgian Language Morphological Parser is capable to analyze all theoretically possible options for the lemmata of Georgian nouns, pronouns, adjectives, adverbs, numerals, functional words and for most of the lemmata from about 150 sets of verb constructions. A demo of the Georgian language morphological transducer is anticipated during the seminar session.

Keywords and phrases: Finite-state transducer, computational morphology.

AMS subject classification (2000): 68T50.

The main task of the outlined endeavour was to prepare and conduct a feasible derivation of a lingware - the grammar and lexis - for text analysis and generation in the Georgian language based on the **Finite-State Automata** paradigm. The Finite-State techniques have been very popular and successful in computational morphology and other lower-level applications in natural language engineering. The basic claim of finite-state approach is that a morphological analyzer for a natural language can be implemented as a data structure called **a Finite-State Transducer**. The FSTs are bidirectional, principled, fast, and (usually) compact. Defined by the linguists using declarative formalisms, and created using algorithms and compilers that reside within a pre-written finite-state implementation, finite-state systems consist admirable examples of the separation of language specific rules and language-independent engine.

In early examinations of formal language theory and its possible application to natural-language processing, the Finite-State power was dismissed as inadequate for handling interesting natural language phenomena – "Chomsky Hierarchy":

Finite-State
Context-Free
Context-Sensitive
Turing Machine.

For this reasons they were largely overlooked for decades. Attention on the 1960s through most of the 1980s moved on to more powerful, but less computationally attractive, context-free and context-sensitive grammars. But since the 1990s finite-state power has been rehabilitated in numerous research and commercial applications, in conferences held and books written. Although it is generally recognized that finite-state techniques cannot do everything in computational linguistics, the rehabilitation of finite-state power came from an insight into phonological rewrite rules or, more generally, alternation rules. In applications where finite state methods are appropriate, they are extremely attractive, offering mathematical elegance that translates directly into computational flexibility and performance. An early finite state system, Two-Level Morphology, was developed by K. Koskenniemi [2]. It gave linguists a way to do finite state morphology before there was a library of finite state algorithms and before compilers for alternation rules were developed.

The academic grammars and dictionaries for the Georgian language abound, though, this does not mean that there exists support for computational applications involving this language, since these resources are not available in a form that makes them applicable for computational processing. A natural language engineering system includes many smaller processing components that contribute to precise sub-tasks, and solve a specific language sub-problem. The basis for all large scale natural language processing system application be it symbolic / rule-based, be it quantitative / statistical or probabilistic is a large coverage morphological analyzer, especially for highly inflectional languages that comprises inflection, derivation and compounding. A natural language will typically contain tens of thousands, or even hundreds of thousands of roots. A morphological analyzer is intended to perform a complete analysis of an inflected word form and produce a citation form (a canonical form of a word used as a headword in dictionaries) and a set of morpho-syntactic features (number, person, gender, case, etc) for using the respective information in different language technology applications. The morphological analyzer then becomes a component in a larger system that performs parsing, spelling correction, indexing, data-mining, machine translation, etc

As an implementation environment for developing a Georgian language lexical transducer we applied a toolkit known as the *Xerox Finite-State Calculus* [1]. This product has been successfully utilized for English, French, Spanish, Portuguese, Italian, Dutch, German, Finnish, Hungarian, Turkish, Danish, Swedish, Norwegian, Czech, Polish, Russian, Japanese.

The Georgian Language Finite-State Lexical Transducer consists of the 7 different modules for the Noun, Pronoun, Adjective, Numeral, Adverb, Minor-Categories and Verb parsing. The developed FST lexical transducer as a Morphological Parser is a part of the lingware for Georgian language and produces tagged and lemmatized output of the source Georgian plain text.

The Noun, Pronoun, Adjective, Numeral wordform's structure in Georgian is well known and is described by means of the following template:

NOUN_Stem+PLURAL_MARKER+CASE_MARKER+Emph_Vocal+POSTFIX+Emph_Vocal

R + (eb) ∼ / (n/T) + 7 options + (a) + 9 options + (a)

The structural unites introduced in the Italic, are optional.

Standard academic grammars of Georgian list up to 21 classes of noun stems. The

Georgian Noun analysis and generation module based on FST tools uses flag diacritics - a mimic feature structure unification compiled out at the end pure finite state. The number of citation forms of the Georgian Noun lexicon exceeds 20.000 lemmata and the Morphological Transducer is capable to parse all theoretically possible Noun forms.

An example of a typical output of a noun analysis:

[kacebisatvis] (”for the men”)
them=kac, numerus=pl{eb}, casus=genitius, postfix_Benefective_for {”tvis”}, cat=N.

[qalaqamde] (”until the city”)
stem=qalaq, postfix_till{mde}, cat=N.

[bankirma] (”banker”)
stem=bankir, casus=ergatius , cat=N.

For the Georgian FST parser, alongside the mentioned 4 moduls, the adverbs’, the Minor-classes’ (functional words) and the Verb analysis moduls have been developed. The first two, due to their simple morphological structure, are aggregated in a program code without sophisticated morpho-syntactic rules. The corresponding modules utilize a matching algorithm for the input adverbs’ and functional words’ recognition.

The most time consuming endeavour in the parser development was the Georgian verb analysis module. Despite the broad skepticism about the feasibility of computational analysis of Georgian verb the output of the respective module demonstrates consistency and reliable recall. Moreover, the verb parsing module had been constructed in a way that it provides the verb analysis output with the syntactic valency information crucial for the Georgian text shallow parsing / syntactic chunking procedures.

A typical result of the Georgian verb analysis contains a morphological structure of a finite verb and an information concerning its syntactic valency. The output of parsing procedure for two Georgian verbs [vyidit] and [damexatebodes] are as follows:

[vyidit] (”we sell it”)

Subj3/v + yid + them/i + t = atsmko/Subj1P1 + Obj3Sg
Subj3/v + yid + them/i + t = atsmko/Subj1P1 + Obj3Pl

[damexatebodes] (”[if] it would be painted for me”)

Prv/da+Obj1Sg/m+Pas/e+xat+eb+od+e+s=kavshirebiti-2/Subj3Sg+ Obj1Sg.

In the output the verb morphological data is supplemented by the syntactic information describing verb syntactic valency.

The first release of the **Georgian Language Finite-State Morphological Parser** was informally tested and positively evaluated by MA students of the Tbilisi State University as well as by the linguist who were not part of the development team. Besides, the results of the mentioned edeavour has been published and reported at several conferences, among them on the FSMNLP 2008 workshop organized at the Joint-Research Centre of the European Commission (JRC), September, 2008 in Ispra, Italy.

In the meantime the Georgian morphological Finite-State Transducer is ported to the LINUX UBUNTU version. The first experiments with a freely available Georgian text in UTF-8 encoding has been successfully accomplished.

The FSTmorphological parser may be used as a basis for **Machine Translation** to supply a Machine Translation system's engine with **tagged, lemmatized and chunked collection of sentences in Georgian language**. However, it also may become the basis for all kinds of probabilistic systems as it would allow in a unique way to train statistical parameters as well as fit into a Statistical/Hybrid MT concept and prepare a basis for development of a Georgian MT system in the multilingual context.

REFERENCES

1. Beesley K., Karttunen L. Finite State Morphology. *CSLI Publications, Center for Study of the Language and Information, place Leland Stanford Junior University*, 2003.

2. Koskenniemi K. Two-level Morphology: A General Computational Model for Word-form Recognition and Production. *Publication 11, University of Helsinki, Department of General Lingusitics, Helsinki*, 1983.

Received 15.04.2009; revised 2.08.2009; accepted 27.11.2009.

Author's address:

O. Kapanadze
OK'OMPLEX-Innovative Information and Language Technologies
6, Beridse St., Tbilisi 0118
Georgia
E-mail:ok@caucasus.net