IRSTI 20.19.27 https://doi.org/10.26577/JPCSIT.2023.v1.i1.07

U.A. Tukeyev

Al-Farabi Kazakh National University, Almaty, Kazakhstan
e-mail: ualsher.tukeyev@gmail.com

A NEW COMPUTATIONAL MODEL FOR TURKIC LANGUAGES MORPHOLOGY AND PROCESSING

Abstract. Effective communication between representatives of different nations in the modern global world has become a very relevant problem. Towards its solution, considerable support can come from artificial intelligence tools and, in particular, from natural language processing components. Along this direction, this article proposes the development and the exploitation of new computational morphology model for Turkic languages, based on a complete set of endings (CSE – model). Based on the CSE-model of morphology, a methodology has been developed for the creation and use of universal programs (data-driven) for processing natural languages. These include word stemming, text segmentation and morphological analysis. One advantage of the proposed methodology is that it is oriented towards linguists that only have to prepare i) a list of complete sets of endings for new languages according to the described method, and ii) a list of stop words that do not have endings. Then, based on the prepared lists, the developed universal programs for stemming, segmentation, morphological analysis are used. Experiments carried out for the Kazakh, Kyrgyz and Uzbek languages show a high efficiency of the proposed morphology model, algorithms and tools.

Key words: computational model, Turkic languages, morphology, endings, natural language processing.

1 Introduction

Turkic languages make up a family including more than 35 languages [1], which are spoken by more than 160 millions of people across several countries. The Turkic group of languages includes state languages like Azerbaijan, Kazakh, Kyrgyz, Uzbek, Turkish, Turkmen. The languages of the subjects of the states are Altai, Balkar, Bashkir, Karakalpak, Crimean Tatar, Kumyk, Nogai, Tatar, Tuvan, Uyghur, Khakass, Shor, and Yakut.

In the modern world, the world of globalization, languages and migration are posing crucial challenges to the society. As the importance of favoring the effective communication between people has increased dramatically, artificial intelligence is seen, in its various forms, as a key enabling factor for circulating, sharing and accessing knowledge across languages and cultures. One of the cutting edge areas of artificial intelligence is natural language processing (NLP), whose tasks include, among others: word stemming, morphological analysis, text segmentation, syntactic tagging (POS-tagging), machine translation, language understanding, information retrieval, summarization, information extraction.

However, there are thousands low resource languages in the world without NLP using. Now in NLP area there are two group of computational models and methods for processing languages: finite-state transducers (FST) and machine learning methods. FST group require using user-oriented programming language for description source data for new languages, what is not easy for linguists. Second group, machine-learning group require good volume of electronic source data for machine learning, what there are not for many low resource languages.

A contribution of this paper is the new computational model of morphology based on the complete sets of endings (CSE-model) for the Turkic languages. The proposed approach allows the user to use universal (data-driven) algorithms and programs for a number of NLP tasks, such as determining the roots of words, morphological text analysis and text segmentation. One of its key features of this approach is that for a new language, only the linguistic resource of that language must be prepared in the form of a computational relational data model. Then a universal program is used for the corresponding task, driven by the developed data. This approach is particularly important for the large number of low-resource languages, for which the training material available for data-driven solutions is still insufficient. For these languages, the societal impact of our contribution lies in the fact that it offers a valid alternative to ease the creation of dedicated

core linguistic processors, deploy them to perform high-level NLP tasks and, in turn, favor communication. Therefore, the proposed methodology is that it is oriented towards linguists for improving NLP level of their languages area.

The remainder of this paper is organized as follows. Section 2 provides an overview of the previous works conducted in the field of models of natural languages morphology, approaches to the NLP tasks of stemming, segmentation, and morphological analysis of a natural languages. Section 3 describes the proposed methodology based on CSE-model on the example of Kazakh. Section 4 describes a computational data model, algorithms and programs for stemming of words on example of Kazakh. Three versions of stemming algorithms, programs are considered: lexicon-free stemming, stemming with stemming dictionary and stemming with stemming dictionary and stop-word dictionary. Section 5 describes a computational data model and algorithm for morphological segmentation based on CSE-model on example of Kazakh. Section 6 describes computational data model and algorithm for morphological analysis based on CSE-model on example of Kazakh. Section 7 describes experimental results and its analysis. Finally, section 8 presents the conclusions and suggests for future work.

2 Related works

There are three generally accepted models of natural languages morphology [2], namely: 'Item and Arrangement' (IA-model); 'Item and Process' (IP-model); 'Word and Paradigm' (WP-model).

The IA-model focuses on the agglutinative character of word forms. Its main modeling tool performs a linear segmentation of word forms into morphemes. Considering morphemes as its minimal units of grammatical description, the IA-model is well suited for describing the morphology of agglutinative languages.

The IP-model focuses on the concept of the dynamic nature of allomorphs, introducing one or more levels of word forms representation. Each morpheme of a word form necessarily has a single deep representation, as well as rules for transition to more superficial levels of representation, taking into account the context, at which allomorphic variation of the morpheme's representation is possible. Thus, depending on the context, the surface representations of the word form's morphemes will differ. Considering morphemes and symbols of deep representation ("deep phonemes") as its minimal units of grammatical description, the IP-model makes it possible to simplify the description of inflected languages' morphology.

The WP-model focuses on the concept of inflection by paradigm. In this morphology model, the word is considered as a whole rather than a combination of a stem and an ending. Inflection in the WP-model is considered by the similarity, and the minimal unit of grammatical description is the word form.

In practice, in the implementation of natural language processing tasks, models and methods of finite-state transducers (FST) and machine learning methods are actively used.

In FST methods the state-in-arts methods is two-level (TWOL) morphology computational model [3], which is mainly based on the IP-model for morphology. Software tools have been developed for the implementation of this technology, which are used for many languages. To use these tools, special user interface languages have been developed for the initial data (the rules of the two-level morphology technology). However, mastering and using a custom language for specifying the initial data for rule-based methods based on two-level morphology is a rather laborious process. From the point of view of the author this is a major obstacle for the widespread use of rule-based technologies by linguists for stemming, segmentation and morphological analysis, especially for low-resource languages.

Different from the previous paradigms, in the computational model of morphology (CSE- model) proposed in this paper, the minimal units of grammatical description of morphology are the word endings and stems. The word endings can be represented by a sequence of morphemes or, in the simplest case, one single morpheme. The endings in CSE-model is considered as a whole.

Currently, there are various approaches to the NLP tasks of stemming, text segmentation and morphological analysis.

Stemming has been addressed by means of both rule-based and machine learning methods [4].

For the segmentation task, two well-known solutions are BPE (Byte Pair Encoding) and MORFESSOR [5, 6]. The BPE method, which is based on a combinatorial-statistical approach, has shown poor results when dealing with agglutinative languages. MORFESSOR is also based on a statistical approach but, at the same time, it requires specifying a list of language affixes.

For morphological analysis, the many published works propose either rule-based methods, or feature-based methods or neural network solutions [7, 8].

Feature-based methods and neural networks methods for solving NLP problems require the preparation of a significant amount of initial data for training, which presents a critical bottleneck in low-resource language settings.

In light of previous considerations, in this paper we propose to use a CSE morphology model based on the enumeration of a complete set of language's endings. Based on the CSE-model of morphology, we developed computational relational data models for stemming, segmentation and morphological analysis, which universal (data-driven) algorithms and programs have been developed for these tasks.

3 Methodology

3.1 CSE – model of morphology

The construction of the CSE model of morphology, and its use for language analysis, are based on the derivation of the complete set of endings for a given language. In addition, in the full version of the model, it is necessary to collect a set of language stems, which it is possible to obtain the complete set of word forms for the language of interest. Thus, to describe the morphology of a language, it is necessary to specify either a complete set of word forms, or a complete set of endings and stems. The latter option is of course preferable, as the presentation of morphology with a complete set of endings and stems is more economical in terms of the volume of description than listing all possible word forms [9]. The grammatical dictionary of Zaliznyak [10] can be attributed to the morphology's model of enumeration of word forms.

Let's now consider the process of inferring a complete set of endings by taking the Kazakh language as an example.

All the Kazakh affixes can be divided into two classes: affixes to nominal stems (nouns, adjectives, numbers) and affixes to verb stems (verbs, participles, gerund, moods, voices).

The scheme for inferring the endings for each class of affixes is considered separately. However, the following four-step procedure is the same for all the cases:

– determination of a combination of possible placements of basic affixes' types;
– selection of a placement for basic affix types (performed by checking their semantic acceptability in the language);
– enumeration of possible variants of endings for each variant of a semantically acceptable placement of basic affix types;
– arrangement of endings into a complete set of endings for a given language.

3.1.1 Determination of a combination of possible placements of basic affixes' types.

Let's first consider the scheme of derivation of combinations of possible placements of basic affix types on example of nominal stems [9]. The set of affixes to the nominal stems of words in the Kazakh language has four types: – plural affixes (denoted by K); – possessive affixes (denoted by T); – case affixes (denoted by C); – personal affixes (denoted by J). The stem will be denoted by S. Let's consider all possible variants for placements of affixes' types: from one type, from two types, from three types and from four types. The number of placements is determined by the formula:

$$A_{nk} = \frac{n!}{(n-k)!}. \quad (1)$$

Then, the number of placements will be determined as follows: $A_{41} = 4!/(4-1)!=4$; $A_{42} = 4!/(4-2)!=12$; $A_{43} = 4!/(4-3)!=24$; $A_{44} = 4!/(4-4)!=24$.

There are hence 64 possible placements.

3.1.2 Selection of semantically acceptable placements of basic affixes' types.

Let's now consider the selection of semantically acceptable placements of affixes' types [9]. The basic affixes' type (K, T, C, J) are all semantically valid by definition.

Placements of two types of basic affixes can be as follows:

KT, TC, CJ, JK
KC, TJ, CT, JT
KJ, TK, CK, JC.

The analysis of the semantics of placements of the two types of affixes shows that only the following placements are acceptable (in bold): KT, TC, CJ, KC, TJ, KJ. Indeed, for example, TK – after possessive affixes plural affixes are not used, CK – after case affixes, plural affixes are not used, JC- after personal affixes, case affixes are not used, ST – after case affixes, possessive affixes are not used, JT- after personal affixes, possessive affixes are not used.

The placement of three affix types for semantic assumption is checked according to the rule:

if a placement of three types contains the invalid placement of two types, then that placement is not valid.

Then, there will be 4 acceptable placements of three affix types (KTC, KTJ, TCJ, KCJ).

The valid placement of the four affixes' types is 1 (KTCJ).

Based on the above rules, the number of acceptable placements from one type is 4, from two types is 6, from three types is 4, and from four types is 1. The total number of acceptable placements for words with nominal stems is hence 15.

3.1.3 Enumeration of possible endings.

For the enumeration of possible endings, let's consider the example of placement of the type KT of the nominal stem [9]. The possible endings for placement type KT (Plural – Possessive) are shown in Table 1. The combinations of possible endings for KT: K*T= (6 affixes K) *(5 affixes (different) T), which results in 30 endings of KT. For choosing the affixes in T for each affixes of K, we refer to the harmony rules of the Kazakh language.

Table 1 – Inferring of endings for placement type KT (Plural – Possessive)

	Affixes type K	Affixes type T		Number of endings
		Singular	Plural	
Examples	dar- der- lar- ler- tar- ter-	- ym, im	- ymyz, imiz	6*5=30
		- yń, iń	- yńyz, ińiz	
		- yńyz, ińiz	- yńyz, ińiz	
		- y, i	- y, i	
ana-	-lar-	ym,yń,yńyz,y	- ymyz	5
ini-	-ler-	im,iń,ińiz,i	- imiz	5
at-	-tar-	ym,yń,yńyz,y	- ymyz	5
ıt-	-ter-	im,iń,ińiz,i	- imiz	5
ań-	-dar-	ym,yń,yńyz,y	- ymyz	5
pán-	-der-	im,iń,ińiz,i	- imiz	5

3.1.4 Arrangement of variants of endings into a complete set of endings of a given language.

The assembly of endings options into a complete set of endings for a given language results from combining all the derived language endings into a single list of language endings. The complete set of endings for a language, in conjunction with a variety of a language stems, determines the morphology model for the given language.

Based on the four-step process described above, the complete set of inferred endings includes: 4679 endings for Kazakh [11], 4768 endings for Kyrgyz [13], and 747 endings for Uzbek [12].

3.2 Computational data models for segmentation and morphological analysis.

Based on the proposed CSE-model for morphology, we now describe how dedicated models for text segmentation and morphological analysis have been built.

The computational data model for text segmentation is a relational (table) data model, consisting of two columns [14]. The first "endings" column contains the complete set of language endings, while the second "segmented endings" column contains the endings of the language, segmented into affixes (Table 2).

Table 2 – Computational data model for the segmentation of Kazakh endings (fragment)

The word endings	The endings as sequence of affixes	Examples
gandarmensizder	gan@@dar@@men@@sizder	bar-gandarmensizder (you are with people who going)
largamyn	lar@@ga@@myn	apa-largamyn (I am to my sisters)
arlardasyndar	ar@@lar@@da@@syndar	ait-arlardasyndar (You are with whom will speak)
atyndargamyn	atyn@@dar@@ga@@myn	bar-atyndargamyn (I am to whom who will go)
rlergemin	r@@ler@@ge@@min	tole-rlergemin (I am to whom who will pay)

The computational data model for morphological analysis is a relational (table) data model, consisting of two columns. Here, too, the first "endings" column contains the full set of endings of the language, while the second column is a sequence of morphological characteristics describing the morphological analysis of the corresponding endings (Table 3). To describe the morphological characteristics, the tags by the Apertium rule-based MT system are used [15].

Table 3 – Computational data model for morphological analysis of Kazakh (segment of table)

Endings	Morphological analysis	Comments
largamyn	<NB>*lar<pl> *ga<dat>*myn<p1>	NB – nominal base type; pl – plural; dat – dative case; p1- 1-st person
gandarmensizder	<VB>*gan<pp>*dar <pl>* men<inst>*sizder<p2><frm>	VB-verbal base; pp-past participle; pl – plural; inst- instrumental case; p2 – 2nd person; frm- formality
arsyn	<VB>*ar<fut> *syn<p2>	VB-verbal base; fut – future tense; p2 – 2nd person

Based on the proposed approach, each new language will have its own computational data models for segmentation and morphological analysis. The following paragraphs will describe the universal algorithms for stemming, segmentation, morphological analysis based on the proposed CSE-model for morphology and the previously discussed computational data models.

3.3 Lexicon-free stemming algorithm according to the CSE morphological model.

The main concept of lexicon-free stemming based on the CSE-model of morphology is described below. The first step consists in finding an assumed ending of maximum length for a given input word. This will be at most two symbols less than the length of the word, as we assume that the stem cannot contain less than two symbols. The hypothesized ending of the given word is searched in the list of possible endings. If the ending is not found in the list, then we proceed by decreasing the length of the hypothesized ending. Accordingly, the hypothesized ending for the word is decreased by one symbol on the left-hand side, and this symbol is appended to the hypothesized stem of the word. The process is iterated by searching again the received ending in the list of possible endings. The above steps are repeated until the hypothesized ending is found in the list of endings or the length of the hypothesized ending becomes zero.

In the following pseudo-code representation, e(w) is the ending of the analyzed word w, st(w) is the stem of w, L(w) is the length of w, L[e(w)] is the calculated length of the ending.

The steps of the lexicon-free stemming algorithm are the following:

1. Calculation of L(w).
2. Calculation of the maximum length of an ending of the analyzed word: L[e(w)] = L(w) – 2, where 2 is the minimum length of the word stem.
3. Selection of the ending e(w) of the length L[e(w)] for the analyzed word w.
4. Search e(w) on matching with an ending from the list of endings. If it matches, then the stem of the word is determined: st(w) = w – e(w). Go to step 7.
5. Otherwise, the calculated length of the ending of the analyzed word is decreased by one: L[e(w)] = L[e(w)] – 1.
6. If L[e(w)] < 1, then word w is without the ending. Go to step 7. Otherwise, go to step 3.
7. End.

At the beginning of the algorithm, a given word is checked in a list of stop-words for the language of interest. If the input word appears in the list, then the processing terminates.

3.4 Stemming algorithm with stems lexicon according to the CSE morphological model.

To ensure a higher quality of our stemming algorithm, an improved version has also been developed by using a list of language stems (stems lexicon). The difference between this algorithm and the lexicon-free stemming algorithm described in the previous section is that, once the stem for an input word is selected, its presence is also checked in the stem lexicon.

The steps of the lexicon-enhanced stemming algorithm are described in the following pseudo-code:

1. Calculation of L(w).

2. Calculation of the maximum length of an ending of the analyzed word: L[e(w)] = L(w) – 2, where 2 is the minimum length of the word stem.

3. Selection of the ending e(w) of the length L[e(w)] for the analyzed word w.

4. Search e(w) on matching in the list of endings. If it matches, then the stem of the word is selected: st(w) = w – e(w).

5. Search stem st(w) on matching in the list of stems of language. If it matches, then go to 8;

6. Otherwise, the calculated length of the ending of the analyzed word is decreased by one: L[e(w)] = L[e(w)] – 1.

7. If L[e(w)] < 1, then word w is without the ending. Go to step 8. Otherwise, go to step 3.

8. End.

3.5 The universal algorithm for segmentation of words according to the CSE-model of morphology.

This algorithm includes two stages: 1) the splitting of a given word into a stem and an ending, and 2) the segmentation of the word ending into component affixes.

The first stage of the algorithm, word stemming, is described in the previous subsections 3.3, 3.4.

The second stage, the segmentation of the word ending into affixes, is realized using a single state transducer, presented as the decision table of segmented affixes for given ending (Table 2). The example of the morphological segmentation of Kazakh words is presented on Table 4.

Table 4 – The example of the morphological segmentation of Kazakh words

Examples	Stemming	The segmentations of endings
bargandarmensizder (you are with people who going)	bar (go-stem)-gandarmensizder (ending)	gan@@dar@@men@@sizder
apalargamyn (I am to my sisters)	apa (sister-stem)-largamyn	lar@@ga@@myn
aitarlardasyndar (You are with whom will speak)	ait (speak-stem)-arlardasyndar	ar@@lar@@da@@syndar
baratyndargamyn (I am to whom who will go)	bar (go-stem)-atyndargamyn	atyn@@dar@@ga@@myn
tolerlergemin (I am to whom who will pay)	tole (pay-stem)-rlergemin	r@@ler@@ge@@min

3.6 The universal algorithm for morphological analysis of words according to the CSE-model of morphology.

This algorithm includes two stages: 1) the splitting of a given word into a stem and an ending, and 2) the morphological analysis of the word's ending.

Again, the first stage of the algorithm, word stemming, is the one described in subsections 3.3, 3.4.

The second stage, the morphological analysis of the word, is realized using a single state transducer, presented as a decision table of the morphology analysis for given ending (Table 3). The computational data model based on the CSE-model of morphology is determining tags' sequences for a complete set of word endings. Examples of morphological analysis performed with the computational data model based on the CSE-model of morphology are given in Table 4.

Table 5 – The example of the morphological analysis of Kazakh words

Word	The morphological analysis of given word	Comments
dostaryńyzdanmyn (I am from your friends)	^dos + <NB>*tar<pl> *yńyz<p2><frm> *dan<abl>*myn<p1>	dos- friend (stem); NB – nominal base type; pl – plural; p2 – 2nd person; frm- formality; abl – ablative case; p1- 1-st person

kelgensizder (you are who came)	^kel + <VB>*gen<pp>*siz<p2><frm> *der<pl>	kel- come (stem); VB-verbal base; pp-past participle; p2 – 2nd person; frm- formality; pl – plural
bararsyn (you will go)	^bar + <VB>*ar<fut> *syn<p2>	bar- go (stem); VB-verbal base; fut – future tense; p2 – 2nd person

3.7 The methodology of using the proposed CSE-model of morphology and the developed universal NLP programs for a new language.

The methodology for using the proposed CSE-model of morphology and the developed universal NLP programs for a new language includes:

- building the CSE-model of morphology for a new language according to the method described in subsection 3.1;
- using the lexicon-free stemming program to generate a list of stems for a new language according to the procedure described in subsection 3.3;
- using the stemming program supported by the stem lexicon for the new language according to subsection 3.4;
- creating the segmentation and morphological analysis computational models according to subsection 3.2;
- using the universal segmentation, morphological analysis programs for the new language's computational data models.

4 Experiments

The experiments with the proposed CSE-model for morphology were carried out on three languages (Kazakh, Uzbek and Kyrgyz [12, 13, 14]), by developing the universal programs for stemming, segmentation and morphological analysis [16, 17].

For the Kazakh language, the computational data models have been developed for the stemming, segmentation and morphological analysis. The experiments in the Kazakh and Kyrgyz languages were carried out for the stemming and segmentation [17]. The experiments were carried out for the stemming in the Uzbek language [18]. Overall, the accuracy measure for stemming and segmentation achieved 80-90%.

5 Conclusion and future works

This paper proposed a methodology for the development and use of universal approach for stemming, segmentation and morphological analysis based on a new morphology model (CSE-model) on the complete set of endings. For each of the considered NLP problems, a computational relational data model has been developed. Specifically, relational data models were built and evaluated for three languages: Kazakh, Kyrgyz and Uzbek. Experimental results showed a high efficiency of the developed technology for solving the considered NLP tasks. The advantage of the proposed methodology is that it is oriented towards linguists. In order to solve the problems of stemming, segmentation, morphological analysis, it only requires i) building a complete set of language's endings for stemming using the described method, ii) building an endings segmentation table for the segmentation task, and iii) constructing a table of morphological analysis of endings for the task of morphological analysis and use the appropriate universal program. Future work is planned both in the direction of increasing the effectiveness of the developed algorithms and programs, and in the direction of using this methodology for other languages of the Turkic group.

Acknowledgement

The author would like to express gratitude to Marco Turchi and Matteo Negri for advice on improving this manuscript.

References

1. Gutman A. and Avanzati B. (2013). The languages gulper. Turkic languages. http://www.languagesgulper.com/eng/Turkic.html.
2. Plungyan V.A. (2003). Common morphology: Introduction to problematics: Educational manual. 2-d edition, edited.-M.: Editorial UPCC. – 384 p. (in Russian)
3. Koskenniemi K. (1983). Two-level morphology: A general computational model of word form recognition and production. Tech. rep. Publication No. 11. Department of General Linguistics. University of Helsinki.
4. Moral C., De Antonio A., Imbert R. and Ramírez J. (2014). A survey of stemming algorithms in information retrieval // IR information research. – Vol.9. -No 1. – http://informationr.net/ir/19-1/paper605.html#.X7tJjc0zY2w

5. Sennrich R., Haddow B., Birch A. (2016). Neural machine translation of rare words with subword units. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, vol. 1, 1715-1725 pp.

6. Creutz M., Lagus K. (2002). Unsupervised discovery of morphemes. Proceedings of the ACL 02 workshop on Morphological and phonological learning, Volume 6, 21-30 pp.

7. Kessikbayeva, G., Cicekli, I. (2014). Rule Based Morphological Analyzer of Kazakh Language. Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM, Baltimore, Maryland USA . 46–54 pp.

8. Morita H., Kawahara D., Sadao Kurohashi S. (2015). Morphological Analysis for Unsegmented Languages using Recurrent Neural Network Language Model. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2292–2297 pp.

9. Tukeyev U. (2015). Automaton models of the morphology analysis and the completeness of the endings of the Kazakh language. Proceedings of the international conference "Turkic languages processing" TURKLANG-2015 September 17–19, Kazan, Tatarstan, Russia, 91-100 pp (in Russian)

10. Zaliznyak's grammar dictionary. https://gufo.me/dict/zaliznyak (in Russian)

11. Tukeyev U., Karibayeva A. (2020). Inferring the Complete Set of Kazakh Endings as a Language Resource. In: Hernes M., Wojtkiewicz K., Szczerbicki E. (eds) Advances in Computational Collective Intelligence. ICCCI 2020. Communications in Computer and Information Science, vol 1287. Springer, Cham. https://doi.org/10.1007/978-3-030-63119-2_60

12. Matlatipov S., Tukeyev U., Aripov M. (2020). Towards the Uzbek Language Endings as a Language Resource. In: Hernes M., Wojtkiewicz K., Szczerbicki E. (eds) Advances in Computational Collective Intelligence. ICCCI 2020. Communications in Computer and Information Science, vol 1287. Springer, Cham. https://doi.org/10.1007/978-3-030-63119-2_59

13. Toleush A., Israilova N., Tukeyev U. (2021) Development of Morphological Segmentation for the Kyrgyz Language on Complete Set of Endings. In: Nguyen N.T., Chittayasothorn S., Niyato D., Trawiński B. (eds) Intelligent Information and Database Systems. ACIIDS 2021. Lecture Notes in Computer Science, vol 12672. Springer, Cham. https://doi.org/10.1007/978-3-030-73280-6_26

14. Tukeyev U., Karibayeva A. and Zhumanov Zh. (2020). Morphological Segmentation Method for Turkic Language Neural Machine Translation. Cogent Engineering, Volume 7, 2020 – Issue 1 https://doi.org/10.1080/23311916.2020.1856500 .

15. Forcada M. L. et al. Apertium: A free/open-source platform for rule-based machine translation. Machine Translation 25(2):127-144 pp., DOI: 10.1007/s10590-011-9090-0.

16. http://github.com/NLP-KAZNU

17. Tukeyev U., Karibayeva A., Turganbayeva A., Amirova D. (2021) Universal Programs for Stemming, Segmentation, Morphological Analysis of Turkic Words. In: Nguyen N.T., Iliadis L., Maglogiannis I., Trawiński B. (eds) Computational Collective Intelligence. ICCCI 2021. Lecture Notes in Computer Science, vol 12876. Springer, Cham. https://doi.org/10.1007/978-3-030-88081-1_48.