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

[†]Kazan Federal University
Kazan, Republic of Tatarstan
(Russian Federation)
ilnar.salimzyan@gmail.com

[‡]HSL-fakultehta
UiT Norgga árktalaš 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 sub-branches 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 closely-related 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 free/open-source morphological transducers for three closly related languages: Kazakh, Tatar, and Kumyk. Morphological transducers are computational models of the languages' morphology, and are used to output morphological analyses from word forms and vice-versa.

These transducers were all developed as part of the Apertium project, which is aimed at creating rule-based MT systems for lesser resourced languages. As such, the transducers were developed with the intent that they be used as morphological analysers and generators in RBMT systems. However, this sort of finite-state transducer has a wide variety of usages, including for creating proofing tools such as spellcheckers and grammarcheckers, morphological annotation for linguistic research, creating resources for learners and CALL, and lemmatised dictionary lookup.

All three transducers were designed as "ports" of Apertium's Kyrgyz morphological transducer (Washington et al., 2012). The Tatar and Kazakh transducers are used in a Kazakh–Tatar MT system (Salimzyanov et al., 2013).

This paper overviews the languages (§2.), provides background on these morphological transducers and related previous work (§3.), describes the development effort and contents of the transducers (§4.), evaluates the effectiveness of these transducers (§5.), and summarises the findings (§7.).

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 sub-branches. 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).[1] The geographic distribution of the languages is shown in map 1.

These languages display 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 display heavy influence from Persian.

2.1. Kazakh

Kazakh /qɑzɑq/ is spoken primarily in Kazakhstan, and 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).

[1] It is the professional opinion of the authors of this paper that Kyrgyz constitutes a fourth branch.

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. The Kypchak languages shown on the map are Tatar (`tat`), Kazakh (`kaz`), and Kumyk (`kum`). The other codes represent Bashkir (`bak`), Kyrgyz (`kir`), Karakalpak (`kaa`), Nogay (`nog`), Karachay-Balkar (`krc`), Urum (`uum`), Crimean Tatar (`crh`), and Karaim (`kdr`).

2.2. Tatar

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

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

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>`,[2] etc.) and generate one or more 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 reimplementation 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 Beesley (2003) – has been widely used for other Turkic languages, such as Turkish (Çöltekin, 2010), Crimean Tatar (Altintas, 2001), Turkmen (Tantuğ 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. The system is unfortunately not freely available so we have not 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.

The transducers are available / under development in apertium's subversion repository,[3] 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

[2] For a description of the tags used throughout this paper, please see Appendix A.

[3] https://svn.code.sf.net/p/apertium/svn/languages/

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 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 an 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 of weeks to reach the same level of coverage (see Figure 1). This article explores how the development of the Kumyk transducer benefitted from knowledge gained from the development of the Tatar and Kazakh transducers.

The morphotactics[4] 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 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

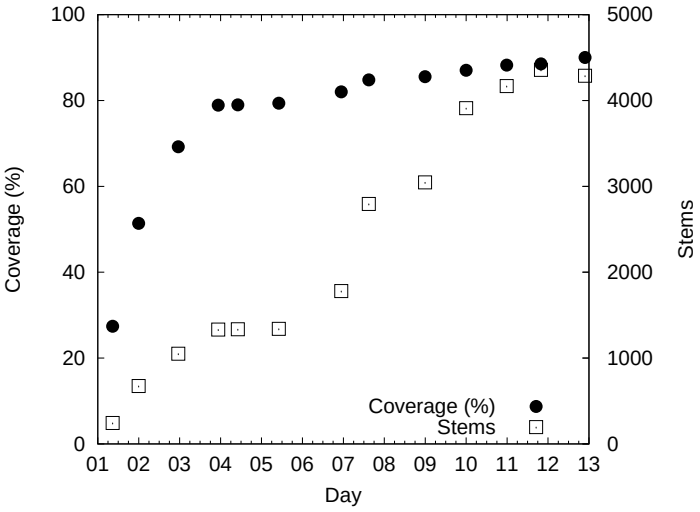

Figure 1: Number of stems and coverage of corpus over time for the Kumyk transducer. Time is measured in days starting from the 2nd October 2013. The graph shows that to reach 80% coverage, took around a week, and to reach 90% took another week.

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 lexica[5] (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 undertaken 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-

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

[5]A 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.

Part of speech	Number of stems		
	Kazakh	Tatar	Kumyk
Noun	2640	2795	2568
Verb	1470	1143	386
Adjective	754	816	219
Proper noun	5701	5361	1443
Adverb	171	177	63
Numeral	63	63	44
Conjunction	46	45	13
Postposition	50	43	12
Pronoun	32	28	17
Determiner	39	34	9
Total:	11224	10737	4845

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. Adding stems from the other main categories requires more in depth categorisation.

mood, etc. The tags are represented as multicharacter symbols, between less-than < and greater-than > symbols. 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.

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 than adjectives used substantively and attributively.

However, this is not completely productive, as not all adjectives can be used substantively or adverbially. A grammar which allows all adjectives to function as nouns and adverbs will overgenerate (and overanalyse). However, most grammatical descriptions of these languages do not mention that there are exceptions to this 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 ‘default’ 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, 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 that require more explicit categorisation, for example, the word for ‘today’[6] cannot be used attributively in Kypchak languages without the presence of an extra morpheme -KI. Compound nouns using possessive morphology, such as Kazakh аya райы ‘weather’, do not take additional morphology the same way other substantives in the language do, due to the presence of a final possessive morpheme (e.g., ‹аya райынан› ‘weather-`abl`’, not *‹аya райыдан›).

4.4. Orthography-phonology mapping 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) for future work,

[6]Tatar: бүген; Kazakh: бүгін; Kumyk: бугюн. The word is etymologically composed of the words for ‘this’ and ‘day’.

Kazakh	Tatar	Kumyk
Құдай Өзінің жаратқандарының бәріне қарап, өте жақсы екенін көрді.	Аллаһ Үзе яраткан нәрсәләргә карап, аларның бик яхшы икәнен күрде.	Аллагъ Оьзю яратгъан затлагъа къарап, олар бек яхшы экенин гёрген.
Құдай<n><nom>	Аллаһ<n><nom>	Аллагъ<n><nom>
Өз<prn><ref><px3sp><gen>	Үз<prn><ref><px3sp><nom>	Оьз<prn><ref><px3sp><nom>
жарат<v><tv><ger_past><pl><px3sp><gen>	ярат<v><tv><gpr_past>	ярат<v><tv><gpr_past>
бәрі<prn><qnt><px3sp><dat>	нәрсә<prn><itg><pl><dat>	зат<n><pl><dat>
қара<v><tv><gna_perf>	кара<v><tv><gna_perf>	къара<v><tv><gna_perf>
,<cm>	,<cm>	,<cm>
—	алар<prn><pers><p3><pl><gen>	олар<prn><pers><p3><pl><nom>
өте<adv>	бик<adv>	бек<adv>
жақсы<adj>	яхшы<adj>	яхшы<adj>
е<cop><ger_past><px3sp><acc>	и<cop><ger_past><px3sp><acc>	э<cop><ger_past><px3sp><acc>
көр<v><tv><ifi><p3><sg>	күр<v><tv><past><p3><sg>	гёр<v><tv><past><p3><sg>
.<sent>	.<sent>	.<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-to looking, their very good being saw". The output has been abbreviated to only show the appropriate tag sequence in context, the actual output would give multiple interpretations that would require further disambiguation. Refer to Appendix A for tag descriptions.

but many of them been implemented in the transducers described in the current paper.

Types of these phenomena include characters used to represent multiple phonological forms, loanwords, and acronyms and numerals. A brief overview of each follows, along with examples and details of our solution.

4.4.1. Ambiguous characters

In Tatar, the 'yoticised' vowel letters ‹е›, ‹я›, and ‹ю› each ambiguously represent a set of sounds. For example, ‹е› is the non-initial orthographic variant of ‹э› (as in ‹дәресләр› 'lessons'), but can also represent /j/ followed by the phoneme of ‹э› (as in ‹егетләр› 'boys') or ‹ы› (as in ‹еллар› 'years').

In Kumyk, ‹ё› and ‹ю› are used between consonants to represent front rounded vowels (as in ‹гёзлер› 'eyes' and ‹гюнлер› 'days'), but word-initially can represent ‹й› followed by either a front vowel (as in ‹юреклер› 'hearts', ‹ёнкюлер› 'darlings') or a back vowel (as in ‹юлдузлар› 'stars', ‹ёллар› 'roads').

In Kazakh, the vowels /ə/ ‹і› and /ə/ ‹ы› followed by a glide /w/ ‹у› or /j/ ‹й› are written ‹у› 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 ‹у›. For example, ‹қиюда› /qəjəwdɑ/ 'in the process of chopping down' and ‹киюде› /kəjəwd̪ʲə/ 'in the process of getting dressed' have back- and front-harmonising values, respectively, for ‹и› and ‹ю›.

These problems can often be solved by building the context into the morphophonological rules, though the addition of new contexts and exceptions to existing contexts results in rather complex `twol` rules.

4.4.2. Loanwords

In Kazakh, Tatar, and Kumyk, most Russian loanwords are spelled using Russian orthography, despite incompatibilities between Russian letter-to-phoneme mapping and that of the Turkic languages.

In Kumyk, the character ‹ё› is used between consonants to represent a mid front rounded vowel (e.g., ‹сёзлер› 'words'); however, in Russian words, ‹ё› is used after a consonant to represent the sound of ‹о› while indicating that the previous consonant is palatalised (e.g., ‹самолётлар› 'aeroplanes').

In Tatar, the character ‹и› normally represents a high front vowel (e.g., ‹галимнәр› 'scientists'), but in Russian words it harmonises as a back vowel (e.g., ‹артистлар› 'artists').

In Kazakh, ‹е› is a front diphthong (e.g., ‹елдің› 'country-`gen`'), but in e.g., family names with Russian morphology, it harmonises as a back vowel (e.g., ‹Назарбаевтың› 'Nazarbayev-`gen`').

Our solution for these consistent exceptions is to make a separate continuation lexicon for e.g., Russian nouns. This continuation lexicon adds a character that the phonology deletes, but that triggers phonological rules.

Since `twol` "phonological" rules can be understood to be applied all at one stage (as opposed to being ordered), rules triggered by these characters must make reference to abstract characters that exist on the input tape but are null on the output tape. This complicates existing rules that have been designed to ignore null characters on the output tape. Hence, the `twol` rules resulting from the combination of processes simultaneously triggered by and ignoring output-null charac-

ters can quickly become quite unwieldy.

4.4.3. 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 by (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>`[7] has the morphotactic representation 5{э}{с}>-{D}{A}н, where {э} and {с} stand for phonological triggers, > represents a morpheme boundary, and {D}{A}н is the representation of the ablative morpheme at the morphotactic level. The symbol {э} signals that the following vowel needs to harmonise to a front unrounded vowel, and {с} signals that there is a final voiceless consonant. So, with 5{э}{с} 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. This solution has the same issue as loanwords above.

5. Evaluation

We have evaluated the morphological analysers[8] in two ways. The first was by calculating the naïve coverage and mean ambiguity on freely available corpora. Naïve 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. The mean ambiguity measure was calculated as the average number of analyses returned per token in the corpus.

Language	Corpus	Tokens	Coverage (%)
Kazakh	Wikipedia	25.6M	85.61 ± 1.37
	News	3.8M	92.12 ± 2.72
	Religion	851K	92.49 ± 1.66
	Average	–	90.07 ± 1.91
Tatar	Wikipedia	159K	86.35 ± 2.17
	News	5.2M	89.75 ± 0.07
	Religion	382K	91.25 ± 2.55
	Average	–	89.12 ± 1.60
Kumyk	Wikipedia	–	–
	News	286K	91.10 ± 0.86
	Religion	227K	92.47 ± 1.03
	Average	–	91.78 ± 0.94

Table 3: Corpora used for naïve coverage tests

5.1. Corpora

We tested the coverage of the Kazakh and Tatar analysers over three separate domains: encyclopaedic text,[9] news,[10] and religion.[11] As there is currently no Wikipedia in Kumyk, we tested only news and religion.[12]

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 comparing against a gold standard) that are provided

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

[8] The versions of the analysers used were: r50547 for Kazakh, r50260 for Tatar, and r50300 for Kumyk.

[9] The following Wikipedia dumps were used: `kkwiki-20131006-pages-articles.xml.bz2` and `ttwiki-20130225-pages-articles.xml.bz2`.

[10] All 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.

[11] We used a Kazakh bible translation available from https://kkitap.net/ and Quran translation available from http://kuran.kz/, as well as a Tatar translation of the New Testament available from http://ibt.org.ru/ and Quran translation available from http://tanzil.net/.

[12] The bible corpus consists of Genesis and the New Testament, as available from http://ibt.org.ru/, and the news corpus consists of all Kumyk content from http://sh-tavisi.etnosmi.ru/.

Language	Precision (%)	Recall (%)
Kazakh	98.61	57.98
Tatar	95.03	85.65
Kumyk	-	-

Table 4: Precision and recall presented as percentages

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.

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 out-of-vocabulary 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 whether 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 Morphology*. CLSI.

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. Csató (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. In *Proceedings of MT Summit XIV*.

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. In *Proceedings of the 8th Conference on Language Resources and Evaluation, LREC2012*.

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. In *Proceedings of the 1st International Conference on Computer Processing of Turkic languages (TurkLang2013)*. pp. 192–200.

Ольмесов, Нураммат Хайруллаевич (2000). *Сопоставительная грамматика кумыкского и русского языков*. Махачкала: ИПЦ ДГУ.

A Glossary of symbols

Symbol	Description
n	Noun
v	Verb
adj	Adjective
adv	Adverb
prn	Pronoun
cm	Comma
cop	Copula
ifi	Past definite
past	Past
ger_past	Past gerund
gna_past	Past verbal adverb
gpr_past	Past verbal adjective
itg	Interrogative
nom	Nominative
gen	Genitive
dat	Dative
abl	Ablative
p1	First person
p3	Third person
px1sg	First person, singular possessive
px3sp	Third person possessive
pers	Personal
qnt	Quantifier
ref	Reflexive
sg	Singular
pl	Plural
tv	Transitive
sent	End-of-sentence marker