УДК 004.912

Б.О. Кочконбаева - улук окутуучу,
Ош технологиялык университети

## КЫРГЫЗ ТИЛИ ҮЧҮН СӨЗДҮН НЕГИЗИН АНЫКТОО МОДЕЛИ

*Табигый тилдердеги сөздүн негизин аныктоо моделдери жана алгоритмдери каралып, тексттеги сөздү сөздүктөгү формасына айландыруу (лемматизация) жолдору бейнеленет. Кыргыз тили үчүн сөздүктөгү сөздөрдү жана анда камтылбаган бирдиктерди териштирүү эрежелери талкууланат.*

*Ачкыч сөздөр: кыргыз тили, морфология, тексттеги сөз, сөздүктөгү сөз, сөздүн негизи, автоматтык аныктоо алгоритми.*

Б.О. Кочконбаева - ст. преподаватель,
Ошский технологический университет

## МОДЕЛЬ ОПРЕДЕЛЕНИЯ ОСНОВЫ СЛОВА ДЛЯ КЫРГЫЗСКОГО ЯЗЫКА

*Рассматриваются модели и существующие алгоритмы лемматизации слов естественных языков. Описаны алгоритмы автоматического выделения основ для ряда естественных языков и возможные пути определиния словарной формы слова для кыргызского языка. Разработаны правила анализа слов для кыргызского языка и алгоритм для обработки как словарных, так и отсутствующих в словаре слов.*

*Ключевые слова: кыргызский язык, морфология, текстовая форма слова, словарная форма слова, основа слова, алгоритм автоматического выделения основы слова.*

B.O. Kochkonbaeva - senior lecturer,
Osh technological university

## MODEL OF DETERMINATION OF THE BASIS OF WORDS FOR THE KYRGYZ LANGUAGE

*Models and existing algorithms for lemmatizing words of natural languages are considered. The algorithms of automatic selection of the bases for a number of natural languages and possible ways of determining the dictionary form of a word for the Kyrgyz language are described. The rules for analyzing words for the Kyrgyz language and an algorithm for processing both dictionary and missing words are developed.*

*Key words: Kyrgyz language, morphology, text form of words, dictionary form of words, bases of words, automatic word base selection algorithm.*

**Киришүү.** Кыргыз тили типологиялык жактан формак тилдеринин тобундагы агглютинативдик тилдердин катарына кирет жана татаал морфологияга ээ. Сөздөр негизинен уңгуга мүчөлөрдүн уланышы менен ишке ашат. Сөз ичинде негизинен мүчөлөрдүн саны бир, эки-үч, кээде андан да көп болушу мүмкүн.

Бүгүнкү күндө маалымат издөө процессинде автоматтык морфологиялык анализ аркылуу сөздөрдү сөздүктөгү формасына келтирүү негизги багыттардын бири болуп турат. Табигый тилдеги текстте колдонулган сөздөрдү сөздүктөгү негизги формасына

келтирүү текстти автоматтык иштөөнү, маалымат издөөнү, машина котормосун ж.б. жумуштарды жеңилдетет.

Бул макалада кыргыз тилиндеги морфологиялык анализдөө аркылуу сөздөрдүн негизин табуу маселеси каралат. Анализдөө үчүн зат атооч, сын атооч жана этиш сөздөрдүн негизги формаларын анализдөөчү морфологиялык модель тандалды.

**Маселени чечүүнүн жолдору.** Кыргыз тили үчүн тексттеги сөздөрдү негизги формага келтирүүдө морфологиялык талдоонун усулдарын жана ыкмаларын карап чыгабыз. Морфологиялык талдоодо принципиалдуу эки башка жолду пайдаланышат: сөздүктөргө негизделген усулдар жана сөздүксүз морфологиялык анализаторлор.

Морфологиялык үлгүлөргө, сөздөрдү лемматизациялоо алгоритмдерин курууга багытталган жолдорго токтолобуз. Иштеп жаткан жолдор эки класска бөлүнөт: стеммингдин жана лемматизациянын алгоритмдери.

*Стемминг* – берилген баштапкы сөз үчүн сөздүн негизин табуу процесси. Сөздүн негизи дайыма эле уңгусу менен дал келе бербейт.

Тексттеги сөздү сөздүктөгү сөзгө айландыруу амалы *лематизация* деп аталат. Мында текстти татаал сөздөрдүн чегин ажыратуу маселеси келип чыгат. Тилекке каршы, учурдагы орфографиялык эрежелер бул маселени аттап өтүп, мындай сөздөрдү бөлөк жазууга артыкчылык берет [1].

Кээ бир түшүндүрүүчү аныктамаларды карап көрөлү.

Лемма - лемматизациялоо (сөздүк, канондук) сөздүн формасы. Мисалга алсак, кыргыз тилинде төмөнкүлөр лемма болуп саналат:

1) Зат атоочтор. Негизги формасы деп жекелик санда келген атооч жөндөмөдөгү сөз эсептелет: атасы→ата, балам→бала.
2) Сын атоочтор. Сын атоочтун баштапкы формасы болуп жай сын эсептелет: кызыл, жашыл ж.б.
3) Этиштер. Этиштин баштапкы формасы катары чак, жак, сан, таңгыч жана ыңгай категорияларыны мүчөлөрү жалганбаган негизги мамиледеги туунду жана тубаса формасы алынат: бар-, кел-, иште- ж.б.

Морфологиялык талдоо негизги эки маселенин чечип берет:

- талдоонун милдеттери – сөздүн тексттеги формасы боюнча сөздүн негизин аныктоо,
- синтездин милдеттери – негизги форма боюнча сөздү бардык формаларын түзүү.

Көпчүлүк популярдуу болгон алгоритмдер лемматизацияны (негизги формага келтирүү) сөздүн негизин (стеммингдин алгоритмин) колдонуу менен жүзөгө ашырышат. Бирок бул жерде эки көйгөй көмүскөдө калган, алар кыргыз тилине мүнөздүү: биринчиден, негизги формадагы синтез сөздүн негизин алуу ыкмасына көз каранды, экинчиден, көпчүлүк жүзөгө ашыруулар бардык мүмкүн болгон леммаларды синтездейт, алардын ичинен жеке натыйжаны гана тандабай, же сөздүн негизин аныктоого токтолот.

Ар кандай принциптерге таянган, лемматизациянын эки эң популярдуу алгоритмдерин талдап чыгабыз: Портердин алгоритми жана Яндекстин алгоритми.

Портердин стемминг алгоритми 1980-жылы англис тили үчүн Мартин Портер аркылуу жарык көргөн [2]. Анда кадамдардын иреттүүлүгү бейнеленген, анын ар биринде белгилүү эрежелерде мүчөнүн белгилүү өзгөрүүлөрүнүн бири боло алат.

Бул эреже төмөнкү структурага ээ [2]: < шарт>< мүчө>→< жаңы мүчө>.

Портердин алгоритминин негизги идеясы мында: форма жана сөз жасоочу мүчөлөрдүн чексиз саны бар, жана сөздүн негизи кандайдыр бир негиздердин базаларын (сөздүктөрүн) колдонбой туруп кайрадан пайда болот: көптөгөн орун алган мүчөлөр гана (бул учурда татаал мүчөлөр жөнөкөйлөргө бөлүнөт) жана колу менен берилген эрежелер.

Портердин алгоритми эч кандай сөздүктөрдү жана негиздердин базаларын колдонбогону, тез аракеттенүү жана спектрды колдонуу үчүн (анын жок сөздөр менен иштөөсү жаман эмес) эсептелет жана негизди тандоо тактыгы тарабынан жакшы деп

караганда бир эле учурда минус болуп саналат. Мындан тышкары, Портердин алгоритминин минустарына адамдык факторду да көп киргизишет; текшерүү үчүн эрежелер кол менен берилүүсү жана кээде тилдин грамматикалык өзгөчөлүктөрү менен байланышуусу, катанын ыктымалдуулугун көбөйтөт [3].

Яндекстин алгоритми (Mystem – бул Илья Сегаловичдин иштеп чыгуусу (Яндекс, 1998) [4]. Бул морфологиялык талдоонун алгоритми сөздүктүк болуп саналат. Алгоритмдин негизги өзгөчөлүгү төмөндөгүчө: сөздүктө сүрөттөлбөгөн, же жок сөздүн сөз формалары үчүн, анын сөз өзгөртүүгө ыктымал болгон үлгүсүн сөздүн бир же бир нече негизги формаүнүн варианттарын алгоритм жаратат, - андан кийин, сөздүктү жаны лексемалар менен толуктап, бул сөздүктө (же ушул сыяктуу эле башка сөздүктө) жаралган гипотетикалык макалаларды кийин колдонуу үчүн сактаса болот.

Алгоритм талдануучу текстти сөзмө-сөз иштеп чыгат. Ар бир сөз тилдеги номинативдик милдетке ээ болбогон, лексикалык өз алдынча ээ болбогон сөздөрдүн тизмесине караштуулугу текшерилет, кыргыз тилиндеги мисал: жана, ары, да, бери ж.б.у.с. Бул тизмеге предлогдор, бөлүкчөлөр, кээ бир междометиялар жана кыргыз тилинде көбүрөөк колдонулган тактоочтор кирет. Андан тышкары, узундугу үч символдон кем болбогон бардык сөздөр иштелип чыгарылбайт. Эгерде сөз лексикалык өз алдынча ээ болбогон сөз болсо, анда ал өзгөрүлбөгөн жыйынтык деген катарга жазылат, алгоритм кийинки кезектеги сөздү иштеп чыгууга өтөт.

Кийинки аракет – сөздү издөө, ал өз алдынча сөз формакүмү болуп саналат, башкача айтканда, ар бир сөз үчүн негизди тандоо алгоритми аракетке киргизилет. Мүчөлөрдүн дарагынын жардамы аркылуу сөздөн мүчө алынат жана негиздердин дарагындагы көздөлгөн негиз изделет. Эгерде негиз даракта болгон болсо төмөнкүлөрдү текшеребиз: берилген негиздин жана берилген мүчөдин айкалышы мүмкүнбү, алынган үлгү керектүү сөз формакүмүн канааттандырабы. Эгерде ооба деген жооп болсо, берилген үлгүгө дал келген лемма кайтарылат.

Бул алгоритмдин оң жактары: негизги форманын ар бир варианты үчүн бардык грамматикалык маалыматтар сунушталат, бул маалыматтарды кийин программа сунуштаган көптөгөн негизги формалардын ичинен бирөөсүн тандоо үчүн колдонсо болот.

Берилген алгоритмдин терс жактары төмөндөгүлөр: ал дайыма эле киргизилген сөз жок болсо коюлган милдетти аткара албайт. Ошондой эле ал сөздүн кичирейтилген формасын да текшере албайт. Мисал катары биз «коенек» деген сөздү карасак алгоритм бул милдетти аткара алган жок, ошондуктан бул сөздү колдонуучу сөздүккө кошууга туура келди. Башкача айтканда, алгоритм дайыма эле сөздүктө жок сөздөрдү иштеп чыгара албайт.

**Чечимди тандоо.** Жогоруда көрсөтүлгөндөй, морфологиялык талдоого принципиалдуу формадан айырмалаганга жол бар. Биринчи усул, биз карап чыккан, бул Портердин алгоритми, сөздүгү жок режимде иштейт. Экинчи усул – Яндекстен болгон алгоритм, сөздүк менен иштейт. Морфологиялык талдоонун көйгөйлөрү сөздүк колдонулган анализаторлор аркылуу толугураак чыгарылат, алар табигый тилдердеги тексттерде сөздөрдү колдонуу үчүн, аларсыз мүмкүн эмес, грамматикалык өзгөчөлүктөрүн аныктоого мүмкүнчүлүк берет. Бирок бул усулсөздүктүн жардамы аркылуу бир кемчилдикке ээ:

Эгерде талданузучу сөз формасы сөздүктө жок болсо, анда ал жөнүндө кандайдыр-бир морфологиялык маалыматты алууга мүмкүн эмес.

Ошондуктан, издөө индексин курууда, толук тексти менен издөө үчүн морфологиялык талдоонун көйгөйүн чечүү үчүн, тилдик сөздүктөрдү колдонгон, негиздерди белгилөө ыкмаларын иштеп чыгуу милдети коюлат.

Мындай формадагы ыкмалар толугу менен морфологиялык талдоонун көйгөйлөрүн чече албайт, сөз формакүмдөрүн жана сөздүн формаларынын грамматикалык сыпаттарын аныктоо кыйынга турат. Бирок, мындай алгоритмдер

тексттик массивдерди индекстөө үчүн, кыргыз тили үчүн табигый текст менен иштөө процедурасын түзүү үчүн болгон көйгөйлөрү натыйжалуу болуп чыгат.

Ар кайсы тилдерде каралып чыккан усулдардын талдоосун колдонуу мүмкүнчүлүктөрүн түшүнүү үчүн кыргыз тилинин лингвистикалык классификациясын карап чыгуу керек. Морфологиялык структурасы боюнча кыргыз тили агглютинативдүү тилге жатат. Агглютинативдик тилдер үчүн аффиксациянын сөз жасоочу жана сөз өзгөртүү системасы өнүккөн, мүчөлөрдүн грамматикалык бир маанилүүлүгү, алмашуулардын жоктугу жетиштүү.

Учурдагы чечимдердин талдоосунун негизинде жогоруда айтылган алгоритмдерди байланыштырган кыргыз тили үчүн негизги формадагы сөздөрдү алуу алгоритминин модели жана эрежелери иштелип чыккан.

**Кыргыз тилинин морфологиялык модели.** Кыргыз тилинде сөз да, сөз формалары да уңгуга мүчө же мүчөлөрдү конкатенациялоо жолу аркылуу жасалат. Мында ар бир мүчөгө көптөгөн семантикалык сыпаттар менен байланышкан жана мүчөлөрдү кошуу ирети тартиби менен аныкталат. Мисалы, зат атоочтор үчүн сөздүн негизине башында көптүк форманын мүчөсү кошулат, андан кийин таандык мүчө, андан ары жөндөмө мүчөсү келет жана соңунда гана жак мүчөсү (зат атоочтун жандууларына гана) кошулат [6].

Жаңы сөз формалары негизги формалардын морфологиялык жана семантикалык сыпаттарын эске алуу менен жасалат: сөздүн негизги формасы; андан кийин, солдон оңго жылып, тиги же бул мүчөнү кошуу үчүн сөздүн негизги формасынын акыркы тамгасынын (тыбыштын) категориясы (үндүү жана үнсүз үнсүздөр) аныкталат. Анын жалпы морфологиялык формасы: уңгу (корень) + мүчө (окончание) [5].

Кыргыз тилиндеги сөздөргө үндүүгө карай кууш үндүү менен келген мүчөлөрдүн улануу модели төмөнкүчө ишке ашат [7]:

Белгилүү болгондой, морфемалар тилдин эң кичине маани берүүчү (семантикалык) бирдиги болуп саналат, алардан сөздүн формасы түзүлөт, андан ары, ошого жараша, лексема дагы. Кыргыз тилинде мүчөлөр төрт негизги формага бөлүнөт. Төмөнкү мүчөлөр сөздүн негизин аныктоочу иштелип жаткан алгоритмде колдонулат.

$P_i$–$i$ = 1, 2, 3, 4 - мүчөлөрдүн төмөнкү көптүктөрү.

Терминалдык символдор мүчөлөрдүн төмөнкү топтомун билдирет:

$P_1$ – үч тамгалуу мүчөлөрдүн көптүгү (көптүк форманын мүчөсү);

$P_2$ – мүчөлөрдүн көптүгү (илик жөндөмөсүнүн мүчөлөрү);

$P_3$ – мүчөлөрдүн көптүгү (жеке жалгоолору);

$P_4$ – мүчөлөрдүн көптүгү (жөндөмө мүчөсү).

1-таблицада морфемдик курамдын аныктоосу баяндалган ($P_i$, кайсы $i$ = 1, 2, 3, 4):

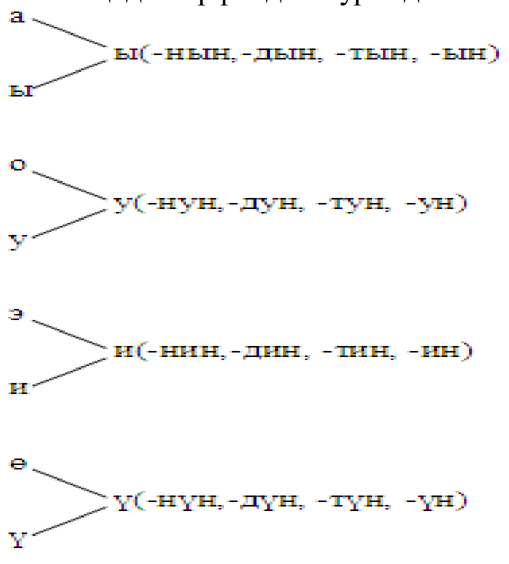

**Таблица 1**

**Мүчөлөрдүн көптүгү**

| | Мүчөнүн көрсөткүчтөрү | Мүчөлөр |
|---|---|---|
| | Көптүк көрсөткүчтөрү – $P_1$ | -лар,-дар,-тар,-лер,-дер,-тер,-лор,-дор,-тор,-лөр,-дөр,-төр. |
| | Таандык көрсөткүчтөрү – $P_2$ | -ым,-им,-ум,-үм,–м; -ың,-иң,-уң, –үң, –ң;-ыңыз,-иңиз, -уңуз, -үңүз,-ңыз, -ңиз,-ңуз,-ңүз; -ы, -и, -у, -ү, -ын, -ин, -ун, -үн,-сы, -си, -су, -сү, -сын, -син, -сун, -сүн; -ыбыз, -ибиз, -убуз, -үбүз, -быз, -биз, -буз, -бүз; -ыңар, -иңер, -уңар,-үңөр,-ңар,-ңер,-ңар,-ңөр; -ыңыздар,-иңиздер, -уңуздар, -үңүздөр, -ңыздар, -ңиздер, -ңуздар, -ңүздөр; -ы, -и, -у, -ү, -ын, -ин, -ун, -үн; -сы, -си, -су, -сү,-сын, -син,-сун, -сүн. |
| | Жак көрсөткүчтөрү – $P_3$ | -мын, -мин, -мун, -мүн, -сың, -сиң, -суң, -сүң,-сыз, -сиз, -суз, -сүз,-быз, -биз, -буз, -бүз,-сыңар,-сиңер, -суңар, -сүңөр,-сыздар,-сиздер, -суздар, -сүздөр |
| | Жөндөмө көрсөткүчтөрү -$P_4$ | -нын,-нин,-нун, -нүн, -дын, -дин, -дун,-дүн,-тын, -тин,-тун,-түн,-ын,-ин,-ун,-үн,-га,-ге,-го,-гө,-ка, -ке,-ко,-кө,-а,-е,-о,-ө,-ны,-ни,-ну,-нү,-ды,-ди,-ду,-тү,-ты,-ти,-ту,-тү,-ы,-и,-у,-ү,-да,-де,-до,-дө-та,-те,-то,-тө,-дан,-ден,-дон,-дөн,-тан,-тен,-тон,-төн, -ан,-ен,-он,-өн. |

Мүчөлөрдүн улануу эрежелерине ылайык төмөнкү белгилөө киргизилет:

A – көптүктү билдирүүчү мүчө+жөндөмө мүчө

B – көптүктү билдирүүчү мүчө+жак мүчө

C – көптүктү билдирүүчү мүчө+таандык мүчө

D – көптүктү билдирүүчү мүчө+таандык мүчө+жөндөмө мүчө

E –көптүктү билдирүүчү мүчө+таандык мүчө+жак мүчө

F – таандык мүчө + жөндөмө мүчө

G - таандык мүчө + жак мүчө жана

Q – бир орундуу предикат;

W – сөздүн негизги формаларынын көптүгү.

Ар бир $z$ сөздү $z = y+x$ эки же андан көп сөздөрдүн биригүүсү формасында алабыз.Эгерде сөз $x \in P_i$ болсо, анда бардык $i = 1, \ldots, 4$ үчүн $P_i(x)$деп алабыз.

Эгерде сөз $x \in W$ болсо, анда W(x) деп белгилейбиз.

Эгерде сөз $x \in Q$ болсо, анда $Q(x)$деп белгилейбиз.

Анда биздин A–G эрежелерибиз төмөнкү формулалар боюнча өзгөрөт [6].

Каалагандай $z$ сөзү $x_0 + x_1 + x_2 + \ldots + x_k$ тамгалардан турсун дейли. ($x_i$ –сөздөгү тамгалардын эң көбү).$i = k, x = x_i$ десек:

1-кадам. $$A = \begin{cases} P_4(x) \to Q(z\backslash x),\ \text{мында}\ z\backslash x = x_0 + x_1 + x_2 + \cdots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_1(x) \to Q(z\backslash x),\ \text{мында}\ z\backslash x = x_0 + x_1 + x_2 + \cdots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \end{cases}$$

Бул шарттар текшерилет, эгерде аткарылбаса 2-кадамга өтөбүз.

2-кадам

$$B = \begin{cases} P_2(x) \to Q(z\backslash x),\text{мында}\ z\backslash x = x_0 + x_1 + x_2 + \cdots + x_{i-1} \\ z = z\backslash x,\quad i = i-1 \\ P_1(x) \to Q(z\backslash x),\text{мында}\ z\backslash x = x_0 + x_1 + x_2 + \cdots + x_{i-1} \\ z = z\backslash x,\quad i = i-1 \end{cases}$$

$$3\text{-кадам} C = \begin{cases} P_4(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_3(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_1(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \end{cases}$$

$$4\text{-кадам} D = \begin{cases} P_2(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_3(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_1(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \end{cases}$$

5-кадам

$$E = \begin{cases} P_1(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x, \quad i = i-1 \\ P_2(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x, \quad i = i-1 \end{cases}$$

$$6\text{-кадам} F = \begin{cases} P_2(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_3(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \end{cases}$$

$$7\text{-кадам} G = \begin{cases} P_3(x) \to Q(z\backslash x), \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \\ P_1(x) \to Q(z\backslash x),\ \text{мында } z\backslash x = x_0 + x_1 + x_2 + \dots + x_{i-1} \\ z = z\backslash x,\ i = i-1 \end{cases}$$

Бул кадамдардан кийин сөздөрдүн акыркы символун текшерүү токтойт же 1-кадамга кайрадан кайтат. Аягында сөздүн негизин алабыз.

## Жыйынтык:

Бул макалада тексттеги сөздүн сөздүктөгү формасын табуунун негизги эки алгоритми каралды жана ошол алгоритмдердин негизинде жаңы модель түзүлдү. Бул модель жөнөкөйлүгү жана ишенимдүүлүгү менен айырмаланат.

## Адабияттар:


1. **Садыков, Т.** Кыргыз тилинин компьютердик лингвистикасынын негиздери [Текст] / Г.Э. Жумалиева, М.Ж. Түмөнбаева, Б. Шаршембаев // Бишкек, 2015.
2. **Абдувалиев, И.** Азыркы кыргыз тили: морфология [Текст] / Т. Садыков // Бишкек: Сорос-Кыргызстан, 1997.
3. **Федотов, А.М.** Модель определения нормальной формы слова для казахского языка [Электронный ресурс] / Д.А. Тусупов, М.А. Самбетбаева, А.С. Еримбетова, А.М. Бакиева, А.И. Идрисова // URL:https://cyberleninka.ru
4. Азыркы кыргыз адабий тили. Бишкек: Аврася-Пресс, 2009.
5. **Porter, M.F.** Analgorithm for suffix stripping // [Text] Program.1980, T.14, №3. P. 130–137.
6. **Willett, P.** The Porter stemming algorithm: then and now // [Text] Program: Electronic Library and Information Systems.2006. B. 3, Vol.40, C. 219–223.
7. **Segalovich, I.** «A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine» [Text] MLMFA, 2003. P. 273–280.