

Finite-state morphological transducers for three Kypchak languages

Jonathan North Washington[†], Ilnar Salimzyanov[‡], Francis M. Tyers^{*}

[†]Departments of Linguistics and Central Eurasian Studies
Indiana University
Bloomington, IN 47405 (USA)
jonwashi@indiana.edu

[‡]Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart
Stuttgart (Germany)
ilnar@ilnar

^{*}HSL-fakulteha
UiT Norgga árkálaš universitehta
9019 Romsa (Norway)
francis.tyers@uit.no

Abstract

This paper describes the development of free/open-source finite-state morphological transducers for three Turkic languages—Kazakh, Tatar, and Kumyk—representing one language from each of the three subbranches of the Kypchak branch of Turkic. The finite-state toolkit used for the work is the Helsinki Finite-State Toolkit (HFST). This paper describes how the development of a transducer for each subsequent language took less development time. An evaluation is presented which shows that the transducers all have a reasonable coverage—around 90%—on freely available corpora of the languages, and high precision over a manually verified test set.

Keywords: Kazakh, Tatar, Kumyk, morphology, transducer

1. Introduction

This paper describes the development of morphological transducers for three closely related languages: Kazakh, Tatar, and Kumyk.

The transducers for these languages

A finite-state transducer which models the morphology of a language has a wide variety of usages. For example, for creating proofing tools such as spellcheckers and grammarcheckers, and for machine translation.

2. Languages

The three languages for which transducers were developed belong to the Northwestern branch of the Turkic family, which is often referred to as the Kypchak branch. This branch can be divided into three subbranches. Kumyk is a member of the Western Kypchak group, Tatar is a member of the Northern Kypchak group, and Kazakh is a member of the Southern Kypchak group (Johanson, 2006, 82-83). As such, each of these three languages represents one of the three branches of Kypchak. The geographic distribution of the languages is shown in map 1.

These languages have different amounts of linguistic influence from other Turkic branches (e.g., moderate Oghuz (SE) influence in the Western group, slight Oghuz influence in the Northern group) and from Mongolic languages (moderate influence on the Southern group, lighter in the other groups), and all have heavy influence from Persian.

2.1. Kazakh

Kazakh /qazaq/ is spoken primarily in Kazakhstan, where it is the national language, where it is co-official with Russian. Large communities of native speakers also exist in China, neighbouring Central-Eurasian republics, and Mongolia. Ethnologue estimates the total number of speakers to be around 8 million (Lewis et al., 2013).

2.2. Tatar

Tatar /tatar/ is spoken in and around Tatarstan, by approximately 5.4 million people (Lewis et al., 2013). It is co-official with Russian in Tatarstan — a republic of the Russian Federation. A majority of native speakers of both languages are bilingual in Russian.

2.3. Kumyk

Kumyk /qumuq/ is spoken in Dagestan, a republic of the Russian Federation, where it is co-official with a number of other national languages (Lewis et al., 2013). There are approximately 430 thousand speakers (Lewis et al., 2013).

3. Background

3.1. Morphological transducers, previous work

The objective of a morphological transducer is twofold: firstly to take surface forms (e.g., алдым) and generate all possible lexical forms, and secondly to take lexical forms (e.g., ал<v><tv><ifi><p1><sg>, алд<n><px1sg><nom>, etc.) and generate one or more



Map 1: The three sub-branches of Kypchak (North, South, West), roughly divided with black lines, showing the geographic distribution of the three languages for which transducers were developed: Tatar (tat), Kazakh (kaz), and Kumyk (kum).

surface forms. As they are implemented as finite-state transducers, they are reversible by default. For more information on using finite-state transducers for morphological analysis and generation, the reader is referred to Beesley (2003).

The transducers were designed based on the Helsinki Finite State Toolkit (Linden et al., 2011) which is a free/open-source reimplement of the Xerox finite-state toolchain, popular in the field of morphological analysis. It implements both the **lexc** formalism for defining lexicons, and the **twol** and **xfst** formalisms for modeling morphophonological rules. It also supports other finite state transducer formalisms such as **sfst**. This toolkit has been chosen as it – or the equivalent XFST – has been widely used for other Turkic languages, such as Turkish (Çöltekin, 2010), Crimean Tatar (Altintas, 2001), Turkmen (Tantug et al., 2006), and Kyrgyz (Washington et al., 2012), and is available under a free/open-source licence.

The authors learnt of another Kazakh morphological transducer in existence (Бекманова & Махимов, 2013) only after this paper was submitted and our transducer was released. We have not yet been able to evaluate this system or compare it to ours.

Creating morphological transducers in the above-mentioned formalisms involves encoding linguistic knowledge about the language in the formalisms. The **lexc** and **twol** formalisms resemble linguistic formalisms, allowing the coders to work with abstractions resembling linguistic categories such as lexemes, morphemes, phonemes, and even archiphonemes.

3.2. Description

The transducers are available / under development in apertium’s subversion repository,¹ in the directories apertium-kaz, apertium-tat, and apertium-kum. The revision of the entire subversion repository that the

numbers (stem counts, evaluation, etc.) in this paper represent is r48137.

4. Methodology

4.1. Development effort

The three transducers discussed in this paper are for Kazakh, Tatar, and Kumyk. The Kazakh and Tatar transducers were originally created as part of an experimental Kazakh-Tatar machine translation system in December of 2010. The Kazakh transducer was expanded during Google Code-In 2010 and 2011, and the Tatar transducer was expanded as part of a prototype Tatar and Bashkir machine translation system (Tyers et al., 2012). The Kazakh-Tatar machine translation system, along with the two transducers, was expanded to production-level quality as part of a Google Summer of Code project in 2012 (Salimzyanov et al., 2013).

The Kumyk transducer was developed starting at the beginning of October, 2013 as experiment to see how difficult it would be to extend lessons learned from the development of the Tatar and Kazakh transducers to a related language. While the Kazakh and Tatar transducers took around six months work to reach their current coverage level, the Kumyk transducer only took a couple weeks to reach this level of quality (see Figure 1). This paper explores how the development of the Kumyk transducer benefitted from knowledge gained from the development of the Tatar and Kazakh transducers.

The morphotactics² of Turkic languages are complex enough that even a linguist who is fluent in the language and has a good linguistic understanding of it may not understand how exactly all morphemes combine. Native speakers educated about the morphology of their languages also do not have an explicit knowledge of the complete morphotactics. Hence it often becomes necessary to use fieldwork methodology to

¹<https://svn.code.sf.net/p/apertium/svn/languages/>

²The morphotactics of a language is the way in which morphemes can be combined to create words.

elicit the full extent of the morphotactics, be this a linguist with little to no knowledge of a Turkic language working with a native speaker, or a native speaker who understands the extent of what knowledge is necessary to encode in the transducer. When there is no native speaker of a particular language available, the authors have found that information previously encoded about a closely related language or the intuitions of a speaker of a closely related language may be combined with the use of textual corpora to “elicit” information about the morphotactics of a language. Depending on the contents of corpus and chance, this may not result in a completely accurate model, but it is possible to be thorough.

The Kazakh morphotactics were originally developed based on the Kyrgyz transducer, which was co-authored by the first author, who is fluent in and has a good linguistic knowledge of Kyrgyz and two native speakers of Kyrgyz. The first author also developed the Kazakh morphotactics who is fluent in and has a good linguistic understanding Kazakh. The morphotactics of Tatar were developed for the most part by the third author, a native speaker of Tatar, who also worked to polish off the morphotactics of the Kazakh transducer.

In order to create the Kumyk transducer, we approached one of the major parts of speech at a time. We started with the nouns, copying the continuation lexica³ (nominal morphotactics) from the Kazakh transducer. The suffixes were then replaced with the Kumyk suffixes according to the grammar (Ольмесов, 2000). Where the grammar was not explicit regarding a suffix form, we looked the corpus for possible forms and at their contexts. The same process was done for verbs.

4.2. Transducer contents

Each transducer’s lexc source consists of lists of stems, with each stem pointing at a complex continuation lexicon containing the appropriate morphology for the type of stem.

The tagset for each transducer is designed to be compatible with the others. Each transducer consists of about 120 separate tags, of which close to 20 cover the main parts of speech (noun, verb, adjective, adverb, postposition, interjection, etc.). The remaining tags cover morphological subcategorisation for e.g. case, number, person, possession, transitivity, tense-aspect-mood, etc. The tags are represented as multicharacter symbols, between less-than < and greater-than > sym-

³A continuation lexicon is a set of morphemes, for example, in the Turkic languages there is a continuation lexicon for ‘case’ which includes the possible case suffixes.

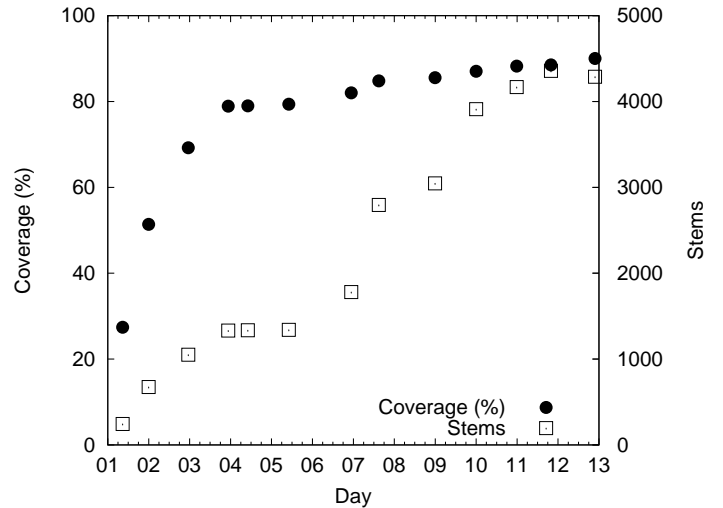


Figure 1: Number of stems and coverage of corpus over time. Time is measured in days starting from the 2nd October 2013. The graph shows that to reach 80% coverage, took around a week, and to reach 90% took another week.

Part of speech	Number of stems		
	Kazakh	Tatar	Kumyk
Noun	2463	2692	2588
Verb	1587	1333	262
Adjective	823	852	217
Proper noun	5412	3523	1444
Adverb	197	205	63
Numeral	63	67	45
Conjunction	52	50	15
Postposition	56	46	12
Pronoun	17	17	18
Determiner	42	37	9
Total:	10942	9009	4687

Table 1: Number of stems in each of the categories. The large number of noun stems as compared to other parts of speech in the Kumyk transducer can be explained by the relative ease of categorising nouns as opposed to adjectives, adverbs and verbs. Adding a noun essentially involves choosing between loan word and native word. Where adding stems from the other main categories requires more in depth categorisation.

bols. The tagset is quite extensive and still not entirely stabilised, so a full listing is not included here. However, the tags are listed in the source code of the transducers, along with comments describing their usage.

Table 1 lists the number of stems of the primary categories in each transducer.

Kazakh	Tatar	Kumyk
Құдай Өзінің жаратқандарының бәріне қарап, өте жақсы екенін көрді.	Аллаһ Үзе яраткан нәрсәләргә карап, аларның бик яхшы икәнән күрде.	Аллаһъ Обзю яратгъан затлагъа къарап, олар бек яхшы экенин гёрген.
Құдай<n><nom> Өз<prn><ref><px3sp><gen> жарат<v><tv><ger_past><pl><px3sp><gen> бәрі<prn><qnt><px3sp><dat> қара<v><tv><gna_perf> ,<cm> — өте<adv> жақсы<adj> е<cop><ger_past><px3sp><acc> көр<v><tv><ifi><p3><sg> ,<sent>	Аллаһ<n><nom> Үз<prn><ref><px3sp><nom> ярат<v><tv><gpr_past> нәрсә<prn><itg><pl><dat> кара<v><tv><gna_perf> алар<prn><pers><p3><pl><gen> бик<adv> яхшы<adj> и<cop><ger_past><px3sp><acc> күр<v><tv><past><p3><sg> ,<sent>	Аллаһъ<n><nom> Обз<prn><ref><px3sp><nom> ярат<v><tv><gpr_past> зат<n><pl><dat> къара<v><tv><gna_perf> ,<cm> олар<prn><pers><p3><pl><nom> бек<adv> яхшы<adj> э<cop><ger_past><px3sp><acc> гёр<v><tv><past><p3><sg> ,<sent>

Table 2: An example of the output of each of the morphological transducers for the same sentence (“And God saw every thing that he had made, and, behold, it was very good.”, Genesis 1:31). The sentence may be very roughly glossed as “God own-his created things/everything-at looking, their very good being saw”. The output has been abbreviated to only show the appropriate tag in context, the actual output would be ambiguous.

4.3. Categorisation and tagset

The open categories in Turkic languages can be broadly split into two groups: nominals and verbals. The nominals group can be further split into nouns, adjectives and adverbs. We define nouns as stems that can be subjects of a finite verb, adjectives as stems that principally qualify nouns, and adverbs as stems that principally modify verbs.

Most of the stems in the lexicon can be used, with no extra suffixes in any of these functions, for example the adjective *жақсы* in Kazakh can be used as attributively, *жақсы кітап* ‘good book’, adverbially *жақсы ойнадыңыз* ‘(you) played well’, and substantively (as a noun) *‘жақсы келетін еді’* ‘(the) good (ones) would come’. This is a productive process and relevant to the morphotactics as an adjective used substantively may take the full range of case and possessive suffixes, and adjectives used adverbially will take a different set of clitics to adjectives used substantively and attributively.

However, this is not completely productive. A grammar which allows all adjectives to function as nouns and adverbs will overgenerate (and overanalyse). However, most grammars do not mention that there are exceptions to this productive process.

Other analysers for Turkic languages—such as TRmorph (Çöltekin, 2010)—approach this problem by positing a zero-derivation rule by which any noun can be ‘derived’ into an adjective or an adverb, any adjective can be ‘derived’ into a noun or adverb, etc.

Our approach is to describe this rather in terms of function (not unlike Hengeveld (1992)). We posit that nominals can be used either substantively, attributively or adverbially. Each part of speech has a ‘de-

fault’ function: nouns are by default substantive, adjectives are by default attributive, and adverbs, adverbial. When they are used outside this default use they receive a tag to mark their function. Where these functions are ambiguous, then they may be disambiguated in context.

The advantage of this approach as opposed to simply allowing one part of speech to derive into another is that it allows us to be more principled, and decreases overgeneration, for example, nouns used attributively will not take the whole adjective continuation (e.g. they will not take comparison, and cannot later substantivise). At the same time, it prevents having to have two lexemes, one for each usage.

There are certain lexemes which require more explicit categorisation, for example, the word for ‘today’⁴ cannot be used attributively without the presence of an extra morpheme -KI. Compound nouns using possessive morphology, such as Kazakh *ая райы* ‘weather’ are not well-formed without a possessive suffix.

4.4. Technolinguistic and orthographical issues

There is a class of phenomena encountered during the development of these transducers, united by the fact that the morphophonology needs information about the stem’s phonological form that is not provided by the orthographic representation. These phenomena were left by Washington et al. (2012) as future work, but have been extensively implemented in the transducers described by the current paper.

Types of these phenomena include acronyms, “irregular” vowel harmony, Russian loanwords, and numer-

⁴Tatar: бүген; Kazakh: бүгін; Kumyk: бүгюн. The word is a contraction of the words for ‘this’ and ‘day’.

als. A brief overview of each follows, along with examples, followed by details of our solution.

Ambiguous and ‘irregular’ harmony: In Tatar, the ‘yoticised’ vowel letters ⟨e⟩, ⟨я⟩, and ⟨ю⟩ each ambiguously represent a set of sounds. For example, ⟨e⟩ can represent the sounds of /j/ followed the sound of ⟨ə⟩ (as in ⟨егетләр⟩ ‘boys’) or ⟨ы⟩ (as in ⟨еллар⟩ ‘years’), and is also the non-initial orthographic variant of ⟨ə⟩ (as in ⟨дәресләр⟩ ‘lessons’). In Kumyk, ⟨ё⟩ and ⟨ю⟩ are used intervocalically to represent front rounded vowels (as in ⟨гёзлер⟩ ‘eyes’ and ⟨гюнлер⟩ ‘days’), but word-initially can represent ⟨й⟩ followed by either a front (as in ⟨юреклер⟩ ‘hearts’, ⟨ёнкюлер⟩ ‘darlings’) or a back (as in ⟨юлдузлар⟩ ‘stars’, ⟨ёллар⟩ ‘roads’) vowel. In Kazakh, the vowels /ə/ ⟨i⟩ and /ə/ ⟨ы⟩ followed by a glide /w/ ⟨y⟩ or /j/ ⟨й⟩ are written ⟨y⟩ and ⟨и⟩, depending on the glide; for example. The character ⟨ю⟩ is used in turn to represent ⟨й⟩ followed by either multi-phoneme value of ⟨y⟩.

Loan words: In Kumyk, the character ё in Russian words is pronounced as in Russian where in native words it represents the sound /ø/. These two pronunciations will select for different suffixes, the Russian pronunciation taking back vowel suffixes самолётлар ‘aeroplanes’, and /ø/ taking front vowel suffixes сёзлер ‘words’.

Acronyms and numerals: Acronyms and numerals are challenging as they will often be pronounced out loud in their non-abbreviated forms, for example in Kazakh, 30-дан ‘from thirty’, 5-тен ‘from 5’. The ablative suffix -DAn alters for the phonology, but in the numeral string there is no indication of how they should alter.

Dealing with these phenomena would not be necessary if we were setting out to develop a simple computational model of the phonology of the language. However, a wide-coverage morphological analyser and generator needs to be able to deal with all phenomena that are found in corpora.

Regarding numerals, work has been done for Finnish in the finite-state framework (Karttunen, 2006); however, this relies on converting all numerals to their fully spelt out form, which would involve complex operations on the transducer.

Our solution is to add phonological information at the end of morphemes that need it in the form of special “abstract letters” that trigger phonological processes at the morphophonological stage and are deleted.

For example, the string 5<num><subst><abl>⁵ has the morphotactic representation 5{ə}{c}>{D}{A}H, where {ə} and {c} stand for phonological triggers, > represents a morpheme boundary, and {D}{A}H is the representation of the ablative morpheme at the morphotactic level. The symbol {ə} signals that the following vowel needs to harmonise to a front unrounded vowel, and {c} signals that there is a final voiceless consonant. So, with 5{ə}{c} in the lexicon, and a consistent continuation lexicon specifying the underlying form of case affixes, the rules that operate on {D} and {A} are able to do produce the correct output form in each language, avoiding the incorrect default *5-дан.

5. Evaluation

We have evaluated the morphological analysers in two ways. The first was by calculating the naïve coverage⁶ and mean ambiguity on freely available corpora. The mean ambiguity measure was calculated by performing an evaluation of precision and recall on some smaller, hand-validated test sets.

5.1. Corpora

We tested the coverage of the Kazakh and Tatar analysers over three separate domains: encyclopaedic text,⁷ news,⁸ and religion.⁹ As there is currently no Wikipedia in Kumyk, we tested only news and religion.¹⁰

The coverage of each transducer over the various corpora is shown in table 3.

5.2. Precision and recall

Precision and recall are measures of the average accuracy of analyses provided by a morphological transducer. Precision represents the number of the analyses given for a form that are correct. Recall is the percentage of analyses that are deemed correct for a form (by

⁵The resulting forms, 5-тен^{kaz}, 5-тән^{tat}, and 5-ден^{kum} would be pronounced бестен^{kaz}, биштән^{tat}, and бешден^{kum}.

⁶Naïve coverage refers to the percentage of surface forms in a given corpora that receive at least one analysis. Forms counted by this measure may have other analyses which are not delivered by the transducer.

⁷The following Wikipedia dumps were used: kkwiki-20131006-pages-articles.xml.bz2.

⁸All content from <http://www.azattyq.org/> for 2010 was used for Kazakh, as well as all content from 2005 to 2011 on <http://tat.tatar-inform.ru> for Tatar.

⁹We used a Kazakh bible translation available from <https://kkitap.net/> and a Tatar translation of the New Testament available from <http://ibt.org.ru>

¹⁰The bible corpus is from <http://ibt.org.ru/> and the news corpus consists of all Kumyk content from <http://sh-tavisi.etnosmi.ru/>.

Language	Corpus	Tokens	Coverage (%)
Kazakh	Wikipedia	18.2M	85.1 ± 1.33
	News	3.2M	-
	Religion	849K	92.7 ± 1.57
	Average	-	-
Tatar	Wikipedia	128K	-
	News	4.6M	-
	Religion	205K	88.7 ± 0.5
	Average	-	-
Kumyk	Wikipedia	-	-
	News	287K	91.4 ± 0.7
	Religion	227K	92.0 ± 1.1
	Average	-	91.7 ± 0.7

Table 3: Corpora used for naïve coverage tests

comparing against a gold standard) that are provided by the transducer.

To calculate precision and recall, it was necessary to create a hand-verified list of surface forms and their analyses. We extracted 1,000 unique surface forms at random from a news corpus for each language, and checked that they were valid words in the languages and correctly spelt. Where a word was incorrectly spelt or deemed not to be a form used in the language, it was discarded and a new random word selected.

This list of surface forms was then analysed with the most recent version of the analyser, and each analysis was checked. Where an analysis was erroneous, it was removed; where an analysis was missing, it was added. This process gave us a ‘gold standard’ morphologically analysed word list of 1,000 surface forms with their analyses. The list is publically available for each language.

We then took the same list of surface forms and ran them through the morphological analyser once more. Precision was calculated as the number of analyses which were found in both the output from the morphological analyser and the gold standard, divided by the total number of analyses output by the morphological analyser.

Recall was calculated as the total number of analyses found in both the output from the morphological analyser and the gold standard, divided by the number of analyses found in the morphological analyser plus the number of analyses found in the gold standard but not in the morphological analyser.

The results for precision and recall are presented in table 4.

Language	Precision	Recall
Kazakh	98.61	57.98
Tatar	95.03	85.65
Kumyk	-	-

Table 4: Precision and recall

The low recall for Kazakh can be explained by the fact that the corpus is much bigger, giving more hapax words and proper names. There were 403 unknown words out of 1,000 in the Kazakh list. Of these 403, 160 were proper nouns, and 148 were common nouns. The lower precision for the Tatar transducer can be partly explained by the less transparent orthography of Tatar.

6. Future work

One direction for future work is to develop transducers for more languages. We have already constructed usable prototype transducers for three other Kypchak languages: Bashkir (North), Nogay (South), and Karakalpak (South). Since our ability to develop transducers is limited by availability of resources, including corpora in the languages and native-speaker consultants, the Western Kypchak languages (aside from Kumyk) have been more neglected by our team. However, these language communities would benefit from computational tools (such as spellcheckers) for their languages, and work on them may be bootstrapped from the existing transducers, so working on morphological transducers for these languages is also a priority.

The principle obstacle to increasing coverage of the lexicons is the categorisation of stems. Future work would be investigating ways of automatically categorising stems by subcategory. For example, verbs stems by transitivity; adjective stems by if they can be used adverbially or substantively; etc.

7. Conclusions

We have described morphological transducers for three Kypchak languages—one from each branch of Kypchak—including the development process and performance of the analysers. The development of the third transducer (for a related language) was substantially quicker than the first two as a result of being able to reuse large portions of the morphotactic description from the first two transducers.

Acknowledgements

We would like to thank the Google Code-in (2011) for supporting the development of the Kazakh transducer, and in particular the effort by Nathan Maxson. We would also like to thank the Google Summer of Code (2012) for supporting the development of both the Kazakh and the Tatar transducers.

The authors would also like to express their gratitude to Aida Sundetova and Ağarhim Sultanmuradov for assistance in evaluating precision and recall.

References

- Altintas, K. (2001). A morphological analyser for Crimean Tatar. *Proceedings of Turkish Artificial Intelligence and Neural Network Conference*.
- Beesley, Ken (2003). *Finite-state morphological analysis*.
- Hengeveld, Kees (1992). Parts of speech. In Michael Fortescue, Peter Harder, & Lars Kristoffersen (Eds.), *Layered structure and reference in a functional perspective*, Benjamins, pp. 29–55.
- Johanson, Lars (2006). History of Turkic. In Lars Johanson & Éva Á. Csátó (Eds.), *The Turkic Languages*, New York: Routledge, chap. 5, pp. 81–125.
- Karttunen, Lauri (2006). *Numbers and Finnish Numerals*, vol. 19, pp. 407–421.
- Lewis, M. Paul, Simons, Gary F., & Fennig, Charles D. (Eds.) (2013). *Ethnologue: Languages of the World*. Dallas, Texas: SIL International, seventeenth edn. <http://www.ethnologue.com>.
- Linden, Krister, Silfverberg, Miikka, Axelson, Erik, Hardwick, Sam, & Pirinen, Tommi (2011). *HFST—Framework for Compiling and Applying Morphologies*, vol. Vol. 100 of *Communications in Computer and Information Science*, pp. 67–85. ISBN 978-3-642-23137-7.
- Salimzyanov, Ilnar, Washington, Jonathan North, & Tyers, Francis M. (2013). A free/open-source Kazakh-Tatar machine translation system.
- Tantuğ, A.C., Adalı, E., & Oflazer, K. (2006). Computer analysis of Turkmen language morphology. *Advances in natural language processing, proceedings (Lecture notes in artificial intelligence)*, pp. 186–193.
- Tyers, Francis, Washington, Jonathan North, Salimzyan, Ilnar, & Batalov, Rustam (2012). A prototype machine translation system for Tatar and Bashkir based on free/open-source components. In *Proceedings of the First Workshop on Language Resources and Technologies for Turkic Languages at the Eight International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey.
- Washington, Jonathan North, Ipasov, Mirlan, & Tyers, Francis M. (2012). A finite-state morphological analyser for Kyrgyz.
- Çöltekin, Çağrı (2010). A freely available morphological analyzer for Turkish. *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC2010)*, pp. 820–827.
- Бекманова, Г. Т. & Махимов, А. (2013). Графематический и морфологический анализатор Казахского языка. pp. 192–200.
- Ольмесов, Нурамат Хайруллаевич (2000). *Сопоставительная грамматика кумыкского и русского языков*. Махачкала: ИПЦ ДГУ.