O‘ZBEKISTON MILLIY UNIVERSITETI XABARLARI, 2021, [1/4/1]
ISSN 2181-7324

FILOLOGIYA
www.uzmuxabarlari.uz
Social sciences

УДК: 811.512.133’373.004

Oqila ABDULLAYEVA,
Toshkent davlat o‘zbek tili va adabiyoti universiteti tayanch doktoranti PhD
E-mail: abdullayevaoqila@gmail.com

ToshDO‘TAU dotsenti, filol.f.n. Azimova I.A. taqrizi asosida

THE PROCESS OF LEMMATIZATION, STEMMING AND TOKENIZATION IN CORPUS LINGUISTICS

Abstract

Although ceative models of corpus and corpuses' linguistic issues are general, but internal forms of corpuses, language models, lemma, the processes of token and tagged, to prepare morphological analysis of corpuses, the problems of results are diffirentiated with each other. Text context which is downloaded to corpus is divided to automatical tokens. Next, every linguistic unit is indicated as one token. In the annotation process of speech parts, lemma which stem word is clarified. This process is named tokenization and lemmatization. The process of lemmatization and stemming tied with each other. In the corpus lingusitics, the charecteristics of these processes, their positive and negative aspects, advantages and disadvantages are not fully compared and analyzed sistematically.

Key words: corpus, lemmatization, tokenization, stemming, NLP, NLU, morphological analysis, annotation.

ПРОЦЕСС ЛЕММЫ, СТЕММИНГА И ТОКЕНИЗАЦИИ В КОРПУСНОЙ ЛИНГВИСТИКЕ

Аннотация

Хотя модель создания языкового корпуса и лингвистические проблемы, решаемые корпусом, схожи, но различаются конкретными проблемными аспектами, такими как внутренние особенности корпуса, языковые шаблоны, лемма, токен и процесс маркировки, подготовка к морфологическому анализу, затраты на получение результатов анализа. Текстовый контекст, загруженный в корпус, автоматически разделяется на токены. Каждая языковая единица обозначается как отдельный токен. В процессе аннотирования частей речи определяется лемма слова, то есть его основная часть. В корпусной лингвистике этот процесс называется токенизацией и лемматизацией. Лемматизация и стемминг тоже связаны. Хотя мы часто сталкиваемся с ними в области корпусной лингвистики, характеристики, преимущества и недостатки, преимущества и ограничения этих процессов не подвергались последовательному анализу.

Ключевые слова: корпус, лемматизация, токенизация, стемминг, обработка естественного языка, понимание естественного языка, морфологический анализ, аннотация.

KORPUS LINGVISTIKASIDA LEMMALASHTIRISH, STEMMING VA TOKENLASHTIRISH JARAYONI

Annotatsiya

Til korpuslarini yaratish modeli va korpuslar hal qiladigan lingvistik masalalar o‘xshash bo‘lsa-da, ammo korpuslarning ichki xususiyatlari, til qoliplari, lemma, token va teglash jarayoni, morfologik tahlilga tayyorlash, tahlil natijalarini olish kabi o‘ziga xos muammoli jihatlari bilan farqlanib turadi. Korpusda yuklangan matn konteksti avtomatik tokenlarga ajratiladi. Har bir til birligi bitta token sifatida belgilanadi. Nutq qismlarini annotatsiyalash jarayonida so‘zning lemmasi, ya’ni o‘zak qismi aniqlanadi. Bu jarayon korpus lingvistikasida tokenizatsiya va lemmatizatsiya deb nomlanadi. Lemmatizatsiya va stemming ham bir-biri bilan bog‘liq. Korpus lingvistikasi sohasida ham ko‘p bora duch kelsakda, mazkur jarayonlarning xarakteristikasi, ijobiy va salbiy tomonlari, afzalliklari va cheklovlari qiyosan bir-biridan farq qilishi izchil tahlilga tortilmagan.

Kalit so‘zlar: korpus, lemmalashtirish, tokenlashtirish, stemming, tabiiy tilni qayta ishlash, tilni tushunish, morfologik tahlil, izohlash.

Kirish. Stemming va Lemmatizatsiya - bu matnni, so‘zlarni va hujjatlarni navbatdagi ishlov berish uchun tayyorlash maqsadida ishlatiladigan tabiiy tilni qayta ishlash sohasidagi Matnni tartibga keltirish (yoki ba’zida so‘zni normallashtirish deb ham atashadi) texnikasi. Stemming va lemmatizatsiya o‘rganilib, algoritmlari 1960-yildan boshlab komyuter fanlari negizida ishlab chiqilmoqda. Lemmatizatsiya, stemming va tokenizatsiya tilni tushunish (NLU), tabiiy tilni qayta ishlash (NLP) va axborot qidiruvda (Information Retrieval, Text Mining) qo‘llaniladigan yuqori darajadagi texnika hisoblanadi.

Mavzuga oid adabiyotlarning tahlili. Sketch Engine korpus menejerida lemmatizatsiya va tokenizatsiya jarayonlari qisqacha tahlil qilingan. Lemmatizatsiya - bu lemmatizator deb ataladigan avtomatik vosita yordamida korpusdagi har bir so‘z shaklining lemmasini, ya’ni o‘zak qismini aniqlashi, shu jarayonda so‘zning asosiy shaklini izlash bilan barobar so‘z orqali vujudga kelgan barcha so‘z shakllarini ko‘rish mumkinligi ta’kidlangan. Misol tariqasida ingliz tilidagi *go* lemmasi orqali will *go, goes, went, gone, going* shakllariga duch kelish mumkin. Mazkur shakllari har qanday kontekstda uchraganda ham, lemma *go* deb belgilanadi.

Tokenizatsiya lemmatizatsiya jarayonidan farqlidir. Matnni qismlarga, ya’ni tokenlarga ajratuvchi holatdir. Bunda tokenizatorlar deb ataluvchi dasturiy ta’minot matnlarni tokenlarga ajratadi. Tokenizatorlar har bir tilga xos bo‘lib, tilning o‘ziga xos xususiyatlarini hisobga oladi. Masalan, ingliz tilida *don’t* ikkita token sifatida tokenlashtiriladi.

Token bu til korpusi ichidagi eng kichik birlik hisoblanib, quyidagi shakllar tokenlardir:
So‘z shakli: *bordim, O‘zbekistonda, o‘qituvchi, covid…..*
Tinish belgilari: *vergul, nuqta, so‘roq belgisi…..*
Raqam: *38, 7500……*
Qisqartmalar, mahsulot nomlari: *NLP, BNC, YIMM…* hamda oraliqlar orasidagi boshqa birliklar.

Tokenlarning ikki turi mavjud: so‘zlar va so‘z bo‘lmaganlar (nonwords). Korpuslar so‘zlardan ko‘ra ko‘proq

tokenlarni o‘z ichiga oladi. Matnlar har bir til uchun maxsus yaratilgan tokenizator dasturi orqali tokenlarga ajratiladi [1].

O‘zbek tili axborot matnlari korpusida korpusga kodlangan dastur orqali matnlar avtomatik tokenlarga ajratiladi. Har bir oraliqdagi birliklar alohida bitta token sifatida olinadi. Bu lingvistga nutq birlikarini teglashi uchun qulay bo‘lsa-da, lekin ba’zi muammoli tomonlari ham bor. Masalan, ( ) qavslarning har ikkala tomoni ham alohida token sifatida ajratiladi. Aynan token ajratib bergan nutq qismi bo‘lgan so‘zlarning har biri lemmaga ajratilib, izohlanayapti.

S.A. Koval monografiyasida lemmatizatsiyani “leksik birliklarning invariantlarini aniqlash, ya’ni yagona leksemaga qadar” deb ta’riflaydi. [2]. K.Sipunin esa lemmatizatsiya har qanday kirish so‘z shakli uchun qaysi leksemaga tegishli paradigmani aniqlashga imkon beradigan va natijada ushbu ikkinchisining so‘z birikmasi nomi - lemmani aniqlab beradigan analitik protsedura deya bayon qiladi. U tadqiqotida lemmatizatsiya algoritmining chegaralarini belgilab berishga harakat qilgan. Bunda 2 ta asosiy eslatma mavjudligini ta’kidlaydi, ya’ni birinchidan, u faqat morfologik izohli so‘z shakllariga yo‘naltirilgandir, mavjud manbalarga (leksik qoidalar yoki grammatik lug‘at bo‘lsin) tayanmasdan lemmatizatsiya 100 % hollarda asosli va lingvistik jihatdan to‘g‘ri bo‘lolmaydi. Ikkinchidan, og‘zaki nutq qismlarining barchasini lemmalarga ajratish murakkabdir, algoritm faqat nominal nutq qismlarining so‘z shakllarini - ismlar, sifatlar, sonlar va olmoshlarni, shuningdek, o‘zgarmas so‘zlarni qayta ishlashga qodir deb hisoblaydi. Sipunin qarashlariga ko‘ra, protsessual ravishda ushbu ishda amalga oshirilgan algoritm jarayonida quyidagi transformatsiyalar ketma-ket kiritilgan so‘z shakllariga qo‘llaniladi: (1) imlo normallashishi, (2) stemming, (3) lemmaning tiklanishi; bundan tashqari, morfologik belgini qayta ishlashga tayyorlashning dastlabki bosqichida unga ma‘lum o‘zgarishlar kiritiladi [3].

Tadqiqot metodologiyasi. Stemming va lemmatizatsiyaning yonma-yon kelish holatlarini ko‘rish mumkin. Ba’zi manbalarda lemmatizatsiya stemmingning asosiy algoritmlaridan biri hisoblanadi. Chunki lemmatizatsiyada so‘zning o‘zak holati yoki kontekstdagi qaysi so‘z turkumiga oidligi namoyon bo‘ladi. Stemmingda ikkita yondashuv mavjud. Birinchi yondashuv stemmingning o‘zi hisoblanib, uning maqsadi kontekstda affikslarni aniqlash va ularni olib tashlash. Ikkinchi yondashuv - lemmatizatsiya. Lemmatizatsiya jarayonida ishlab chiquvchi til va uning grammatikasini yaxshi bilishi kerak. Lemmatizatsiya uchun lug‘atni qidirish kerak; shuning uchun lemmatizatsiya stemmingdan ko‘ra murakkabroq. Biroq, lemmatizatsiyalashda aniqroq natijani kutish mumkin. Masalan, “*better*” so‘zi “*good*” lemmasiga ega. Algoritmda lug‘at qidirish jadvalidan foydalanilmasa, ushbu turdagi so‘zlarning o‘zagini topish mumkin emas. Lemmatizatsiya algoritmi bu so‘zning o‘zagini topishning murakkab usuli. Ushbu jarayon so‘z yoki nutq qismi (POS) kontekstini tushunishni va har bir POS uchun turli xil normallashtirish qoidalarini qo‘llashni talab qiladi. Lemmatizatsiya algoritmini qo‘llash uchun biz tilni va uning grammatikasini tushunishimiz kerak. Lemmani topish uchun turli xil tillarda har xil qoidalar talab qilinishi mumkin. Turli tillarda so‘zlar bir nechta qo‘shma shakllarda bo‘lishi mumkin. Masalan, ingliz tilida “*to talk*” fe’llari *talking, talked va talk* kabi uchrashi mumkin. Agar so‘zning tayanch shakli gap qismi bilan kelgan bo‘lsa, uni odatda so‘z leksemasi deyiladi [4].

Ham stemming, ham lemmatizatsiyaning maqsadi so‘zning flektiv shakllarini va ba‘zan derivativ jihatdan bog‘liq shakllarini umumiy asos shakliga kamaytirishdir.

Biroq, bu ikki jarayon o‘zlarining uslubi bilan farq qiladi. Stemming, odatda, ushbu maqsadga ko‘pincha to‘g‘ri erishish maqsadida so‘zlar oxiridagi qo‘shimchalarni kesib tashlaydigan va ko‘pincha hosila qo‘shimchalarini olib tashlashni o‘z ichiga olgan qo‘pol evristik jarayonni anglatadi. Lemmatizatsiya esa so‘zlarni lug‘at va morfologik tahlil yordamida to‘g‘ri bajarishni anglatadi, odatda faqat fleksion qo‘shimchalarni olib tashlashni va lemma deb ataladigan so‘zning asosini yoki lug‘at shaklini qaytarishni maqsad qiladi [5].

Stemming - bu Text Mining dasturlarida oldindan ishlov berish bosqichi, shuningdek Natural Language ishlov berish funksiyalarining juda keng tarqalgan talabi. Aslida bu Axborot qidirish tizimlarining aksariyat qismida juda muhimdir. Stemmingning asosiy maqsadi so‘zning turli grammatik shakllarini / so‘z shakllarini ot, sifat, fe’l, ergash gap va boshqa shu kabi shakllarini tub shakliga kamaytirishdir. Aytish mumkinki, stemmingning maqsadi so‘zning flektiv shakllarini va ba’zan derivativ jihatdan bog‘liq bo‘lgan shakllarini umumiy bazaviy shaklga kamaytirishdir [6].

Stemming va lemmatizatsiya o‘rtasida juda nozik farq bor deb hisoblashadi. Ya’ni, stemmingda “stem”ni topish uchun so‘zning paydo bo‘lish konteksti yoki biror qoidalarsiz mashhur prefix va suffikslarni olib tashlash orqali so‘zning asosini topish bo‘lsa, lemmatizatsiyada so‘zdagi qo‘shimchalarni faqat qoidalarga va kontekstga asoslanib ajratadi.

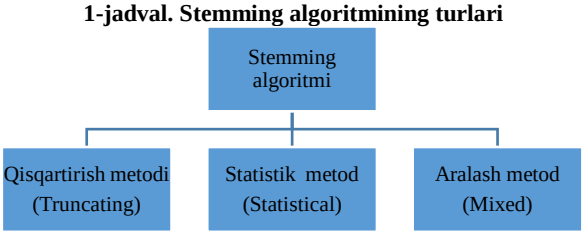
Tadqiqotlarda stemmerlardan foydalanishda e’tiborga olinadigan 2 ta muhim nuqta bor deb qaraladi:

- so‘zning morfologik shakllari bir xil asosiy ma’noga ega deb taxmin qilinadi, shuning uchun bir xil stem belgilandi;
- bir xil ma’noga ega bo‘lmagan so‘zlar alohida saqlanadi [6].

Tahlil va natijalar. Lemmatizatsiya esa kontekst va so‘zning semantikasini hisobga oladigan murakkab jarayon hisobalanadi. Til korpuslarida izohlashning eng asosiy turlaridan biri bu lemmatizatsiya, har bir so‘zni korpusda uning asosini (iqtibos yoki lug‘at) aniqlash va belgilash jarayonidir. Stemming dasturlar yaratish osonroq bo‘lishi mumkin, lekin lemmatizatorlar yaratish har bir til uchun murakkablik kasb etadi. Masalan, ingliz tilidagi *gone, goes, going will go* so‘zlarining stemmi “*go*” deb belgilanadi, ammo *went* so‘zi alohida so‘z sifatida saqlanadi. Lemmatizator *go* qatoriga *went* ni ham kiritadi. Yoki stemmingda *introduction, introducing, introduces* so‘zlarining asosini *introduc* deb belgilasa, lemmatizator *introduce* ni ko‘rsatadi. Bu holat bitta kamchilik bo‘lsa, yana bitta xatosi stemmingda 2 xil o‘zakli so‘zning bir ildizga birikishidir. Masalan, *saw* va *saw* so‘zlari ingliz tilida biri – *ko‘rmoq* so‘zining o‘tgan zamon shakli bo‘lsa, ikkinchisi *arra* degan tarjimani bildiradi. Lekin stemming kontekstni hisobga olmaydi va har ikki o‘rinda ham *arra* deb belgilashi mumkin. O‘zbek tilidagi *qatlama – qatla(ma), tilim- til(im)* so‘zlarining stemi qatla yoki qat va til sifatida belgilanadi. Stem va lemma o‘rtasida yana bitta katta va asosiy farq bor. Stem barcha qo‘shimchalardan holi so‘z ildizining o‘zidir, lemma esa kontekst bilan bog‘liq bo‘lib, fleksionlardan holi, lekin derivativlari bilan birga holatidir. Masalan, *ishchi, ishchilar, ishladi, ishchilarning* so‘zlarida *ish* stem bo‘lsa, lekin *ishchi, ishchilar, ishchilarning* so‘zlarida lemma *ishchi*, *ishladi* fe’lida esa lemma *ishla* hisoblanadi.

A.Ganesh stemming algoritmining 3 ta turi mavjud deb hisoblaydi.

1-jadval. Stemming algoritmining turlari



Qisqartirish usuli eng muhim stemmer metodlaridan biri sanaladi, chunki bu usulda barcha so‘zlarning qo‘shimchalarni olib tashlaydi. Ganesh maqolasida qisqartirish metodida mashhur va xatoliklari bir buncha kam bo‘lgan Lovin, Porter, Paice/Husk, Dawson stemmerlarini tahlil qilgan. Barchasida o‘xshash umumiyliklar bor, masalan, qoidalar to‘plamini yaratish va uchrashi mumkin qo‘shimchalarni belgilab olish.

Statistik metod tahlil va texnikaga asoslangan usuldir. Bu usulga N-Gram, HMM, YASS stemmerlarini misol sifatida keltirgan.

Aralash metodlar orasida korpusga asoslangan stemming usuli ham bor. Afzallik tomoni shundaki, qo‘l mehnati bilan to‘g‘rilash mumkin. Cheklov tomoni har bir korpusda alohida ishlanadi, bir ikkinchiga to‘g‘ri kelmaydi.

Bugungacha dunyo tillarining ko‘pchiligida stemming dasturlari mavjud. Shu jumladan, o‘zbek tilida ham ishlatilishi mumkin bo‘lgan stemmer dasturi A.Ismailov tomonidan yaratilgan. O‘zbek tili uchun stemmer dasturi yaratishda ingliz tili uchun moslangan Lovins, Porters, Dawson, Pace/Husk stemmerlari o‘rganilib, o‘xshash va farqli tomonlari tahlil qilingan. Barchasida bitta maqsad, so‘zlarning qo‘shimchalarini olib tashlash va ularda o‘zgarish transformatsiyasini ko‘rsatish. O‘zbek tili uchun stemmer yaratishda Lovins stemmer modelidan foydalanilgan [7]. Ismailov tadqiqotida stemmerni rivojlantirishning asosiy qiyinligi - bu tillarni tushunishdir deb hisoblaydi. Sababi shundaki, har qanday stemmerning asosiy maqsadi hosil qilingan so‘zlarning ildiz shaklini uning qo‘shimchalarini (suffiks va prefiks) olib tashlash orqali topishdir. So‘z qo‘shimchalarini olib tashlash jarayonida ishlab chiquvchi stemmer qaysi tilda ishlatilishini yaxshi bilishi kerak. Ko‘pgina stemmerslar tilga bog‘liq, ya’ni stemmer faqat ma’lum tillarda ishlaydi. O‘zbek tili agglyutinativ til sanaladi. O‘zbek tilining morfologik shakli ingliz tilidan juda farq qiladi. Ingliz tilida so‘z shakllari qo‘shimcha olishi bilan o‘zgarishga uchraydi yoki umuman o‘zgarmaydi. Masalan, *go* fe’li *going*, *gone*, *will go* shakllaridan tashqari o‘tgan zamon shaklida *went* ga o‘zgaradi. Yoki *fly* fe’li *flying* shaklidan boshqa turli zamonlarda *flew*, *flown* shallariga o‘zgaradi. O‘zbek tili - boy morfologik tuzilmalarga ega bo‘lgan agglyutinativ til. O‘zbek tilida ildiz so‘zlarga suffikslar va prefikslar qo‘shib yaratiladi. Ba‘zi o‘zbek so‘zlari ildiz so‘zlariga ikki yoki undan ko‘p qo‘shimchalarni qo‘shish orqali tuzilgan, inglizcha so‘z shakllari esa hosilaviy ravishda o‘zgaradi. Masalan, *arra* - *arrala* (*handsaw* “ism”) - (“fe’l” *saw*) *kuch*– *kuchli* (*strength*) - (*strong*) *hosil* - *serhosil* (*crop, harvest*) - (*highly productive*).

A.Ismoilov tadqiqotlarida Lovins stemmeri modelidan o‘zbek stemmer algoritmini ishlab chiqish g‘oyasini ilgari surgan. Yuqorida ta’kidlaganimizdek, Lovins stemmeri - bu qo‘shimchani olib tashlovchi algoritmlaridan biri. Lovins stemmeri ikkita protseduraga ega:

1. Stemmingning asosiy protsedurasi.
2. Qayta yozish tartibi (Transformatsiya qoidalari)

Birinchi protsedurada Lovins stemmeri so‘zdagi eng uzun qo‘shimchani topadi va olib tashlaydi. Lovinsda stemming protsedurasiga erishish uchun turli xil qoidalar mavjud. Birinchi qoida - kiritilgan so‘z kamida ikkita yoki undan ko‘p belgidan iborat bo‘lishi kerak. Keyingi jarayon - kiritilgan so‘zdan suffiksni izlash, agar suffikslar topilsa, qo‘shimchani olib tashlash. Ikkinchi protsedura - asosiy so‘zni ma‘noli inglizcha so‘zga aylantirish. Masalan, agar “stemming” so‘zi va Lovins stemmer birinchi protsedurada “ing” ni olib tashlagan bo‘lsa, u holda qayta ishlash protsedurasida Lovins stemmer “stemm” dan juft “mm” ni bitta “m” ga o‘zgartirishi kerak, yakuniy natija “stem” bo‘ladi. Ammo bu stemmer modelini to‘g‘ridan-to‘g‘ri o‘zbek tiliga targ‘ib qilib bo‘lmasdi, chunki u so‘zdagi prefikslarni olib tashlamaydi. O‘zbek tilida esa so‘zlar yasalishida prefikslarning alohida o‘rni mavjud. Shuning o‘zbek stemmerida jarayon biroz o‘zgartirildi, ya’ni 1) lug‘atni tekshirish (O‘zbek tili imlo lug‘ati dasturga yuklab qo‘yiladi); 2) qo‘shimchalarni olib tashlash. O‘zbek tili stemmerida 14 000 ta stem ma’lumotlar bazasiga joylashtirilgan, shuningdek stemming tildagi 5 ta prefix va 42 ta suffiks orqali matndagi so‘zlarning stemini ajratib bera oladi [8]. Ammo o‘zbek tili boy va grammatikasi murakkab tillardan biri hisoblanadi. Taxminiy hisob-kitoblarga ko‘ra tilda 100 dan oshiq so‘z yasovchi qo‘shimchalar va 70 ga yaqin grammatik shakl hosil qiluvchi qo‘shimchalar mavjud. Shuning uchun ham A.Ismoilov tomonidan yaratilgan o‘zbek tili stemmeri ma’lumotlar bazasiga kiritilmagan so‘zlar va qo‘shimchalar tufayli yuqori ko‘rsatgich bilan to‘g‘ri natijalar bera olmaydi. Lekin bu dastur o‘zbek tilida so‘z stemini ajratib bera olgan ilk ish sifatida baholanadi va u o‘z darajasida muvaffaqiyatga erishgan.

Xulosa va takliflar. Tadqiqot ishimizning asosiy qismi hisoblangan O‘zbek tili axborot matnlari korpusida har bir nutq qismi grammatikaga oid kitoblar va lug‘atlar yordamida qo‘l mehnati bilan lemmaga ajratilmoqda. Dunyo tajribasida ko‘rish mumkinki, ilk korpus namunalarida matnlar ustidagi barcha amallar qo‘l mehnati bilan amalga oshirilgan. Ammo yuqoridagi ilmiy nazariyalardan shuni aniqladikki, o‘zbek tili lug‘ati va barcha qo‘shimchalarni ma’lumotlar bazasiga yuklab, o‘zbek tili lemmatizatori algoritmini yaratsa bo‘ladi. Albatta, ilk dasturlarda xatoliklar uchrashi mumkin, chunki til murakkab tuzilmalarga ega, tabiiy til namunalarida kutilmagan qoliplarga duch kelish ehtimoli doim yuqori bo‘lgan.

ADABIYOTLAR

1. https://www.sketchengine.eu/my_keywords/lemmatization
2. Коваль С. А. Лингвистические проблемы компьютерной морфологии. — СПб., 2005.. 76
3. Сипунин Константин Владимирович. Автоматическая лемматизация текстов в корпусе СКАТ на основе морфологической разметки, - Санкт-Петербург, 2018
4. Ingason, Anton Karl, Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. "A mixed method lemmatization algorithm using a hierarchy of linguistic identities (HOLI)." In Advances in Natural Language Processing, pp. 205-216. Springer Berlin Heidelberg, 2008.
5. https://stackoverflow.com/questions/1787110/what-is-the-difference-between-lemmatization-vs-stemming

6. Anjali Ganesh Jivani. A Comparative Study of Stemming Algorithms. IJCTA. Vol 2 (6), 1930-1938. https://www.researchgate.net/publication/284038938
7. Ismailov, M.M. Abdul Jalil, Z. Abdullah and N.H Abd Rahim. A comparative study of stemming algorithms for use with the Uzbek language// 3rd International conference on computer and information sciences (ICCOINS). 2016. 7-12 https://www.researchgate.net/publication/311931154
8. M.M.Jalil, A.Ismailov, N.H.Abd Rahim, Z.Abdullah. The Development of the Uzbek Stemming Algorithm. Advanced Science Letters, 23/5. 2017/5/1. 4171-4174.