ACTA INFOLOGICA
dergipark.org.tr/acin

İSTANBUL UNIVERSITY PRESS

DOI: 10.26650/acin.842758

RESEARCH ARTICLE

Two Level Kazakh Morphology*

İki Düzeyli Kazak Morfolojisi

Züleyha Yiner[1], Atakan Kurt[2]

ABSTRACT

We present a comprehensive two level morphological analysis of contemporary Kazakh with implementation and a disambiguation test data set on the Nuve Framework. Our study differs from the similar studies in a number of ways: (i) Our study covers both derivational and inflectional morphology to a greater extend (ii) Our implementation consisting of orthographic rules, morphotactics, a root lexicon of roughly 24 thousand roots, a lexicon of roughly 150 suffixes is open source which can be downloaded, reviewed and tested. (ii) Roughly 10 thousand manually disambiguated parses are available as a morphological disambiguation data set. (iii) It is easily extensible meaning it can be modified or extended with new rules without any programming. (iv) we are able to tackle emerging problems quickly and easily since Nuve is maintained by our study group. (v) Our implementation can handle separately written morphemes or digraphs etc. directly. (vi) We also have a Turkish morphological parser/generator in Nuve for morphology based machine translation between Turkish and other Turkic languages since these closely related languages have a lot in common from lexical, morphological, and syntactic aspects.

Keywords: Kazakh Morphology, Natural Language Processing, Computational Linguistics

ÖZ

Bu çalışmada Çağdaş Kazakça'nın iki düzeyli kapsamlı bir morfolojisini sunulmuştur. Çalışma Nuve Çatısı üzerinde gerçeklenmiş ve belirsizlik giderme veri seti ile test edilmiştir. Çalışmamız benzerlerinden bir kaç yönden farklılık göstermektedir:(i) Çalışmamız hem yapım hem çekim morfolojisini benzerlerinden daha geniş olarak ele almaktadır. (ii) İki-düzeyli yazım kuralları, ek dizilim kuralları, yaklaşık 24 bin kelimelik sözlük ve yaklaşık 150 adetlik ek sözlüğünden oluşan gerçeklememiz açık kaynak kodlu olarak paylaşıma açılmıştır. Üçüncü taraflarca indirilebilir, gözden geçirilebilir ve test edilebilir. (iii) Gerçeklememiz var olan kuralların değiştirilmesi veya yenilerinin eklenmesiyle kolayca genişletilebilir bir yapıdadır. Programlama gerektirmez. (iv) Nuve Çatısı çalışma grubumuz tarafından geliştirildiği için ortaya çıkan yeni problemleri kolay ve hızlı bir şekilde çözebilmekteyiz. (v) Gerçeklememiz ayrı yazılan ekler, iki sembolden meydana gelen harfler gibi durumları kolayca ele alabilmektedir. (vi) Nuve Türkçenin iki düzeyli morfolojisini de içermektedir. Bu sayede kelime hazinesi, kelime yapısı ve cümle yapısı yönlerinden büyük benzerlikler içeren Türki dillerle Türkçe arasında morfoloji tabanlı makina çeviri yapılabilir.

Anahtar kelimeler: Kazak Morfolojisi, Doğal Dil İşleme, Hesaplamalı Dilbilim

*This submission appeared as part of Züleyha YINER's doctoral thesis, titled "Kazakça Gramer ve Semantik Analiz ve Kazakça-Türkçe Otomatik Çeviri Sistemi" with advisor Doç. Dr. Atakan KURT

[1](Res. Asst.), Siirt University, Faculty of Engineering, Department of Computer Engineering, Siirt, Turkey

[2](Assoc. Prof.), Istanbul University-Cerrahpasa, Faculty of Engineering, Department of Computer Engineering, Istanbul, Turkey

ORCID: Z.Y. 0000-0001-7017-6114; A.K. 0000-0002-9549-8475

Corresponding author:
Züleyha YİNER
Siirt University, Faculty of Engineering, Department of Computer Engineering, Siirt, Turkey
E-mail address: zuleyhayiner@siirt.edu.tr

Submitted: 24.12.2020
Revision Requested: 31.03.2021
Last Revision Received: 13.04.2021
Accepted: 26.04.2021
Published Online: 29.06.2021

Citation: Yiner, Z., & Kurt, A. (2021). Two level Kazakh morphology. *Acta Infologica, 5*(1), 79-98. https://doi.org/10.26650/acin.842758

1. INTRODUCTION

In this paper, a Kazakh two-level morphological description is given in detail. This description is implemented using an open-source morphological analyzer and machine translation system, Nuve. In this description a root word lexicon consisting of more than twenty thousand is used.

A number of morphophonemic processes take place during derivation or inflection of a word in Kazakh similar to those in Turkish including vowel drops, consonant drops, vowel changes as in vowel harmony rules, consonant changes as in consonant harmony rules, etc. These rules express the conditions in which these modifications occur. 22 two-level orthographic rules are written to describe Kazakh's morphophonemic processes in the system. Morphotactics is a sequence of rules that orders suffixes in order to generate a logical and meaningful word. For nominal and verbal paradigms, morphotactics is encoded as FSA.

Kazakh, belongs to the Turkic languages family, is spoken mainly in Kazakhstan as official language, and the other neighboring republics of Kyrgyzstan, China, Tajikistan, Uzbekistan, Pakistan, Russia, and Turkey. Also, it is spoken by more than 16 million people. Kazakh is an agglutinative language like other Turkic languages producing affixations of derivational and inflectional morphemes to root words. A number of morphophonemic rules help us to modify the surface realization of morphological constructions. Vowels in the affixed morphemes have to agree with the preceding vowel in certain aspects to achieve vowel harmony. Under certain circumstances, vowels in the root or affixes are deleted. Similar to vowel harmony, consonants in the root words or in the affixed morphemes experience certain modifications, and sometimes may be deleted. For example, the word аталарымыздан (atalarımızdan), can be broken into morphemes as follows:

ата+лар+ымыз+дан

where + indicates morpheme boundaries. This word can be translated into English as "from our ancestors".

2. BACKGROUND and RELATED WORK

Morphotactics are the rules governing the order of morpheme affixations and are expressed as a finite state machine (FSA) using root words and suffixes. A two-level morphological model has been applied to a number of languages such as those by Oflazer (1994) for Turkish, by Alam (1983) for Japanese, by Antworth (1990) for English, by Kim et al. (1994) for Korean, Turkmen by (Tantuğ et al., 2006; Shylov,2010), by (Görmez et al., 2011; Yiner et al., 2016) for Kyrgyz, by (Orhun et al., 2009; Keskin,2012; Ablimit et al.,2016; Abdukerim,2019), Crimean Tatar by (Altıntaş,2001; Şanlı,2018), Qazan Tatar by (Gökgöz,2011) and so on.

Zafer et al. (2011) gives a brief two-level morphological description for Kazakh. In this description, the Kazakh phonological system is described with 27 two level rules which describe the transformation between the lexical level and the surface level of a word which is written in Latin (not original as in Cyrillic). And finite state machines to define nominal and verbal morphotactics of Kazakh. Both orthographic rules and the finite state morphotactics are implemented on a language independent framework which is Dilmaç (Shylov,2010). Makhambetov et al. (2014) uses a data-driven approach to do morphological analyzing and labels morphemes as transition labels testing on the Kazakh National Corpus (Makhambetov et al., 2013). Their morphological analyzer model has two steps which are segmenting the given word and ranking each candidate segmentation using HMM and Markov chain rules. They do not consider compound words and some phonological rules which are important issues in a language.

Kessikbayeva & Cicekli (2014) gives a rule-based morphological analyzer using Xerox tools for the Kazakh language. This study does not work widely due to the lack of tools for analyzing Kazakh. Also, in this study a Latin transcription of Cyril text is used. Another lack of this study is not being able to handle separate written conjugations that is compound verbs. They used 1000 words randomly selected from web for testing and get nearly 96% correct analyses. Then, in the extended version of this study (Kessikbayeva & Cicekli, 2016), they use Foma which is an open source environment to implement a rule based morphological analyzer. They get approximately 99% correct analyses on test corpora which is nearly 15000

words written in Cyrillic. They use over 57 main alternation rules, including exception rules for each case separately, to define language grammar on Foma. This study is very important in terms of scope compared to the previous studies of the Kazakh language.

Bekmanova et al. (2017) proposes a uniform morphological analyzer both Kazakh (in Latin) and Turkish. But they give just morphological features of Kazakh based on (Eryiğit & Adalı, 2004), no more information how their morphological analyzer works. In (Washington, Salimzyanov & Tyers, 2014), an open-source finite-state transducer is proposed to get a morphological analysis of three Turkic languages; Kazakh, Tatar, and Kumyk with a limited root/stem lexicon.

3. TWO LEVEL DESCRIPTION OF KAZAKH ORTHOGRAPHY

Two-level morphology of (Koskenniemi,1983; Karttunen,1983) is one of the practical models in computational linguistics for morphological analysis of languages. In this model, for all languages rules and lexicons are combined with a parser for analyzing any language. Publicly available tools like PC-KIMMO (Antworth,1990) can be used to implement a two-level morphology. In the two-level morphology approach, both orthographic rules are defined using two-level rules and derivational/ inflectional morphotactics are defined using FSAs. A word has two different representations or forms in this model: lexical and surface forms. The lexical form is a word structure or the representation of a word-formation, whereas the surface form is the written form of the word in the text generated by affixing suffixes by the morphology as given in the lexical form. The transformations from lexical to surface forms are defined with two-level orthographic rules as follows in (Oflazer, 1994).

The Kazakh alphabet has 23 consonants, 9 vowels (Biray, Ayan & Ercilasun, 2015). Table 1, Table 2, Table 3 shows Kazakh alphabet, consonants and vowels respectively.

Table 1

Kazakh Alphabet

Cyrillic	Latin	Cyrillic	Latin	Cyrillic	Latin	Cyrillic	Latin	Cyrillic	Latin	Cyrillic	Latin
а А	a	е Е	e	к Қ	q	п П	p	ф Ф	f	ъ Ъ	‘
ә Ә	ä	ё Ё	yo	л Л	l	р Р	r	х Х	x	ы Ы	ı
б Б	b	ж Ж	j	м М	m	с С	s	һ Һ	h	і І	i
в В	v	з З	z	н Н	n	т Т	t	ц Ц	c	ь Ь	
г Г	g	и И	ï	ң Ң	ň	у У	w	ч Ч	ç	э Э	é
ғ Ғ	ğ	й Й	y	о О	o	ұ Ұ	u	ш Ш	ş	ю Ю	yu
д Д	d	к К	k	ө Ө	ö	ү Ү	ü	щ Щ	şş	я Я	ya

Table 2

Kazakh Vowels

	Unrounded		Rounded	
Front	е,э(e)	і(i)	ө(ö)	ү(ü)
Back	а(a),ә(ä)	ы(ı)	о(o)	ұ(u)

Table 3

Kazakh Consonants

			Labial	Labi-al-Den-tal	Dental	Palato-Al-veolar	Fronto-Palatal	Velar	Glottal Stop
stops	plosive	voiced	б(b)		д(d)	ц(c)	г(g)		
		voiceless	п(p)		т(t)	ч(ç)	к(k)		
continuant	trill	voiced	м(m),у(w)		н(n)		л(l),р(r)	ң(ň)	
	fricative	voiced		в(v)	з(z)	ж(j)		ғ(ğ),й(y)	
		voiceless		ф(f)	с(s)	ш(ş)		х(x)	h(h)

In order to specify the two level morphology of Kazakh a subset of letters -called meta-letters – is used when expressing orthographic rules. These meta-letters are given below with their Cyrillic transliterations:

C = {b,ç,d,f,g,ğ,h,x,j,k,q,l,m,n,ň,p,r,s,ş,t,v,y,z}

= {б,ч,д,ф,г,ғ,h,х,ж,к,қ,л,м,н,ң,п,р,с,ш,т,в,й,з}

C_{ts}= {f,x,h,k,q,p,s,ş,t,ç}={ф,х,h,к,к,п,с,ш,т,ч}

C_{t} = {b,d,g,ğ,j,l,m,n,ň,r,v,y,z} = {б,д,г,ғ,ж,л,м,н,ң,р,в,й,з}

V = {a,e,ı,i,o,ö,u,ü,ä} = {a,e,ы,i,o,ө,ұ,ү,ə}

V_{b} = {a,ı,o,u} = {a,ы,o,ұ}

V_{f} = {e,i,ö,ü} = {e,и,ө,ү}

I = {ı,i} = {ы,i}

A = {a,e} = {a,e}

U = {u,ü} = {ұ,ү}

K = {q,k} = {қ,к}

N = {n,d,t} = {н,д,т}

M = {m,b,p} = {м,б,п}

G = {g,k,q,ğ} = {г,к,қ,ғ}

D = {d,t} = {д,т}

L = {l,d,t} = {л,д,т}

Above C, V, V_{b}, V_{f} represent the set of consonants, vowels, back and front vowels. We use the following notation in the lexical level of the two-level orthographic rules such as * for zero or more occurrence of the preceding letter, | for or (used to represent alternatives), ? for zero or one occurrence of the preceding letter, 0 for nothing on the surface (represents the dropped symbols), @ for the end of word (no more morpheme) and + for representing morpheme boundary (the beginning or end of a morpheme), not to be confused with regular expression + sign.

In the examples below, the first column is written in Cyrillic script and the second column is written in Latin script. In the third column (rightmost), we show the formulation in lexical form and the meanings of the root/words. The first line contains the lexical form of a word, and the second line the intermediate form. We don't show the surface form since it is almost identical to the intermediate form (the 0 place holders in the intermediate level are automatically deleted on the surface form).

1. These three rules express the vowel harmony in Kazakh. Vowels in the affixed suffixes to a word are resolved with respect to the previous vowel in the word. Back vowels are followed by vowels, and front vowels are followed by front vowels in a word according to the vowel harmony in Kazakh in a similar fashion to Turkish vowel harmony. An A on the lexical level in a suffix appended to a root is realized as a or e on the surface if the previous vowel in the root is (or is realized as) a back or front vowel respectively. In other words, the lexical A occurring after a back vowel in a word turns into the back vowel a on the surface:

A:a => V:V_b C* + C*__

A:e => V:V_f C* + C*__

ата+LAp	ata+LAr	N(ancestor) + NNI_PLU
ата0лар	ata0lar	Ancestors
жер+DA	jer+DA	N(ground) + NNI_LOC
жер0де	jer0de	on the ground

In some suffixes like Continuous tense suffix +A, verbal adjective suffix + АтІн or verbal adverb suffix +A, this rule does not provide the right changes for meta-letter +A. Thus, for these special cases, this rule works as below.

A:a => V:V_b C + C*__

A:e => V:V_f C + C*__

A:й => V + __

тұт+A+MIн	tut+A+MIn	V(hold)+CONT+1PS
тұт0а0мын	tut0a0mın	(I will hold)
оқы+АтІн	oqı+AtIn	V(read)+JVD_ATIN
оқы0йтын	oqı0ytın	reading (smt)
кел+АтІн	kel+AtIn	V(come)+JVD_ATIN
кел0етін	kel0etin	coming

2. The lexical I in a suffix appended to a root is realized as ı or i on the surface if the previous vowel in the root is (or is realized as) a back or front vowel respectively:

I:ы => V:V_b C* + C*__

I:i => V:V_f C* + C*__

бала+NIң	bala+NIn	N(kid) + NNI_OBJ
бала0ның	bala0nın	kid’s
көз+LIK	köz+LIK	N(eye)+ NND_LIK
көз0дік	köz0dik	Eyeglasses

3. This rule describes how the lexical U is resolved on the surface according to the Kazakh harmony rule. When a root is affixed with a suffix containing lexical U, lexical U is realized as u on the surface if the last vowel in the previous morpheme is a back vowel (V_b), otherwise, it is realized as ü on the surface.

U:ұ => V:V_b C* + C*__

U:ү => V:V_f C* + C*__

жый+Uв	jıy+Uv	V(collect)+ JVD_GAN
жый0ұв	jıy0uv	Collecting
жет+Uв	jet+Uv	V(reach)+ JVD_GAN
жет0үв	jet0üv	reaching

4. When a suffix starting with lexical L is affixed to a root, L on the lexical level is realized as l on the surface if the last letter of the root is a vowel. Lexical L is realized as d or t if the last letter of the root is one of C_t or C_{ts} respectively:

L:л <=> V + __

L:д <=> C_t + __

L:т <=> C_{ts} + __

ақыл+LI	aqıl+LI	N(intelligence) + JND_LI
ақыл0ды	aqıl0dı	Intelligent
пайда+LI	payda+LI	N(benefit) + JND_LI
пайда0лы	payda0lı	Beneficial
таш+LIк	taş+LIk	N(stone) + NND_LIK
таш0тык	taş0tık	stone-filled

5. When a suffix starting with lexical N is affixed to a root, N on the lexical level is realized as n on the surface if the last letter of the root is a vowel. Lexical N is realized as d or t if the last letter of the root is one of C_t or C_{ts} respectively:

N:н <=> V + __

N:д <=> C_t + __

N:т <=> C_{ts} + __

бала+NI	bala+NI	N(kid) + NNI_OBJ
бала0ны	bala0nı	the kid
көз+NI	köz+NI	N(eye) + NNI_OBJ
көз0дү	köz+dü	(was the) eye
иш+NIн	iş+NIn	N(work) + NNI_GEN
иш0тин	iş0tin	of work

6. This rule which is quite similar to the previous rule deals with d and t transformations on the surface. The D on the lexical level maps to d on the surface, if a suffix starting with lexical D is affixed to a root ending with a vowel or a C_t consonant, otherwise D is resolved as a t on the surface:

D:д <=> [V|C_t] + __

D:т <=> C_{ts} +__

иш+DA	iş+DA	N(work) + NNI_LOC
иш0те	iş0te	at work
жыл+DA	jıl+DA	N(year) + NNI_LOC
жыл0да	jıl0da	in a year

7. This rule which is quite similar to the previous rule deals with b and p transformations on the surface. The B on the lexical level maps to b on the surface, if a suffix starting with lexical B is affixed to a root ending with a vowel or a C_t consonant, otherwise B is resolved as a p on the surface:

B:б <=> [V|C_t] + __

B:п <=> C_{ts} + __

көр+GAн+BIз	kör+GAn+BIz	V(see) + PAST+1PP
көр0ген0біз	kör0gen0biz	I saw

8. This rule deals with the realization of lexical M on the surface. The lexical M corresponds to the surface p when a suffix starting with lexical M is affixed to a root ending with a voiceless consonant C_{ts}. M is realized as b, if the root ends with z or c. Otherwise, M is resolved as m on the surface:

M:м <=> [V|C_t-{z,c}] + __

M:б <=> {z,c} + __

M:п <=> C_{ts} + __

көр+MAстAн	kör+MASTAN	V(see)+ AVD_MASTAN
көр0местен	kör0mesten	without seeing
жүз+MAй	jüz+MAy	V(swim)+ AVD_MAY
жүз0бей	jüz0bey	without swimming
кес+MAйIншA	kes+MAyInşA	V(cut)+ AVD_MAYINSA
кес0пейінше	kes0peyinşe	without cutting

This orthographic rule of meta-letter M does not give the correct result on Question suffix +MA. For this suffix, orthographic rule works as below:

M:м <=> [V|C_t-{м,н,ң,з}] + __

M:б <=> { м,н,ң,з} + __

M:п <=> C_{ts} + __

бала+MA	bala+MA	N(kid)+ QUES
бала0ма	bala0ma	is it kid?
ертең+MA	erteñ+MA	Adv(tomorrow)+QUES
ертең0бе	erteñ0be	is it tomorrow?
жоқ+MA	jok+MA	Adj(absent)+QUES
жоқ0па	jok0pa	is it absent?

In some suffixes that comes after verbal stem like noun derivation from verb suffix +MA, lexical M is realized as m, b or p again but in different circumstances:

M:м <=> { л,м,р,в,й,н,ң } + __

M:б <=> [V|C_t-{л,м,р,в,й,н,ң}] + __

M:п <=> C_{ts} + __

жаз+МАстАн жаз0бастан	jaz+MAstAn jaz0bastan	V(write)+AVD_MASTAN without writing
қара+МАс қара0мас	qara+MAs qara0mas	V(look)+AVD_MAS does not look
бат+МАК бат0пақ	bat+MAK bat0paq	V(dive)+NVD_MAK swamp

9. This rule defines the transformation of lexical G at the beginning of some suffixes to ğ, g, k, q letters on the surface. G at the beginning of a suffix on the lexical level is realized as k and q on the surface when the last vowel of the affixed root is a front vowel V_f or back vowel V_b respectively and the root ends with a voiceless consonant C_{ts}. When the affixed root ends with back V_b or front V_f vowel and an optional voiced consonant C_t, then lexical G is resolved as ğ or g respectively:

G:ғ <=> [V:V_b|V:$V_b$$C_t$*] + __

G:г <=> [V:V_f|V:$V_f$$C_t$*] + __

G:к <=> V:$V_f$$C_{ts}$* + __

G:қ <=> V:$V_b$$C_{ts}$* + __

ал+GАн ал0ған	al+GAn al0ğan	V(take)+ JVD_GAN Taker
көр+GАн көр0ген	kör+GAN kör0gen	V(see)+ JVD_GAN Sighted
өт+GАн+МІн өт0кен0мін	öt+GAn+MIn öt0ken0min	V(pass)+ VVI_TPASTGAN + VVI_PERS1s1 I passed
шап+GАлІ шап0қалы	şap+GAlI şap0qalı	V(run)+ AVD_GALI running

10. When a root ending with the k consonant is affixed a suffix starting with a vowel, k at the lexical level is realized as g on the surface.

к:г => __ + VC?

ек+Іл+GАн ег0іл0ген	ek+Il+GAn eg0il0gen	V(sow) + VVI_PASSIL+ JVD_GAN sown

11. The p sound at the end of a root is realized as b on the surface when a suffix starting with a vowel is affixed to it.

п:б => __ + VC?

қалып+сІ қалыб00ы	qalıp+sI qalıb00ı	N(cast) + NNI_POSS3s cast of

12. When a root ending with the q consonant is affixed a suffix starting with a vowel, q at the lexical level is realized as ğ on the surface.

к:ғ => __ + VC?

соқ+Ip	soq+ Ip	V(hit) + AVD_IP
соғ0ып	soğ0ıp	by hitting

13. A syllable contains exactly one vowel in Kazakh and two consecutive vowels can't occur in a Kazakh word. This rule is based on this fact- states that when two consecutive vowels occur during affixation of suffixes, the one at the beginning of suffix is dropped. More precisely, when a root ending with a vowel takes a suffix starting with a vowel, then that vowel in the suffix is deleted on the surface.

V:0 => V + __

арба+Im	арба+Im	N(car) + NNI_POSS1s
арба00м	arba00m	my car

14. As an exception to the previous rule, in certain stems especially in the names of organs in the head such as (ауыз /mouth), (мойын /neck), (мұрын/nose), the last vowel in the stem is dropped instead of the vowel in the suffix. So when such a root takes a suffix starting with a vowel, then the last vowel in the stem is deleted on the surface.

V:0 => __C* + V

мойын+Im	moyın+Im	N(neck) + NNI_POSS1s
мойн0ым	moy0n0ım	My neck

15. When a suffix starting with a c (s) is affixed to a root ending with a consonant, the c at the lexical level in the beginning of the suffix drops on the surface.

c:0 <=> C* + __I

үй+cI	üy+sI	N(home) + NNI_POSS3s
үй00ү	üy00ü	it's home

16. When a stem takes a suffix containing K meta-letter, K meta-letter seems as k or q in surface form according to the last vowels of a stem.

K:к => $V:V_bC^* + C^*$__

K:к => $V:V_fC^* + C^*$__

жыл+LIK	jıl+LIK	N(year) + NND_LIK
жыл0дық	jıl0dık	Annual

17. This rule is written for ablative case suffix +DAн when coming after 1PS, 2PS, 3PS, and 3PP possessive suffixes. Under this circumstance, lexical D is realized as н on surface form.

D:н <=> Iм|Iң|cI|LArI + __Ан

көз+cI+DAн	köz+sI+DAn	N(eye)+3PS-POSS+ABL
көз00i0нен	köz00i0nen	From his/her eye

18. Aorist tense suffix +Ap is realized as +c when it comes after negative suffix.

Ap:c <=> мА + __

бер+мА+Ар	ber+MA+Ar	V(give)+AORST+3PS
бер0ме00с	ber0me00c	It does not give

19. One type of Optative Moods in Kazakh is done with +GI suffix followed by a person suffix and auxiliary word _келеді. If this auxiliary word _келеді followed by past tense suffix _еді, _келеді will be realized as _келетін in surface form.

ал+GI+Ім+_келеді+_еді	al+GI+Im+_keledi+_edi	V(take)+OPT+1PS+AUX+NARR
ал0ғы00м00келетін00еді	al0ğı00m00keletin00edi	I felt like taking

20. The lexical G in dative case suffix +GA is deleted on the surface if it comes after a personal suffix other than 3PS and 3PP. After 3PS and 3PP possessive suffixes, the lexical G in +GA is realized as н on the surface form.

G:0 <=> Ім|Ің|LApIң|LApIңIз + __A

G:н <=> cI|LApI + __A

жол+LAp+Ің+GA	jol+Lar+Iñ+GA	N(path)+PLU+2PS+DAT
жол0дар0ың00а	jol0dar0ıñ00a	To your paths (singular)
бала+cI+GA	bala+sı+GA	N(child)+3PS-POSS+DAT
бала0сы0на	bala0sı0na	To his/her child

This rule works differently while the dative case suffix +GA appended to 1st, 2nd and 3rd singular person pronouns, +GA is seen as +GAн (+GAn). This rule works as below:

мен+GA	men+GA	P(i)+DAT
ма00ған	ma00ğan	To me

21. Person pronouns can take case suffixes like nouns. This rule is written for transformation of 1 and 2 singular person pronouns мен (men), сен (sen) when take case suffixes such locative, genitive, ablative and the others. If 1 and 2 singular person pronouns take the Dative case suffix +GA, they are realized as mağan and sağan in surface form respectively (this transformation includes an orthographic rule for +GA suffix.). If they take the instrumental case suffix +мен (+men), i vowel epenthesis will occur between pronouns and suffix and they will be realized as менімен (menimen), сенімен (senimen).

сен+GA	sen+GA	P(you)+DAT
са00ған	sa00ğan	To you
сен+мен	sen+men	P(you)+INS
сен0імен	sen0imen	with you

22. The last sound of ol 3rd singular person pronoun drops whenever it takes one of genitive, accusative, dative, locative, ablative case suffixes. If this person pronoun takes instrumental case suffix, л at the end of pronoun is realized as ны in surface form.

ол+NI	ol+NI	P(it)+ACC
о00ны	o00nı	it (object pronoun)
ол+мен	ol+men	P(it)+INS
оны0мен	onı0men	with it

4. KAZAKH MORPHOTACTICS

Morphotactics in computational linguistics is a term for describing how the words in a natural language can be generated or parsed as a sequence of morphemes by affixing suffixes to roots in certain orders as defined by the morphology of the

language itself. The ordering of morphemes a nominal or verbal root takes in Kazakh is well defined and is strictly obeyed during word-formation.

The complete description of morphotactics of a language is usually given as a finite state machine (FSA). The states and the directed edges in an FSA represent the (subtypes of) Part-Of-Speech (POS or word class) of words and the affixed morphemes respectively. The two initial states are usually labelled as nominal root and verbal root since these are basic word classes. Nominal root includes nouns, adjectives, adverbs, etc. in our model. Final states represent the classes of words that take no more morphemes. The other states are intermediate states where generation/parsing can either continue or stop. Parsing or generation stops at the final states. The directed edges in an FSA are labelled with the lexical forms of morphemes (or sometimes with 0 meaning a transfer without a morpheme) a word takes while going from the source to the target state.

Such an FSA is used in creating morphological parsers or generators. Parsers usually have four main components:

1. Orthographic rules for modelling the phonological processes during affixation
2. An FSA for representing the ordering of morphemes during word formations
3. A root lexicon of nouns, verbs, adjectives, adverbs, etc.
4. A suffix lexicon for nominal/verbal derivations and conjugations

We used the Nuve Framework to implement the Kazakh morphology. Our implementation consists of the above components each specified in a separate file in the system. Thus we are able to parse (i.e. get the lexical form of a word from the surface form) and to generate (i.e. get the surface form of a word from its lexical form). In this study our aim was to generate all the words to avoid under generation at the expense over generation (generating invalid words).

4.1. Nominal Morphotactics

In this section, we present nominal morphotactics in the form of an FSA. Nominal morphotactics consist of nominal derivations and nominal conjugations. Nominal conjugation can be given as a simple formula called basic nominal model:

Nominal + Plural + Possessive + Case + Relative

The basic nominal model above represents the ordering of morphemes for nominal conjugation in the general case which is approximate and used for discussing morphology only. Comprehensive and precise nominal models are specified using FSAs given in Fig. 1. The parsing of an ordinary nominal word starts at Nominal root state and follows through Plural state with +LAp suffix, Possessive state with one of +Ім, +Ің, +сІ, +LApІң, +LApІңІз, +LApI suffixes, Case state with one of +NIң, +NI, +GA, +DA, +DAн, +Мен, +Менен suffixes, Relative state with +GI suffix. Since it is possible to treat a nominal as nominal verb (as in the case of a simple sentence with the nominal as the subject and “to be” as the verb), some nominal words will continue through the nominal verb (past and past perfect tense, conditional and imperfect moods), negative and question states shown in the lower part of the nominal FSA. The lexical form of кітаптарымдағы (one which is in my books) is given below as an example basic nominal model:

Кітап	+LAp	+Ім	+DA	+GI
Кітап	0тар	0ым	0да	0ғы
Book	Plural	Poss.1SP	Case	Relative

Nominal derivation which creates new stems is provided through following suffixes in Kazakh shown in Fig. 1: noun to noun (+Ав, +DAс, +LIK, +LIр, +AK, +сІз, +шIK, +шIл, +шAK, +шAң, +шA, +шI, +тай, +LI, +ғылт, noun to adjective (+LI, +ғылт, +қылт, +DAGAн, +ншI, +Nан, adjective to noun (+шIK), noun to verb (+сІн, +сI, +рA, +GAр,+LAт,+LAс, +LAн, +LA, +IK, +I, +DA, +Ай, +Ар, +Ал, +A), verb to noun (+MA,+Ім, +Іс, +GI, +IK, +AK, +ІндІ, +Gіш, +Іш, +Ін, +MAK, +GIн,+мАл, +шAK)verb to adverb (+Іп, +Май, +GAндA, +GAншA, +GAлI, +MAстAн, +MAйІншA, +A), adjective to adverb (+DAGAн).

4.2. Verbal Morphotactics

In this section, we describe verbal morphotactics in the form of an FSA. Verbal morphotactics consists of verbal derivations and verbal conjugations similar to nominal expressed as simple formulas called basic verbal model:

Verb + Negative_MA + Tense + Tense | Mood + Person

Verb + Tense + Negative_jok/emes + Person

There are two ways of creating negatives in verbal model one with Negative_MA another with Negative_jok/emes. көрмегенбіз (körmegenbiz, we did not see or we had not seen) and алмақ емеспін (almaq emespin, I will not take) are given as verbal model examples:

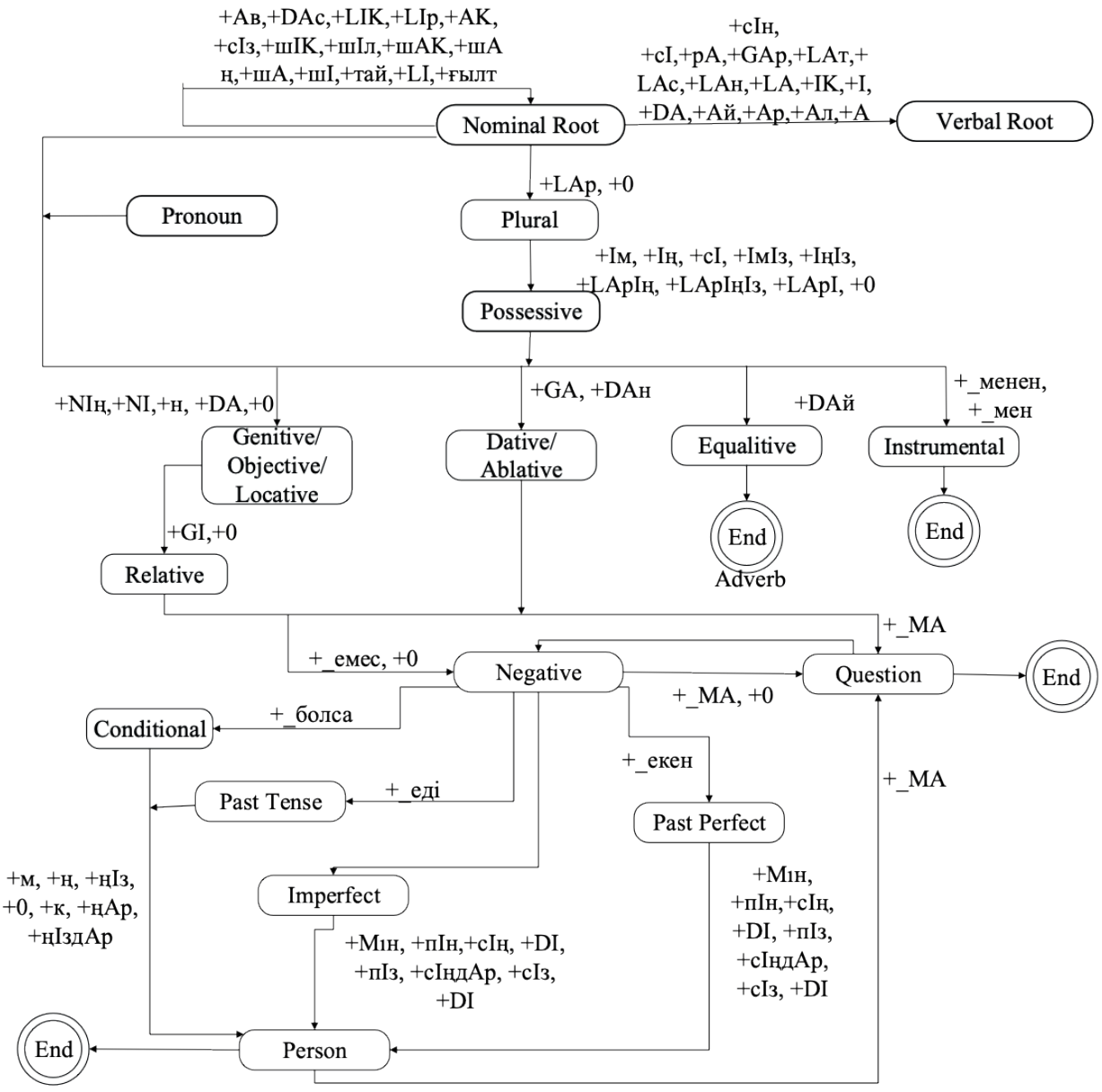

Figure 1. Nominal morphotactics

көр	+mA	+GAн	+BIз
көр	0ме	0ген	0біз
verb	Negative	Past Tense	1.Prsn Plural
ал	+MAK	+_емес	+мІн
ал	0мақ	00емес	0пін
Verb	Future Tense	Negative	1.Prsn Sing

Verbal paradigm is described by using FSA in Fig. 2 and Fig. 3. It is given as two parts for easy reading. Voice suffixes always come before tense suffixes. But it is not obligatory for the verbal root to take voice suffixes, it can take tense suffixes directly. Person suffixes come after simple tense suffixes. If a verbal stem takes a narrative tense suffix, person suffixes will come after narrative suffixes. There is not a specific order for negative verbal inflection. In some tenses, a negative suffix comes before tense suffixes while others are taken as the last suffix before person suffixes. '+0' means transition without a suffix (empty transition).

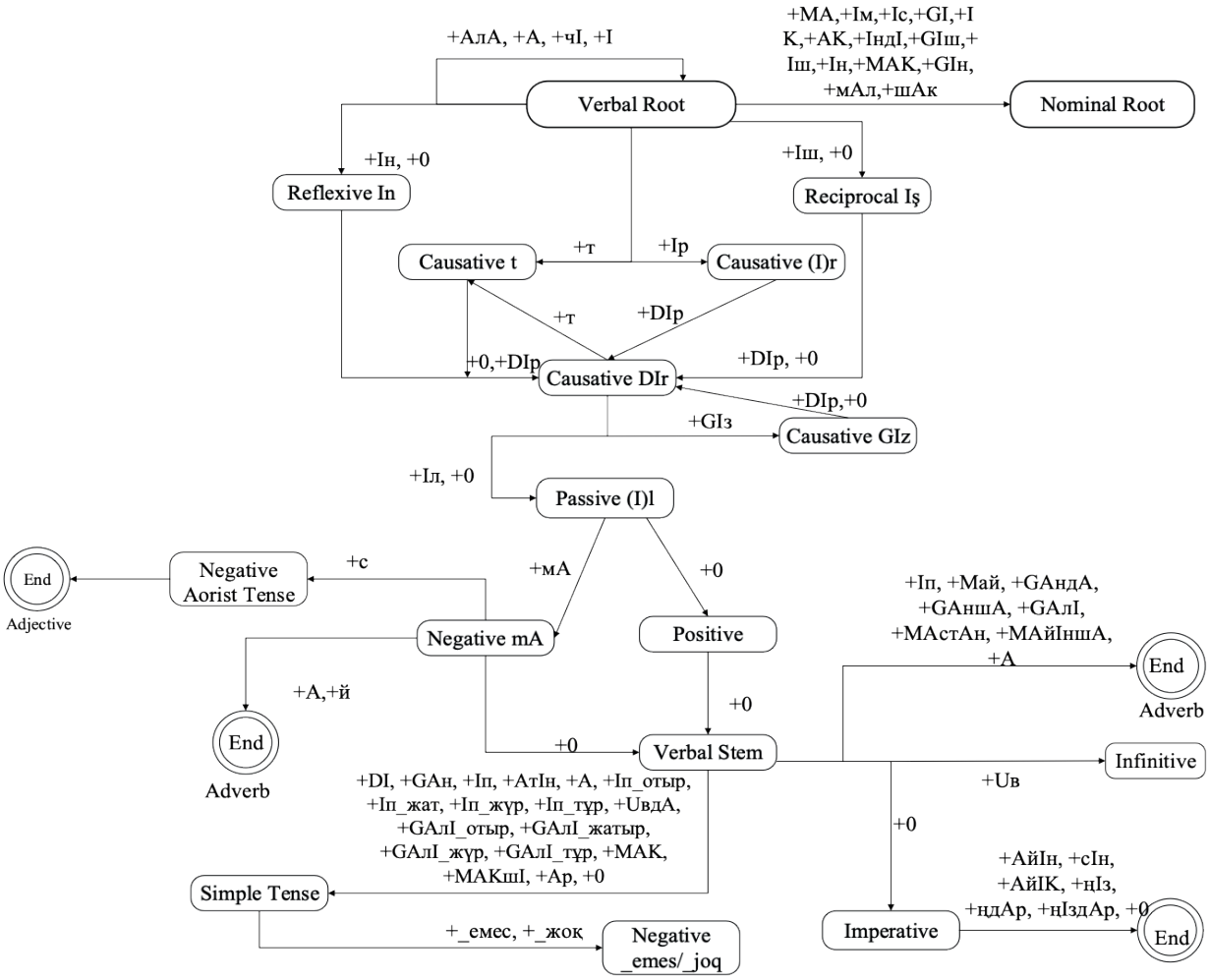

Figure 2. Verbal morphotactics first part

5. IMPLEMENTATION

An extensive description of Kazakh morphology is given above. We used Nuve for the implementation. Nuve [24] is an open-source project for morphological parsing and machine translation. Nuve was primarily developed for Turkic languages and includes an implementation of two level Turkish morphology.

The implementation in Nuve requires four distinct components each provided as a formatted text or XML file: root lexicon (TXT), suffix lexicon (TXT), orthographic rules (XML), morphotactics (XML).

Root Lexicon: There are approximately 24000 entries in the root lexicon. A few entries for the root lexicon are given in Table 5 below. Each root entry is specified using surfaces (different surface forms of the root), lex, active, id (POS: Noun, Adjectives, Verbs, Pronouns, Adverbs, Exclamations, Conjunctions, etc.), flags, rules (specific orthographic rules to apply for this root entry) attributes in this text file in the CSV format. Lex and flags attributes are used for special cases.

Table 4

Root lexicon

root	surfaces	Id	flags	rules
абажур		NOUN	noun	
абайла		VERB	verb	
абақ	абағ	NOUN	noun	MUTATION_qg
бар		VERB	verb, cnt	
кел		VERB	verb, cnt	

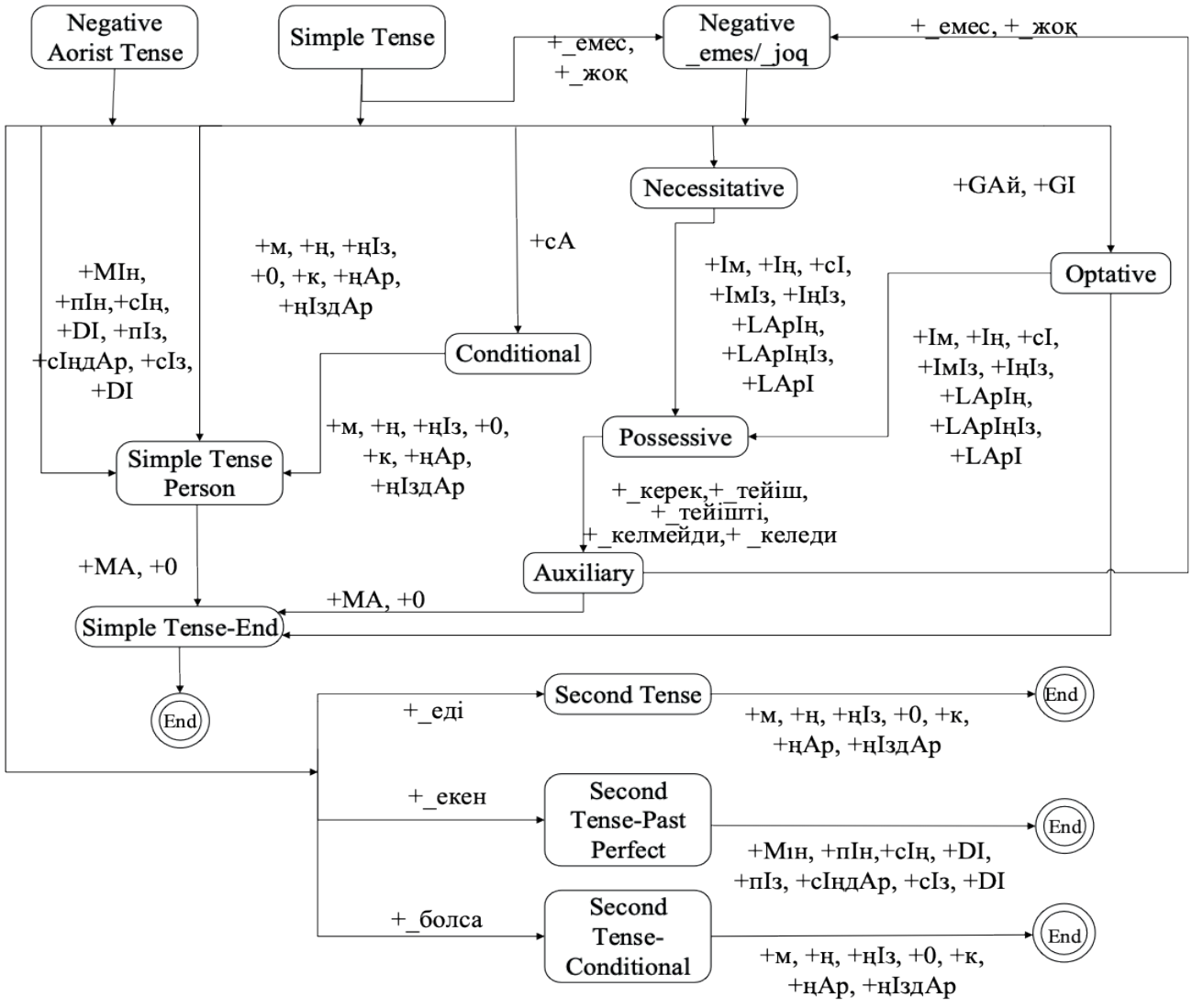

Figure 3. Verbal morphotactics second part

Suffix lexicon: Derivational and inflectional suffixes in the suffix lexicon are defined using id (a descriptive label), lexical (lexical form of the entry), type (suffix type: I: inflectional, D: derivational), suffix group (suffix category), rules (orthographic rules specifically applied to this entry), surfaces (distinct surface forms of the entry) attributes in this text file in the CSV format.

Table 5

Suffix Lexicon

id	lexical	grup	rules	surfaces
ni_plural_LAr	LAp		TRANSFORMATION_A, TRANSFORMATION_L	лар,лер,дар,дер,тер,тар
ni_possesive_1sg_(I)m	Ім	noun-possesive	TRANSFORMATION_I, VOWEL_DROP	м,ым,ім
vi_tense_fut_GAlI_otır	GAлI_отыр	verb-tense	TRANSFORMATION_I, REMOVE_UNDERSCORE, TRANSFORMATION_G, TRANSFORMATION_A	ғалы_отыр, гелі_отыр, қалы_отыр, келі_отыр

Morphotactics file: This file contains suffix groups and the FSA in XML format as shown in Fig. 4. Suffixes are grouped into suffix groups for ease of use. The nodes and edges of the FSA are specified using source and targetGroup elements in the file.

```
<morphology lang=”Kz-KZ”>
        <suffixGroup name=”NOUN_POSSESSIVE”>
                <suffix>NI_POSSESSIVE_1sg_(I)m</suffix>
                <suffix>NI_POSSESSIVE_2sg_(I)ň</suffix>
                ...
        </suffixGroup>
       ...
 <graph>
        <source id=”NOUN”>
                <targetGroup id=”NOUN_POSSESSIVE” />
                <targetGroup id=”NOUN_CASE” />
                <targetGroup id=”ND_NOUN” />
                <targetGroup id=”ND_VERB” />
                ...
       </source>
       ...
 </graph>
</morphology>
```

Figure 4. Morphotactics XML file

Orthography file: Two-level orthographic rules are encoded in XML in this file shown in Fig. 5. The conditions and transformations of the rules are defined using the transformation and conditions elements respectively. Complex logical conditions can be formed using and, or or operands. Symbols or morphemes in the words can be accessed using previous and next attribute values. Symbols can be deleted, inserted or replaced with action attribute of the transformation element.

```
<rule id=”TRANSFORMATION_A” phase=”2”>
    <description>Rule for meta-letter A changing into (a or e) </description>
    <transformation morpheme=”This” action=”Replace” operandOne=”A” operandTwo=”a” flag=”all”>
      <conditions flag=”Or”>
          <condition morpheme=”Previous” operator=”LastVowelEquals” operand=”аәёоұыюя” />
      </conditions>
    </transformation>
    <transformation morpheme=”This” action=”Replace” operandOne=”A” operandTwo=”e” flag=”all”/>
</rule>
```

Figure 5. Orthography XML file

6. DATA SET AND TEST

We tested two level description of Kazakh morphology on data set from Universal Dependency1. The text of UD Kazakh treebank (Makazhanov et al., 2015; Tyers et al., 2015) is collected from various sources such as Wikipedia, some folk tales, sentences from the UDHR, news and phrasebook sentences. The tokenization and morphological processing in the Kazakh UD treebank follow the principles of Turkic lexica in Apertium. Originally, the treebank is randomly split into training (80%), testing (10%), and development (10%) sets. We gave data set to Nuve as input for morphological analysis. The data set consists of over 1K sentences and more than 10K tokens. Over 8K tokens have only one morphological analysis. More than 1500 tokens have two or more morphological analyses. While finding more than one morphological analysis for a word is considered an advantage in terms of performance of a parser, it results in the problem of choosing right morphological parsing among many (morphological disambiguation). Morphological disambiguation is important for many NLP applications which depend on the accurate parsing of words, such as syntactic parsing, word sense disambiguation, spelling correction and machine translation and so on. We plan to use this data set for an experimental study on the morphological disambiguation of Kazakh in the future.

7. SPECIAL CASES AND EXAMPLES

While defining the morphology of the language, some special cases are encountered. In this section, these special cases are presented.

1. There is a confusion about the noun to adjective +LI suffix and the accusative +NI suffix both applied to nouns. For example, құрметті адам (kurmet+LI) kurmetti adam, *respectful man,* and *к*өрсеткен құрметті (kurmet+NI) жакшы көрді(көр+DI) *körsetken kurmetti jakşı kördi, s/he liked the respect shown to her/him*. Our implementation is able to parse both words.

2. Sometimes the negative +MA suffix and the verb to noun derivation +MA suffix may be confused since both are affixed to verbs and have the same surface. For example, **аспа** ас+MA, aspa, *hanging (nominal) or do not hang (verb).* However, this ambiguity can be resolved at the morphological disambiguation phase.

3. After possessive suffixes (except 1st and 2nd plural person), ablative case suffix +DAн (+DAn) becomes +нАн (+nAn). For example:

ата+(I)м+DAн, атамнан, *atamnan, from my father*

үй+(I)іміз+DAн, үйіміздeн, *üyimizden, from our home*

4. +_şe is used in the question form of Conditional Mood as a special case, in addition to the usual +MA question suffix. For example, жазсам ше or жазсам ба? *jazsam şe/be?, should I write*, келсем ше? or келсем бе? k*elsem şe/be?, should I come?.*

5. In Kazakh some morphemes are written separately such as the question suffix +MA, or the continuous tense suffixes +Ip otır, +Ip tur, +Ip jür, +Ip jat. Our implementation is able to parse these separately written suffixes without difficulty as shown below whereas other implementations either parse these suffixes as separate words or can't parse at all.

келіп отырсың, кел(VERB)+Ip otır+2.PrsnSing

6. In the continuous tense, jatır auxiliary verb requires an extra +A suffix only if it is appended to kel- and bar- verbs. This problem is handled with the *cnt* flag attribute in the root lexicon in the implementation. кел+е жатырмын, *I am coming* and бар+а жатырсыздар, *I am going/arriving.*

7. In adjective clauses made with verbal adjective +GAn suffix, the possessive meaning is provided by a pronoun before the clause. For example, мен жазған хат, *men jazğan hat*, *the letter I wrote*, сен күткен арба, *sen kütken arba, the car you*

1 https://universaldependencies.org/treebanks/kk_ktb/index.html

waited for. The person suffix normally attached to the verb, in this case, is replaced with a pronoun preceding the verb. These types of adjective clauses are too complex to be properly parsed by any parser.

барма	бар+мА	V(бар) +NEG+2PS (Imperative) *do not go*
	бар+мА	V(бар) + VND(MA) *going*
үйлерің	үй+LAp+Ің	N(үй) + PLU+2PS-POSS *your homes*
	үй+LApІң	N(үй) + 2PP-POSS *your home (polite)*
жылы	жылы	Adj(жылы) *warm*
	жылы	Adv(жылы) *warmly*
	жыл+сІ	N(жыл) + 3PS-POSS *it's year*
жүз	жүз	Adj(жүз) hundred
	жүз	N(жүз) *face*
	жүз	N(жүз) *tribe*
басты	бас+LI	N(бас) +NND(LI) *having (a) head*
	бас+DI	V(бас)+ PAST+ 3PS (s*/he) stepped on*

8. CONCLUSION

We have provided a comprehensive two level description of Kazakh morphology with an implementation on Nuve Framework consisting of 22 orthographic rules for the definition of morphophonemic processes, the morphotactics for nominal and verbal derivations and inflections using a large root lexicon and a suffix lexicon in which special or exceptional cases are manually annotated for ease of use. Our study covers both derivational and inflectional morphology to a greater extend (ii) Our implementation consisting of orthographic rules, morphotactics, a root lexicon of roughly 24 thousand roots, a lexicon of roughly 150 suffixes is open source and free (https://nuvestudio.com/downloads/kz_KZ/kz_KZ.zip) which can be downloaded, reviewed and tested. (ii) Roughly 10 thousand manually disambiguated word parses are available as a morphological disambiguation data set. (iii) Our implementation is easily extensible meaning it can be modified or extended with new rules without any programming, because all resources are encoded as XML/CSV files and ready to use as soon as they are uploaded into the system. (iv) We are able tackle emerging problems quickly and easily, since Nuve is maintained by our study group. The original version of Nuve was developed in .NET. The current version is in PHP. (v) Our implementation can handle problems such as separately written morphemes or digraphs etc. directly. No indirect methods are necessary to handle these cases. (vi) We also have a Turkish morphological parser/generator in Nuve for morphology based machine translation between Turkish and other Turkic languages (which is the next part of our project) since these closely related languages have a lot in common from lexical, morphological, and syntactic aspects. (vii) Finally, we have Kazakh morphological parser and generator available online to anyone who wishes to use.

Peer-review: Externally peer-reviewed.
Conflict of Interest: The authors have no conflict of interest to declare.
Grant Support: The authors declared that this study has received no financial support.
Author Contributions: Conception/Design of Study- Z.Y., A.K.; Data Acquisition- Z.Y., A.K.; Data Analysis/Interpretation- Z.Y., A.K.; Drafting Manuscript- Z.Y., A.K.; Critical Revision of Manuscript- Z.Y., A.K.;Final Approval and Accountability- Z.Y., A.K.

Hakem Değerlendirmesi: Dış bağımsız.
Çıkar Çatışması: Yazarlar çıkar çatışması bildirmemiştir.
Finansal Destek: Yazarlar bu çalışma için finansal destek almadığını beyan etmiştir.
Yazar Katkıları: Çalışma Konsepti/Tasarım- Z.Y., A.K.; Veri Toplama- Z.Y., A.K.; Veri Analizi/Yorumlama- Z.Y., A.K.; Yazı Taslağı- Z.Y., A.K.; İçeriğin Eleştirel İncelemesi- Z.Y., A.K.; Son Onay ve Sorumluluk- Z.Y., A.K.

References/Kaynaklar

Abdukerim, G., Tursun, E., Yang, Y., & Li, X. (2019). Uyghur morphological analysis using joint conditional random fields: Based on small scaled corpus. *Discrete & Continuous Dynamical Systems-S*, 12(4&5), 823.

Ablimit, M., Kawahara, T., Pattar, A., & Hamdulla, A. (2016). Stem-affix based Uyghur morphological analyzer. *International Journal of Future Generation Communication and Networking*, 9(2), 59-72.

Alam, Y. S. (1983). A two-level morphological analysis of Japanese. *In Texas Linguistic Forum*, 22, 229-252.

Altintas, K., & Cicekli, I. (2001). A morphological analyser for Crimean *Tatar. Proceedings of the 10th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN'2001),* North Cyprus, 180-189.

Antworth, E. L. (1990). PC-KIMMO: A two-level processor for morphological analysis. *Summer Institute of Linguistics, International Academic Bookstore*, Dallas, Texas.

Bekmanova, G., Sharipbay, A., Altenbek, G., Adali, E., Zhetkenbay, L., Kamanur, U., & Zulkhazhav, A. (2017). A uniform morphological analyzer for the Kazakh and Turkish languages. *Proceedings of the Sixth International Conference on Analysis of Images, Social Networks and Texts*, Moscow, Russia, 20–30.

Biray N., Ayan E., Ercilasun G. K., (2015). *Çağdaş Kazak Türkçesi Ses-Şekil- Cümle Bilgisi- Metinler* (2nd ed.). Istanbul, Turkey: Bilge Kültür Sanat.

Eryiğit, G., & Adalı, E. (2004, February). An affix stripping morphological analyzer for Turkish. *Proceedings of the IASTED International Conference Artificial Intelligence and Applications*, Innsbruck, Austria, 299–304.

Gökgöz, E., Kurt, A., Kulamshaev, K., & Kara, M. (2011, May). Two-level Qazan Tatar morphology. *Proceedings of the 1st International Conference on Foreign Language Teaching and Applied Linguistics (FLTAL'11)*, Sarajevo, Bosnia and Herzegovina, 428-432.

Görmez, Z., Ünlü B. S., Kurt, A., Kulamshaev, K., & Kara, M. (2011). An overview of two-level finite state Kyrgyz morphology. *Proceedings of the 2. International Symposium on Computing in Science & Engineering (ISCSE).,* Aydin, Turkey, 48-52

Karttunen, L. (1983, December). KIMMO: a general morphological processor. *In Texas Linguistic Forum*, 22, 163-186.

Keskin, R. (2012). *Two Level Uyghur Morphology and Uyghur Turkish Machine Translation.* (Master's Thesis). Fatih University the Graduate Institute of Sciences and Engineering, Istanbul.

Kessikbayeva, G., & Cicekli, I. (2014, June). Rule-based morphological analyzer of Kazakh language. *In Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM,* Baltimore, Maryland, 46-54.

Kessikbayeva, G., & Cicekli, I. (2016). A rule based morphological analyzer and a morphological disambiguator for Kazakh language. *Linguistics and Literature Studies*, 4(1), 96-104

Kim, D. B., Lee, S. J., Choi, K. S., & Kim, G. C. (1994, August). A two-level morphological analysis of Korean. *In Proceedings of the 15th Conference on Computational Linguistics, Vol 1,* 535-539.

Koskenniemi, K. (1983, August). Two-level model for morphological analysis. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence,* Karlsruhe, Germany, 683-685.

Makazhanov, A., Sultangazina, A., Makhambetov, O., & Yessenbayev, Z. (2015). Syntactic annotation of Kazakh: following the universal dependencies guidelines. A report. *In Proceedings of the International Conference Turkic Languages Processing- TurkLang-2015*, Kazan, Tatarstan, 338-350.

Makhambetov, O., Makazhanov, A., Yessenbayev, Z., Sabyrgaliyev, I., & Sharafudinov, A. (2014). Towards a data-driven morphological analysis of Kazakh language. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi,* 7(2), 31-36.

Makhambetov, O., Makazhanov, A., Yessenbayev, Z., Matkarimov, B., Sabyrgaliyev, I., & Sharafudinov, A. (2013, October). Assembling the Kazakh language corpus. *In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing,* Seattle, Washington, USA, 1022-1031.

Oflazer, K. (1994). Two-level description of Turkish morphology. *Literary and linguistic computing, 9(2),* 137-148.

Orhun, M., Tantug, A. C., & Adali, E. (2009). Rule based analysis of the Uyghur nouns. *International Journal on Asian Language Processing, 19 (1),* 33-43.

Shylov, M. (2010). *Two level Turkmen morphology and a Turkmen Turkish machine translation*, (Master's Thesis). Fatih University the Graduate Institute of Sciences and Engineering, Istanbul.

Şanlı, T. (2018). *Kırım Tatarcası'nın biçimbilimsel çözümlemesi ve Kırım Tatarcası-Türkçe biçimbilimsel makina çevirisi Sistemi*. (Master's Thesis). Istanbul University Institute of Graduate Studies in Science and Engineering, Istanbul.

Tantuğ, A. C., Adalı, E., & Oflazer, K. (2006, August). Computer analysis of the Turkmen language morphology. *Proceedings of the* 5th *International Conference on Natural Language Processing,* Turku*,* Finland, 186-193.

Tyers, F. M., & Washington, J. (2015). Towards a free/open-source Universal Dependency Treebank for Kazakh. In Proceedings of the International Conference Turkic Languages Processing, TurkLang-2015, Kazan, Tatarstan, 276-289 .

Washington, J., Salimzyanov, I., & Tyers, F. M. (2014, May). Finite-state morphological transducers for three Kypchak languages. Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC, Reykjavik, Iceland ,3378-3385.

Yiner, Z., Kurt, A., Kulamshaev, K., & Zafer, H. R. (2016, May). Kyrgyz orthography and morphotactics with implementation in NUVE. *Proceedings of International Conference on Engineering and Natural Sciences*, Sarajevo, Bosnia and Herzegovina, 1650-1658.

Zafer, H. R., Tilki, B., Kurt, A., & Kara, M. (2011, May). Two-level description of Kazakh morphology. *Proceedings of the 1st International Conference on Foreign Language Teaching and Applied Linguistics (FLTAL'11),* Sarajevo, Bosnia and Herzegovina, 560-564.

Zafer, H. R., "Nuve: A natural language processing library for Turkish in C#". [Online]. Available: https://github.com/hrzafer/nuve. (05.12.2020).