Creation of a morphological analyzer based on finite-state techniques for the Uzbek language

I I Bakaev1, R I Bakaeva2

1Tashkent University of Information Technologies, 108, Amir Temur ave., Tashkent, 100200, Uzbekistan
2Bukhara pedagogical college, 5, Bogishamol Str., Bukhara, 200104, Uzbekistan

E-mail : bakayev2101@gmail.com

Abstract. The article analyzes the existing morphological analyzers related to agglutinative languages. The architecture of the morphological analyzer for the Uzbek language is proposed, the functionality is described using the IDEF0 model, and a finite-state transducer for grammatical and morphological rules is created. Here also described a formal mathematical model for rules morphotactics. Based on the rules of grammar, morphotactics, morphonology, an algorithm was developed using a finite state machine and a transducer.

1. Introduction

NLP is a popular area of artificial intelligence that is widely used in machine translation tasks, text analysis, annotation, text generation, autoreferencing, etc. Although these tasks differ in their type of activity, they have common problems that are related to the morphological analysis of texts. These problems led to the need to create a morphological analyzer for natural languages.

Morphological analysis is a method for solving problems that requires an accurate description of the problem of natural language [1,2].

In recent years, researchers have developed many types of morphological analyzer. Below is a description of some of them that relate to agglutinative languages.

In [3], a stochastic finite morphological analyzer for the Turkish language is presented. The analyzer is a standard implementation of a two-level morphological formalism finite-state transducer (FST). For a stochastic morphotactic transducer (this is the way in which morphemes are combined to form words), an ambiguous text corpus of 200 million words is used, then it is formed using a morphophonic (morphophonics studies the area of the relationship between morphology and phonetics) transducer to produce a stochastic morphological analyzer. The authors propose two methods for evaluating the effectiveness of a stochastic analyzer such as spelling correction and morphological language modeling for speech recognition.

By Authors [4] the TRmorph morphological analyzer for Turkish is described. TRmorph is a freely available two-level morphological analyzer. In TRmorph, the implemented morphology specification is suitable for the Turkish language and the analyzer has a large lexicon. In the development of TRmorph, SFST (Stuttgart FST) tools were used. The SFST for the TRmorph morphological analyzer
consists of three main parts. This state machine is specified using regular expressions to describe
morphotactics; a set of two-level rules for determining phonological or spelling changes; and a lexicon
containing 37,101 words which of them has 1,500 root forms. To evaluate the analyzer, a qualitative
analysis was performed on two large corpora.

The following morphological analyzer for the Turkish language TRMOR is presented in [5]. This
analyzer like TRmorph uses the SFST tool. TRMOR, like other analyzers, uses morphotactic and
morphophonological rules and lexicon of word roots. These works describe the morphological
structure, phonological and morphological rules that are implemented in TRMOR. TRMOR analyzes
word forms, breaks them down into morphemes, and checks them for accuracy and correctness. When
testing the accuracy of the analyzer, 1000 words were randomly selected from Wikipedia's word lists.
The test result gave 94.12% accuracy.

This article [6] presents the stages of developing a morphological analyzer for the Turkish
language. The method of affixes is used for analyzing Turkish words without using the lexicon. The
structure and rules of the Turkish language are modeled using finite automata. Unlike other works, all
Turkish suffixes are grouped into five classes. For each class of suffixes, a finite state machine (FSM)
has been developed that describes the morphotactic rules for concatenation of suffixes in reverse
order. During the analysis process, the global FSM module is formed. This FSM provides interaction
of the previously developed FSM module with each other. At the end of the analysis with the global
FSM module, the word is divided into root and suffixes. If you need to add a new suffix to the
analyzer, you need to make a change to the suffix class and update the FSM.

In the work [7], using the corpus of the Kazakh language, the method of segmentation of
inflectional affixes is implemented. The process of forming automatic segmentation of inflectional
affixes is divided into three stages. In the first stage, a finite automaton is constructed and
segmentation is performed for inflectional affixes. In the second stage, special finite automata are
constructed for lexical categories of nouns and verbs. In the third stage, the methods of bi-directional
segmentation and lexical analysis are used for stemming and segmentation of inflectional affixes. The
main goal of this work is to improve the accuracy and speed of stemming inflectional affixes for the
Kazakh language.

The authors [8] analyzed the morphological properties of the Kazakh and Turkish languages.
Ontological models of lexical categories of nouns for the Kazakh and Turkish languages were created.
The following tasks were also solved: the ontology of languages was compared; a unified system of
symbols of morphological features was developed, and the morphological rules of languages were
written using a new system of symbols. The unified morphological analyzer is developed on the basis
of a general morphological analysis algorithm.

The article [9] considers a universal morphological analyzer tool that can split a text into a
sequence of morphemes or syllables. The basic rules and statistical models of the approach to
segmentation of morphological units are taken into account. When developing a tool for controlled
morpHEME segmentation, standardization and manual segmentation of Uyghur morphemes, especially
suffixes, as well as generalized and classified morphophonetic rules for their implementation are
proposed. Developed morphological analyzer may segment the text into phonemes, syllables,
morphemes and words. The results of the experiments are presented.

This paper [10] presents a detailed morphological analysis of the Kazakh language, which is an
agglutinative language. The analysis of the morphology of the Kazakh language and implemented the
formalization of rules for the entire morphotactics of the Kazakh language. The morphological
analyzer covers all the rules of the Kazakh language. The analyzer is also built using a two-level
morphological approach using the tools of the Xerox finite-state transducer (XFST).

The authors [11] describe a weighting finite morphological transducer for the Crimean Tatar
language that can analyze and generate Latin and Cyrillic orthography. When developing the
transducer, the Latin spelling is used. Then a separate transliteration transducer is designed to display
surface forms in Cyrillic. To control non-determinism in an orthographic display, weights are used to
prioritize the shapes visible in the case. All components of the system that detect accuracy above 90% for morphological analysis and about 90% for spelling conversion were evaluated.

This paper [12] describes an implemented morphological analyzer for the Turkmen language using the tools of the Xerox FST. Morphotactic rules are performed using finite automata that are constructed using a lexicon containing word roots, lexicons, and suffixes. The analyzer's lexicon has 1200 root words. The analyzer is designed for machine translation systems between the Turkmen and Turkish languages.

Based on the above, the purpose of this article is to create the architecture and algorithm of the morphological analyzer, to describe grammatical rules and morphotactic rules with the help of an FST and FSM for the Uzbek language, where the lexicon is taken into account.

2. Statement of the problem.
The morphological analyzer for the Uzbek language should provide the following features:

- Describe the morphological properties of words in the text using grammatical rules;
- Check for the correct spelling of words in the Uzbek language (rules morphotactics);

3. Method of designing a morphological analyzer.
The morphological analyzer is a set of algorithms that work with texts to determine the grammatical features of a given word, checking word forms from the dictionary (lexicon) for compatibility Fig. 1.

![Figure 1. Formal scheme of the morphological analyzer](image)

Below we describe the architecture of the morphological analyzer using the IDEF0 model in Fig. 2.

![Figure 2. Description of the morphological analyzer using the IDEF0 model](image)
Figure 3. A more detailed description of the morphological analyzer operation process using the IDEF0 model.

The morphological analyzer consists of four text processing functions. These are word tokenization, normalization, word segmentation, and POS tagging.

3.1. Tokenization.
Tokenization is a text processing process that performs the following actions Fig. 4.

3.2. Normalization.
The normalization process consists of several text processing algorithms, such as decoding abbreviations of words, clearing text from punctuation marks, converting characters to a single case, and so on.

3.3. Segmentation of words.
In morphology, the term word segmentation refers to the division of words into root, prefix, suffix, affixes, and also grammatical rules that are pre-defined lexicons.

3.4. POS(part-of-speech) Tagging.
This is the process of determining the part of speech and grammatical properties of words in the text that is attributed to the corresponding tag. Usually existing POS Tagging algorithms are based on probability theory and template-driven lexicons.

4. Mechanisms of the morphological analyzer.
In a morphological analyzer, each function operates with pre-defined mechanisms:
- lexicon of grammatical rules as a template;
- lexicon of word forms as a database,
- morphotactics and morphonology ruled.

4.1. Lexicon.
Lexicon is a set of descriptions of morphological forms of words and rules for determining whether words belong to a part of speech, which is jointly developed by a linguist and programmer. However, when designing a lexicon, different approaches are used based on the properties and structure of the language. This article offers 2 lexicon approaches for the morphological analyzer, such as a lexicon with grammatical rules and a lexicon of word forms.

A lexicon with grammatical rules based on the FST works as follows:
- Accepts as input word bases, word-forming prefixes, suffixes, and affixes (subjective evaluation, plural, belonging, case) followed only by a valid sequence of tags Fig.5.
- As output, template tags of grammatical rules are replaced with morphemes (affixes) that they correspond tab. 1.

Defining formal tags for the lexicon:
- V–verb
- N–noun
- Num – the numeral
- Pr – Pronoun
- Adv – adverb
- Adj – adjective name
- Pt-past tense affix
- 1pPl – 1 person, plural forms
- Pl–plural forms
- 3pSg – 3rd person, singular
- 2pSg – 2nd person, singular
- CA-case affix

The lexicon of grammatical rules is based on the FST as shown in figure 5:.

![Figure 5A. Finite-state transducer(FST) for verb](image)

![Figure 5B. FST for nouns](image)

![Figure 5C. FST of numerals](image)
The grammatical structure of words and patterns that match the grammatical rules and the result of the FST are described Table 1. You need to consider that the template of the grammar rules can still continue.

### Table 1. Grammar rules Uzbek language.

| №   | Examples                  | Table of Grammatical rules | Result of the final Converter |
|-----|---------------------------|-----------------------------|-------------------------------|
| 1   | yoz+di+k (we wrote)       | V+PT+1pPl                   | yoz +V+PT+1pPl                |
| 2   | kitob+lar+i+ni (his books)| N+Pl+3pSg+CA                | kitob+N+Pl+3pSg+CA            |
| 3   | besh+lar+ing (your five marks) | Num+Pl+2pSg        | besh+Num+Pl+2pSg              |
|     | hamma+lar+ingiz+dan        | Pr+Pl+2pPl+CA               | hamma+Pr+Pl+2pPl+CA           |
| 4   | (from all of you)         |                             |                               |
| 5   | ko’p+ni (several)         | Adv+CA                      | ko’p+Adv+CA                   |
| 6   | yaxshi+si+ni (the best)   | Adj+3pSg+CA                 | yaxshi+Adj+3pSg+CA            |

### 4.2. Lexicon of word forms

In our case, for the lexicon of word forms, we use the ER model (eng. "Entity-Relationship") which gives a visual representation of the lexicon Fig. 6.
Figure 6. ER-model for the lexicon of word forms

Description of lexicon tables of word forms Tab.2:

| №   | Names   | Description                                      |
|-----|---------|--------------------------------------------------|
| 1   | ID      | Identifier                                       |
| 2   | Stem    | Immutable part of a word                         |
| 3   | Lemma   | Word formation                                   |
| 4   | Prefix  | Prefix of the word formation                     |
| 5   | suffix1 | Suffix of the word formation                     |
| 6   | suffix2 | Suffix of the word formation                     |
| 7   | suffix3 | Suffix of the word formation                     |
| 8   | suffix4 | Suffix of the word formation                     |
| 9   | morph_desc | Morphological description of the word formation |
| 10  | Id_Class| The ID of the lexical categories                  |

Table 3. The fields of the table LexClass.

| №   | Names    | Description       |
|-----|----------|-------------------|
| 1   | ID       | Identifier        |
| 2   | Class name | Name of the class |
Based on the data schema, you can see that a language word form can consist of 4 suffixes and one prefix.

4.3. Rules of morphotactics of the Uzbek language.
In computational linguistics, the term morphotactics refers to the ordering of morphemes (affixes) that is glued to the rules of morphology to create new word forms. Morphotactics takes into account the following rules:

- derivational rules;
- allowed combinations between morphemes;
- simple concatenation;
- complex models root or template;
- regularity of language dependence [13].
- For generalizing structures word formation lexical categories analyze the words as shown in table 4.

### Table 4. Uzbek language word patterns.

| №  | Word formation            | Structure                  | Lexical categories |
|----|---------------------------|----------------------------|--------------------|
| 1  | kitob+ xon+ga (to readers)| root + {suffix} +{affix}   | Noun               |
| 2  | aql+ li+lar(clevers)     | root + {suffix} +{affix}   | Adjective          |
| 3  | gul+la+di(florished)     | {prefix}+ root + {suffix}  | Verb               |
| 4  | be+odob+lik+ning (discourtesy) | {prefix}+ root + {suffix} +{affix} | Noun             |
| 5  | ba+davlat(rich)          | {prefix}+ root             | Adjective          |
| 6  | be+ayb+lar(innocent)     | {prefix}+ root +{affix}    | Adjective          |

When analyzing the tab.4. We offered a formal mathematical model for the rules of morphotactics of the Uzbek language:

\[
W = \{PR\} + K + \left\{ \sum_{i=0}^{4} s_{xi} \right\} + \left\{ \sum_{i=0}^{n} a_{f_i} \right\}
\]  

(1)

where \(W\) – word form, \(K\)-word root, \(PR\)-word-forming affixes (suffix), \(SX\)-word-forming affixes(suffix), \(AF\)-affixes

Using the formula (1), we construct a FSM for the rules of morphotactics:

Defining the final state of machine:

- FSM: \(M = (W, Q, q_0, F, \delta)\),
- input alphabet: \(W = \{PR, K, SX, AF\}\);
- set of States: \(Q = \{q_0, q_1, q_2, q_3, q_4\}\)
- transition function: \(\delta(Q, W) : Q \times W \to Q\)
- initial state: \(q_0, (q_0 \in Q)\).
Next, for (1), we create the formula using the algorithm in Fig. 7.

**Figure 7.** The formation algorithm morphotactics with the help of FSM.

We describe the value of the transition function for FSM Fig. 7:

\[
\begin{align*}
\delta(q_0, PR) &= q_1, \quad \delta(q_0, \varepsilon) = q_1, \\
\delta(q_1, K) &= q_2, \quad \delta(q_2, SX) = q_3, \\
\delta(q_2, AF) &= q_4, \quad \delta(q_3, AF) = q_4, \\
\delta(q_4, AF) &= q_4
\end{align*}
\]

FSM has five states, of which three are the final state q2, q3, q4. In the state q2, q3, q4, FSM admits this word and is considered a correctly formed word form.

**4.4. Morphological rules**

In the Uzbek language, with the addition of word-building suffixes and grammatical affixes, different phonetic changes occur in the root of the words, such as the vowel letter "о" go to the letter, "а" or vice versa, the consonants "к" to the letter "г", with the addition of word-building suffixes and affixes, consonants are added to the roots of words, or vowels are dropped.

Below, as an example of nouns, we will review one of this phonetic changes. If we add the word-form suffix to the root of the word, then the vowel letter, and at the root of the words goes to the letter o. (а → о). for example

- Bo'ya+ q= Bo'yoq (paint, paint)
- tara+ q= taroq (comb, comb, comb)
- qiyna+ q= qiynoq (torment, torment, torture).
- so'ra+ q= so'roq (ask, question, interrogate)

Morphonological rules can be described using the FST Fig. 8.

**Figure 8.** Description of morphological rules in the help of the final Converter.
5. Conclusions
The analysis of scientific research related to the creation of morphological analyzers belonging to the family of agglutinative languages is carried out. The functionality of morphological analyzers is also studied.

For the Uzbek morphological analyzer, an architecture is proposed that works on the basis of the lexicon of words, rules of morphotactics and morphonology. All mechanisms are based on the theory of finite automata.

6. Acknowledgements
At the moment, research is continuing. So, the approach is really scaled and morphological and morphemic analysis of words is useful when forming search queries in electronic libraries. These conclusions and suggestions for future research will be presented in the innovative project I-OT-2019-12 "Development and popularization of the electronic "smart library" platform, dedicated to modern scientific and practical interpretation and analysis of the works of our ancestors who contributed.

References
[1] Bolshakova E. 2017 Avtomaticheskaya obrabotka tekstov na estestvennom yazike i analiz dannix (NIU VSHE)
[2] Kurdi M Z 2016 Natural Language Processing and Computational Linguistics 1 (London, United Kingdom: ISTE Ltd and John Wiley & Sons Inc)
[3] Hasim S, Tunga G and Murat S 2009 A Stochastic Finite-State Morphological Parser for Turkish ACL and AFNLP (Singapore) pp 273–6
[4] Gagri G 2010 A Freely Available Morphological Analyzer for Turkish LREC (Malta) pp 820–7
[5] KAYABAŞ 2019 TRMOR: a finite-state-based morphological analyzer for Turkish Turkish J. Electr. Eng. Comput. Sci. 27 3837–51
[6] Gülşen E and Eşref A 2004 An affix stripping morphological analyzer for Turkish AIAA (Innsbruck, Austria) pp 299–304
[7] Gülilä A and Xiaolong W 2010 Kazakh Segmentation System of Inflectional Affixes CIPS-SIGHAN (SIGHAN) pp 46–54
[8] Kessikbayeva G and Cicekli I 2014 Rule Based Morphological Analyzer of Kazakh Language Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM vol Proceeding (Baltimore, Maryland: Association for Computational Linguistics) pp 46–54
[9] Ablimit M, Kawahara T, Pattar A and Hamdulla A 2016 Stem-Affix based Uyghur Morphological Analyzer Int. J. Futur. Gener. Commun. Netw. 9 59–72
[10] Kairakbay B M and David L Z 2013 Finite State Approach to the Kazakh Nominal Paradigm FSMNLP (St Andrews–Scotiland) pp 108–12
[11] Altintas K and Ilyas C 2001 A Morphological Analyser for Crimean Tatar TAINN pp 180–189
[12] Tantüğ C, Eşref A and Kemal O 2006 Computer Analysis of the Turkmen Language Morphology Advances in Natural Language Processing (Finland) pp 186–93
[13] Batura T B 2016 Matematicheskaya lingvistika i avtomaticheskaya obrabotka tekstov (Novosibirsk: RITS NGU)
[14] Bakaev I I and Shafiev T R 2020 J. Phys.: Conf. Ser. 1546 012076