M U N D A R I J A

ADABIYOTSHUNOSLIK

TILSHUNOSLIK

TILSHUNOSLIK

FarDU. Ilmiy xabarlar – Scientific journal of the Fergana State University

Volume 31 Issue 3, 2025-yil DOI: 10.56292/SJFSU/vol31_iss3/a120

UO‘K: 811.8

O‘ZBEK TILI IYERARXIK KORPUSINI YARATISHDA MA’LUMOTLAR BAZASINI YIG‘ISH, GAPLARNI ANNOTATSIYALASH VA TOKENLASH JARAYONI

THE PROCESS OF COLLECTING A DATABASE, ANNOTATING SENTENCES, AND TOKENIZING IN THE CREATION OF A DEPENDENCY PARSING OF THE UZBEK LANGUAGE

ПРОЦЕСС СБОРА БАЗЫ ДАННЫХ, АННОТИРОВАНИЯ ПРЕДЛОЖЕНИЙ И ТОКЕНИЗАЦИИ ПРИ СОЗДАНИИ СИНТАКСИЧЕСКОГО АНАЛИЗА ЗАВИСИМОСТЕЙ УЗБЕКСКОГО ЯЗЫКА

Raximboyeva Xulkar G‘ayratovna[1]
[1]O‘zbekiston Milliy universteti tayanch doktoranti

Matlatipova Muslima G‘ayrat qizi[2]
[2]Urganch davlat universiteti talabasi

Annotatsiya

Ushbu maqolada o‘zbek tili uchun iyerarxik tahlil korpusini yaratish bosqichlari keng yoritilgan. Xalqaro tajribada biror tilning iyerarxik tahlil korpusini yaratish uchun qo‘llaniladigan beshta asosiy soddalashtirilgan bosqich mavjud bo‘lib, ular batafsil ko‘rib chiqiladi. Ushbu bosqichlar quyidagilar: matnlarni tanlash; oldindan ishlov berish vositalari va manbalarini tanlash; annotatsiyalash; o‘zbek tiliga xos qo‘llanmani hujjatlashtirish va tilning universal bo‘lmagan xususiyatlarini aks ettirish; transliteratsiya bosqichidir..

Abstract

This article provides an in-depth overview of the stages involved in building a syntactic treebank for the Uzbek language. It outlines five simplified but essential steps widely used in international practice when constructing a hierarchical corpus for any language. These steps include: text selection; pre-processing (including the selection of tools and resources); annotation; documentation of language-specific guidelines and addressing non-universal linguistic features; and finally, transliteration.

Аннотация

В данной статье представлен подробный обзор этапов создания дерева зависимостей (treebank) для узбекского языка. Рассматриваются пять упрощённых, но ключевых этапов, которые широко применяются в международной практике при построении иерархического корпуса для любого языка. Эти этапы включают: выбор текстов; предварительная обработка (включая выбор инструментов и ресурсов); аннотирование; документирование языковых особенностей и описание специфических, неуниверсальных черт языка; и, наконец, этап транслитерации.

Kalit so‘zlar: *Annotatsiya, bosqich, tokenlash, lemmalash, matn tanlovi, hujjatlashtirish, yo‘riqnoma, natija, jarayon.*

Key words: *Annotation, steps, tokenization, lemmatization, text selection, documentation, guideline, result, process.*

Ключевые слова: *аннотация, этапы, токенизация, лемматизация, выбор текстов, документация, руководство, результат, процесс*.

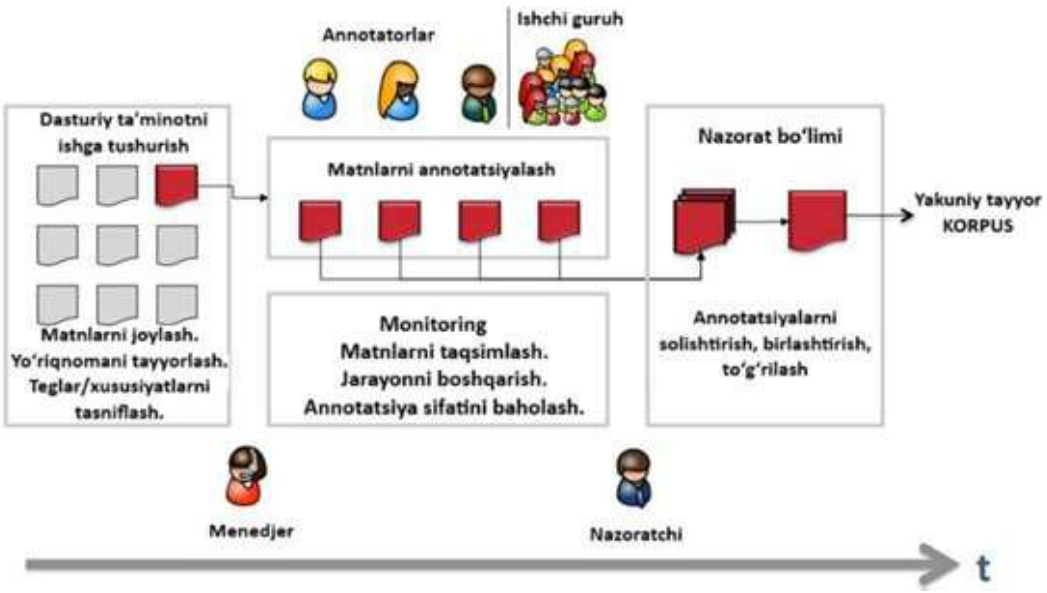
KIRISH

O‘zbek tili uchun iyerarxik tahlil korpusi (dependency treebank)ni yaratish bosqichlari batafsil tavsiflanadi. Korpus uchun kerakli bo‘lgan til materiallarini yig‘ish, ularni maxsus vositalar yordamida annotatsiyalash (teglar bilan belgilash) va tokenlash (so‘zlarni alohida tokenlarga ajratish) jarayonlari izchil bayon qilinadi. Korpus tuzish jarayonida ochiq manbadagi onlayn yangiliklar matnlari hamda mualliflik huquqi bo‘yicha foydalanishga ruxsat etilgan badiiy adabiyotlardan olingan jumlalar asos bo‘ldi. Annotatsiyalash ishlari Germaniyaning Zaarland universitetida ishlab chiqilgan INCEpTION dasturiy platformasi yordamida amalga oshirildi. Jarayonga jami 5 nafar annotator (izohlovchi) jalb etildi, ularning har biri o‘zbek tilining sintaksis va

morfologiyasini yaxshi biluvchi mutaxassislardir. Annotatorlar tomonidan matnning turli qismlarini belgilash ishlari amalga oshirildi. Korpus yaratilishida har bir qadamda sifatni ta'minlash uchun bir necha bosqichli nazorat va tahrir jarayonlari tashkil etildi.

ADABIYOTLAR TAHLILI VA METODOLOGIYA

Ma'lumotlarni yig'ish va tanlash. Dastlab, korpus uchun manbalar tanlab olindi. Ikki turdagi manbadan foydalanilgan: birinchisi, internetdagi ochiq axborot manbalaridan olingan yangiliklar maqolalari; ikkinchisi, jamoatchilik uchun ochiq e'lon qilingan yoki ruxsat olingan XX-XXI asrga oid o'zbek badiiy adabiyotlaridan olingan matnlar. Yangiliklar qismi asosan UzCrawl datasetidan olingan bo'lib (M. Mamasaidov va A. Shopulatov, 2023 yilgi loyiha) – ushbu dataset zamonaviy o'zbek tilidagi turli mavzulardagi yangilik saytlaridagi (masalan, Kun.uz, Daryo.uz kabi) maqolalarni jamlagan [7]. Badiiy adabiyotlar qismi esa internetda ochiq mavjud bo'lgan o'zbek adabiy asarlaridan tanlab olingan. Tanlash jarayonida materiallarning mavzuyi qamrovi keng bo'lishi, til uslubi va murakkabligi jihatdan xilma-xillik bo'lishiga e'tibor berildi. Shu tariqa, j*ami 1200 ta jumla* tanlab olindi — shundan 750 tasi yangilik matnlaridan, 500 tasi badiiy asarlardan. Tanlangan jumlalar korpusga kiritilishdan avval qo'lda tahrir qilinib, ortiqcha yoki takror bo'lib qolgan qismlar chiqarib tashlandi, imlo va tinish belgilari me'yorlari tekshirildi. Natijada, korpusga kiritilayotgan har bir jumla mustaqil va to'liq mazmunli bo'lishi ta'minlandi. Ushbu tanlama korpus kelgusida ham kengaytirilishi mumkin bo'lsa-da, hozircha u mazkur tadqiqot doirasida o'zbek tili sintaksisini qamrab oluvchi yetarlicha reprezentativ kichik hajmli korpus vazifasini o'tashga mo'ljallandi.



1-rasm. Korpusni izohlash (annotatsiya qilish) va yaratish jarayoni.

Annotatsiya vositasi va jamoaviy ishlash. Tanlab olingan matnlar INCEpTION dasturiga yuklanib, har bir jumlani teglar bilan belgilash (annotatsiya qilish) jarayoni bajarildi. INCEpTION — bu matnni annotatsiya qilish uchun mo'ljallangan ochiq platforma bo'lib, unda turli xil annotatsiya qatlamlarini (masalan, morfologik teglar, sintaktik bog'lanishlar va h.k.) ishlab chiquvchi annotatorlar jamoaviy ishlashi mumkin [5]. Platforma bir vaqtda bir nechta annotatorning ishlashini qo'llab-quvvatlaydi, o'zgarishlarni kuzatish imkonini beradi. Bizning tadqiqot uchun INCEpTION'da maxsus Universal Dependencies (UD**)** sxemasiga mos keluvchi annotatsiya loyihasi yaratildi. Unda har bir so'z uchun tegishli *morfologik xususiyatlar* (masalan, asosiy so'z turkumi (UPOS), grammatik kategoriyalar, lemma) va sintaktik bog'lanish ma'lumotlarini kiritish uchun maydonlar belgilandi. Dastur interfeysida jumla so'zlari ro'yxat shaklida va daraxt ko'rinishida chiqib, annotatorlar har bir so'zning xususiyatlarini tanlashlari va uning gap ichida qaysi boshqa so'zga bog'lanishini ko'rsatishlari (bog'lovchi o'q orqali) lozim bo'ldi. Annotatorlar bir-birlarining ishi ustidan ham kuzatib, izohlar qoldirish orqali bahsli holatlarda umumiy muhokama orqali to'g'ri kelishuvga erishildi. Ayniqsa, murakkab sintaktik konstruktsiyalar yoki noaniq ma'noli birikmalar uchraganda,

jamoa a’zolari lingvistik adabiyotlarga tayangan holda birgalikda qaror qabul qildilar. Yakunda, har bir jumla bo‘yicha annotatsiyalar yetakchi lingvist-annotator (korpus muallifi) tomonidan to‘liq ko‘rib chiqildi – bu bosqichda barcha xatolar tuzatildi, bir xil turdagi hodisalarning izchil belgilangani tekshirildi va korpus umumiy yagona uslubda annotatsiya qilinganiga ishonch hosil qilindi. Shunday qilib, jamoaviy mehnat va yakuniy tahrirlash natijasida korpusning annotatsiya sifati yuqori darajaga ko‘tarildi.

Tokenlash va dastlabki avtomatik qayta ishlash. So‘zlar odatda bo‘sh joy bilan ajratiladi, lekin bunda quyidagi mustasnolar mavjud:

- Chiziqcha bilan ajratilgan qo‘shma so‘zlar va to‘liq takrorlashlar (masalan, kuta-kuta, bora-bora), agar har bir so‘zni bir so‘z sifatida qabul qilinsa, ko‘p so‘zli tokenlar yaxlit holda shakllanadigan alohida semantik ifodadir.

- Bog‘lovchi “bo‘lib”, “holda” hol belgilari so‘zga qo‘shilganda ko‘p so‘zli leksema sifatida qabul qilinadi. Bo‘shliqli so‘zlar o‘zbek tilida uchrab turadi.

Korpusda uchta maxsus dastlabki ishlov berish bosqichlari amalga oshirildi:

• metama’lumotlarga qo‘shish - bu har bir namunaviy jumlaga kuzatish uchun identifikatorni, shuningdek, har bir jumla uchun ID berishni o‘z ichiga oladi, korpus ma'lumotlarida metama’lumotlar (#) raqam belgisi bilan ifodalanadi;

• xatolarni o‘chirish - bu matn terish xatolarini tuzatish, bo‘sh joylarni tekshirish, yetishmayotgan tinish belgilarini qo‘shish va jumlalarni formatlashni o‘z ichiga oladi;

• imloviy standartlashtirish – bu turli til darajalari (rasmiy, norasmiy, jargon, dialektlar va boshqalar) uchun imlo qoidalarini bir xil standartga moslashtirish jarayonidir, tilni tahlil qilishda izchillikka olib keladi, mashina tarjimasi va NLP tizimlari uchun aniqroq ma’lumot bazasi quriladi. Biroq imloni standartlashtirish qat’iy bo‘lishi shart emas — asosan, o‘zbek tili so‘zlovchilari uchun tabiiy bo‘lgan ifodani saqlab qolish va imloni ortiqcha murakkablashtirmaslik o‘rtasidagi muvozanatni ta’minlash maqsadida, iyerarxik tuzilmadagi lemmalarning o‘zgarishi orqali amalga oshirilgan.

Imloviy standartlashtirish asosiy jihatlari quyidagicha:

birlashtirilgan imlo qoidalari – korpusdagi barcha matnlar (ilmiy, adabiy, matbuot, internet-muloqot) umumiy imlo normalariga moslashtiriladi;

darajaviy farqlar – har bir til darajasi (masalan, rasmiy yoki norasmiy matnlar) uchun mos imlo variantlari belgilanadi;

dialektal va jargon so‘zlar – mahalliy yoki jargon so‘zlar standart imloda qanday yozilishi kerakligi aniqlanadi;

elektron qidiruv uchun optimallashtirish – so‘zlar variantlari (masalan, "qilmoq" / "qilish") bir xil yozuvga keltiriladi.

NATIJA VA MUHOKAMA

Negizlash (lemmalash) bosqichida har bir so‘zning lug‘aviy shakli (lemma), ya’ni lug‘atlarda uchraydigan asosiy shakli aniqlanadi. O‘zbek tilida negizlash - so‘zni morfologik tahlil qilish orqali uning asosiga (lug‘at shakliga / negiziga) qisqartirishni o‘z ichiga oladi. O‘zbek tili agglyutinativ til bo‘lgani uchun, lemma sifatida odatda hech qanday grammatik qo‘shimcha olmaydigan sof lug‘aviy shakl tanlanadi (masalan, “boryapti” so‘zining lemma shakli “bor”). Har bir so‘zni asosiy yoki lug‘at shakligacha qisqartirish jarayoni (jumlalarni negizlash / lemmalash) ko‘plab tabiiy tillarni qayta ishlash vositalarida asosiy jarayondir. O‘zbek tili aglyutinativ tillar oilasiga mansub bo‘lib, ko‘pincha zamon, mayl, shaxs va holni bildirish uchun qo‘shimchalardan foydalanadi. Shuning uchun o‘zbek tilidagi negizlash qo‘shimchalarni olib tashlashni talab qiladi. Matnli korpusni shakllantirish maqsadida negizlash (lemmalash) jarayonida quyidagicha misollarni uchratish mumkin:

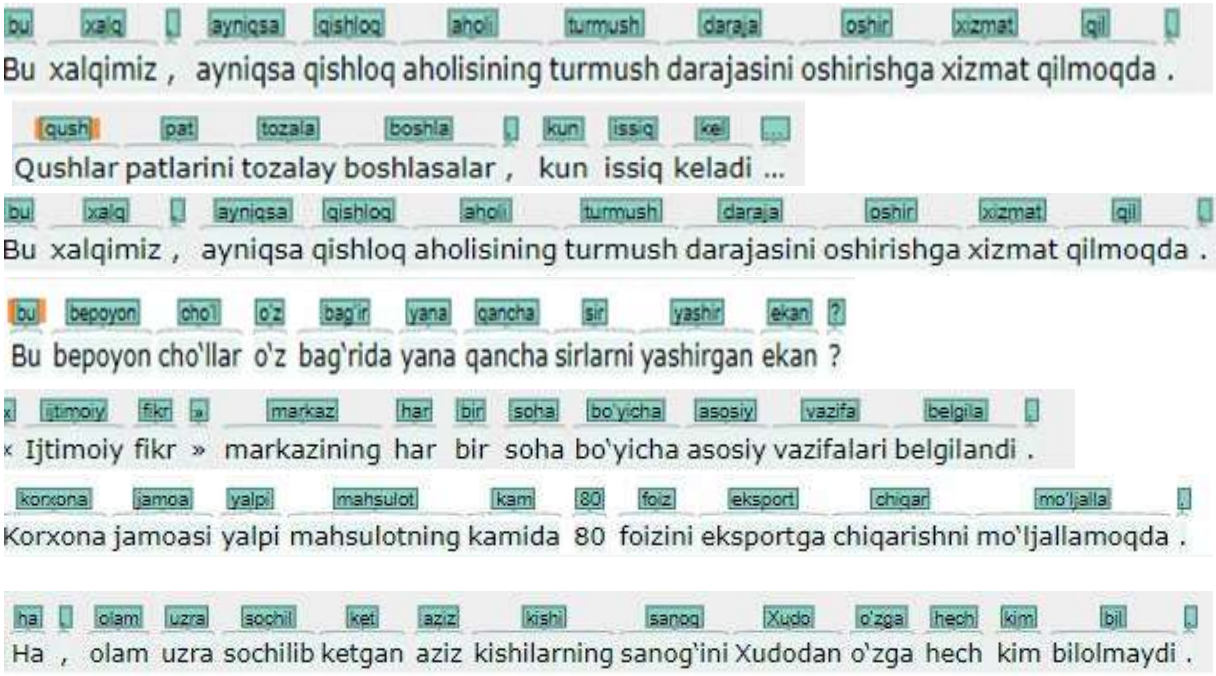
Gap: *Men kitob o‘qiyapman va ukam musiqa tinglayapti*. (*Lemmalar:* Men, kitob, o‘qi, va, uka, musiqa, tingla);

Gap: U dars tayyorlayapti, chunki ertaga imtihon bor. (*Lemmalar*: U, dars, tayyorla, chunki, ertaga, imtihon, bor);

Gap: Agar ob-havo yaxshi bo‘lsa, biz sayrga chiqamiz. (*Lemmalar:* Agar, ob-havo, yaxshi, bo‘l, biz, sayr, chiq).

Tadqiqot ishida ishlab chiqilgan o‘zbek tili iyerarxik korpusida jumlalarni negizlash amalga oshirilgan quyidagi misollarni keltirish mumkin:

TILSHUNOSLIK



2-rasm. O‘zbek tili iyerarxik korpusida gapdagi so‘zlarning negizlarini (lemmalarini) belgilash.

So‘zlarning asos qismlari shu shaklda olindi va tizimda avtomatik tarzda lemmalash amalga oshirildi. Gapdagi har bir so‘z, tinish belgilari alohida lemma qilib olindi.

Lemmalash qoidalarini qo‘llash: korpusda so‘zning asosiy shaklini topish uchun lemmalash funksiyasidan foydalaniladi, bunda kirish sifatida so‘z va uning natijasi sifatida POS tegi chiqariladi.

Misol uchun, agar *Python*dan foydalanilganda, NLTK-dagi WordNetLemmatizer funksiyasi so‘zlarni POS teglari asosida lemmalashi mumkin:

```
from nltk.stem import WordNetLemmatizer
from nltk.corpus import wordnet
lemmatizer = WordNetLemmatizer()
def get_wordnet_pos(word):
tag = nltk.pos_tag([word])[0][1][0].upper()
tag_dict = {"J": wordnet.ADJ, “N": wordnet.NOUN, “V": wordnet.VERB, “R": wordnet.ADV} return tag_dict.get(tag, wordnet.NOUN)
lemmatizer.lemmatize ("yugurish", get_wordnet_pos ("yugur"))
```

Lemmalarni asl so‘zlarga xaritalash. Har bir asl so‘zni lemmasi bilan almashtiriladi. Misol uchun, “yugurish”ning lemmalashtirilgan shakli “yugur” bilan almashtiriladi.

Annotatsiya jarayonini samarali tashkil etish uchun, avvalo, barcha jumlalar tokenlarga ajratildi (ya’ni har bir so‘z alohida satrga chiqarildi). Bu bosqichda mavjud tayyor vositadan – NLTK tokenizatori yordamidan foydalanildi [6]. NLTK (Natural Language Toolkit) – Python dasturlash tilidagi tilni qayta ishlash kutubxonasi bo‘lib, undagi standart tokenizator satrni bo‘sh joy va tinish belgilari bo‘yicha so‘zlarga ajratadi. NLTK yordamida 1200 ta jumla 9800 ta tokenga ajratildi, ya’ni korpus umumiy hajmi 9800 so‘zni tashkil etdi. Tokenlash jarayonida quyidagi qoidalar va maxsus holatlar inobatga olindi:

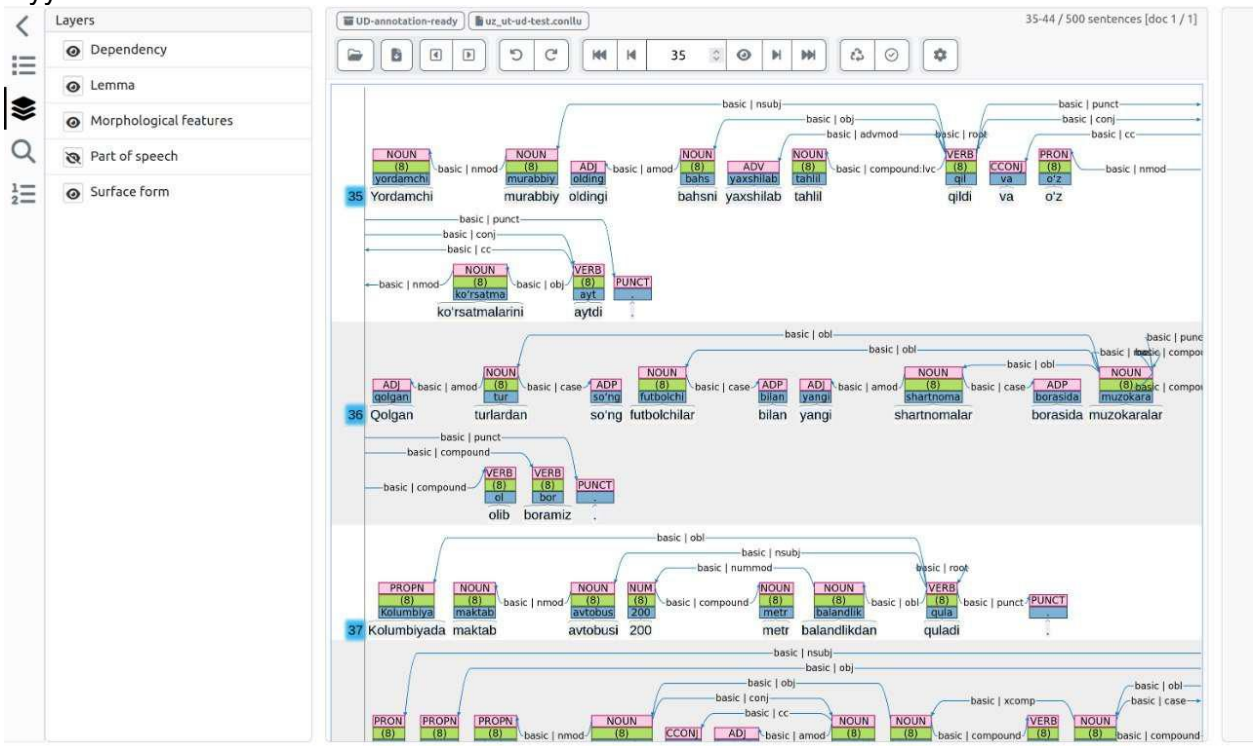
Oddiy holatda, gapdagi har bir so‘z yoki birikma bo‘shliq bilan ajralgan bo‘lsa, alohida token sifatida olindi. Masalan, “Talaba kitob o‘qidi” jumlasida uchta token – “Talaba”, “kitob”, “o‘qidi” ajratiladi.

Tinish belgilaridan so‘ng ham token chegarasi qo‘yildi. Masalan, vergul, nuqta, so‘roq-belgisi kabi tinish belgilari alohida token sifatida belgilandi. “Keldi, ko‘rdi, g‘alaba qozondi.” kabi gapda vergul va nuqta ham alohida token bo‘ladi.

Keltirib chiqarilmagan qo‘shimchalar: Ba’zi tinish belgilari so‘zga qo‘shilib yozilgan bo‘lsa, ularni ajratish kerak bo‘ladi. Masalan, “kitobni.” oxirida nuqta bor – tokenlash jarayonida “kitobni” va “.” alohida tokenlar sifatida olinadi.

Qisqartmalar: matnda nuqta bilan tugaydigan qisqartma so‘zlar bo‘lsa, ular ichidagi nuqta tokenlash vaqtida ajratib yuborilmasdan, so‘zning tarkibida saqlab qolindi. Chunki UD formatida nuqta abbreviatura tarkibida kelsa, u so‘zning bir qismi deb qaraladi. Masalan, “mln.” (million qisqartmasi) bir butun token sifatida olindi. Xuddi shuningdek, “A. Navoiy” kabi ism familiya qisqartma holatida – “A.” alohida token (nuqta bilan birga) va “Navoiy” alohida token tarzida ko‘rib chiqildi.

Ko‘p so‘zli birikmalar (multi-word expressions): UD qoidalariga ko‘ra, token ichida bo‘shliq bo‘lishiga yo‘l qo‘yilmaydi, ya’ni ikki yoki undan ortiq so‘zdan tashkil topgan birikma bitta token sifatida birlashtirilmaydi. Buning o‘rniga, agar mazmunan yaxlit bo‘lgan birikma bir necha so‘zdan iborat bo‘lsa, ular alohida tokenlar sifatida qoldiriladi va keyinchalik maxsus sintaktik bog‘lanish bilan o‘zaro bog‘lanadi. Masalan, “Tog‘li Qorabog‘” geonomik nomi ikki so‘zdan iborat bo‘lsa ham, UD tizimida u *flat* munosabat bilan bog‘langan ikkita alohida token sifatida beriladi. Ya’ni, “Tog‘li” va “Qorabog‘” so‘zlari har biri token, va annotatsiyada ular *flat* teg orqali birikkan bir nom sifatida tasvirlanadi. Bunday yondashuv, ayniqsa, ism-familya, murakkab joy nomlari, birikma fe’llar va so‘z birikmalarini belgilashda qo‘llanildi. Natijada, tokenlash qoidalari til xususiyatlariga mos holda ishlab chiqilib, barcha jumlalar birxil mezonda so‘zlarga ajratildi va keyingi annotatsiya bosqichiga tayyorlandi.



3-rasm. INCEpTION dasturida O‘zbek tili iyerarxik korpusi uchun annotatsiya jarayoni bosqichlari va tokenlash qoidalarining qo‘llanilishi: chap tomonda matn bo‘laklarga (tokenlarga) ajratilib, har biriga ID raqami berilgan; o‘ng tomonda esa, shu tokenlarga tegishli morfologik teglar va sintaktik bog‘lanishlar ko‘rsatilgan (bog‘lovchi o‘qlar bilan).

Bosqichma-bosqich annotatsiyalash jarayoni. Korpus uchun tokenlar tayyor bo‘lgach, har bir jumlani to‘liq belgilash bir necha bosqichda amalga oshirildi.

Quyida annotatsiyalashning asosiy bosqichlari ketma-ketligi va har bosqichda bajarilgan ishlarning qisqacha bayonini keltiramiz:

Lemmalash (asosiy shaklni aniqlash): har bir token uchun uning lug‘aviy asosi (lemma) avtomatik tarzda aniqlab chiqildi. Buning uchun mavjud dasturiy vositalardan foydalanildi – xususan, o‘zbek tilining UzbekLemmatizer modelidan foydalanildi. Avtomatik lemmatizator yordamida har bir so‘zning boshlang‘ich shakli (masalan, fe’lning infinitiv yoki otning birlik-bosh kelishikdagi shakli) belgilanib, annotatsiya tizimiga kiritildi. Avtomatik jarayon taxminan 85-90% aniqlik bilan lemmalarni to‘g‘ri topib berdi, biroq ayrim hollarda xatolar kuzatildi – masalan, ma’nosi bir-biridan farq qiluvchi, lekin yozilishi bir xil bo‘lgan so‘zlarda (omonimlarda) lemmalash jarayonida noto‘g‘ri

tanlov qilishi kuzatildi. Shu sababli, keyingi bosqichlarda annotatorlar har bir so‘zning lemmalarini yana qo‘lda tekshirib chiqdilar va zarur joylarda to‘g‘rilashdi. Masalan, “och” so‘zi gapga qarab “ochmoq” (yopiq narsani ochish) yoki “och” (to‘qlikning antonimi) lemmasiga ega bo‘lishi mumkin – bunday noaniq holatlarda tegishli lemmani tanlash annotatorning qaroriga havola qilindi.

Morfologik teglash (so‘z turkumlari va xususiyatlar): keyingi qadam – har bir so‘zga uning grammatik so‘z turkumi va batafsil morfologik xususiyatlarini teg tarzida belgilashdir. Bu jarayon qisman avtomatik, qisman qo‘lda bajarildi. Avvalo, maxsus UzMorphAnalyzer dasturi (Ulug‘bek Salaev, 2023) yordamida har bir so‘zning barcha mumkin bo‘lgan grammatik tavsiflari ishlab chiqildi [4]. UzMorphAnalyzer o‘zbek tilidagi so‘zlarni tahlil qilib, ularning so‘z turkumini (masalan, ot, fe’l, sifat va h.k.) hamda shaxs-son, zamon, kelishik kabi grammatik kategoriyalarini aniqlaydi. Ushbu dastur natijalari asosida dastlabki Universal POS (UPOS) teglari avtomatik tarzda berildi: masalan, UzMorphAnalyzer natijasida “kitobni” so‘zi uchun NOUN (ot) deb belgilandi, “o‘qidi” so‘zi uchun VERB (fe’l) va hokazo. Biroq avtomatik belgilash to‘liq ishonchli chiqmagani bois, barcha UPOS teglar 5 nafar annotator tomonidan birma-bir tekshirilib, zarur tuzatishlar kiritildi. Shu jarayonda an’anaviy o‘zbek tili grammatikasidagi toifalar bilan UD standartidagi 17 ta so‘z turkumi orasidagi mapping (mos kelish) jadvali ham tuzildi. Masalan, an’anaviy grammatikada “ravishdosh” deb ataluvchi turkum UD tizimida VERB (fe’l) sifatida belgilandi va unga qo‘shimcha VerbForm=Conv xususiyati berildi.

TADQIQOT MUHOKAMASI.

Morfologik xususiyatlar (FEATS) uchun UD tomonidan belgilangan 42 xil universal xususiyatlardan mos keladiganlari tanlandi. Bular qatoriga Kelishik (Case), Son (Number), Egalik olmoshlari (Possessive), Daraja (Degree), Zamon (Tense), Holat (Mood), Shaxs (Person), Niqob (Polarity) va Daraja (Voice) kabi turli kategoriyalar kiradi. Har bir so‘zga uning gapdagi shakliga mos keluvchi xususiyatlar to‘plami biriktirildi. Masalan, “kitobni” so‘ziga Case=Acc (tushum kelishigi) va Number=Sing (birlik) xususiyatlari, “o‘qidi” fe’liga esa Tense=Past (o‘tgan zamon), Person=3, Number=Sing va Polarity=Pos kabi belgi qiymatlari qo‘yildi. Boshida 150 ta jumla shu tarzda to‘liq qo‘lda morfologik teglab chiqildi va Stanza (tabiiy til tahlili Python paketi) vositasi [3]. yordamida mashinani o‘qitish orqali qolgan jumlalarga avtomatik teg qo‘yishga harakat qilindi. Stanford Stanza – bu turli tillar uchun modelni o‘qitib, analiz qiluvchi zamonaviy kutubxona (Qi va boshq., 2020) [2,15]. Biz 150 ta qo‘lda teglangan jumlalarni Stanza modeliga o‘rgatib, boshqa jumlalar uchun so‘z turkumlari va xususiyatlarini avtomatik topish amaliyotini ham bajardik. Natijada ba’zi oddiy kategoriyalar (masalan, Number yoki Case singari) deyarli xatosiz topildi, murakkabroq joylarda esa xatolar kuzatildi. Shuning uchun yakunda barcha 1200 ta jumlaning morfologik teglari yana qo‘lda ko‘rib chiqildi va avtomatik bosqichda yuzaga kelgan xatolar tuzatildi. Bu bosqich yakunida korpusning har bir so‘zi uchun to‘g‘ri LEMMA, UPOS va FEATS to‘plami aniqlandi.

Sintaktik bog‘lanishlarni belgilash (iyerarxik tahlil): korpus yaratish jarayonidagi eng muhim bosqich – bu gapdagi so‘zlar orasidagi sintaktik munosabatlarni, ya’ni iyerarxik bog‘lanishlarni belgilashdir. Universal Dependencies doirasida har bir bog‘lanish HEAD (bosh so‘z) va DEPREL (bog‘lanish turi) parametrlari bilan ifodalanadi.

XULOSA

Ushbu maqolada korpusimizda UD tavsiya etgan 37 ta asosiy sintaktik munosabatlardan 32 tasini qo‘llangani aks etadi (qolganlari bizning kiritgan jumlalarimizda uchramadi). Sintaktik bog‘lanishlarni belgilash qo‘l mehnatini talab qiluvchi, murakkab vazifa bo‘lgani bois, uni ham qisman avtomatlashtirishga harakat qildik. Dastlab, yuqorida morfologik teglar bilan boyitilgan 150 ta jumlaga to‘g‘ridan-to‘g‘ri qo‘lda sintaktik bog‘lanishlar qo‘yildi. Bunda maxsus grafik vositalardan – Grew (Graph Rewriting tool, Guillaume, 2021) [1,168]. – foydalandik; Grew yordamida UD qoidalariga zid kelmaydigan daraxt tuzilmasini tekshirib borish mumkin. Shuningdek, sintaktik tahlil modelini mashinada o‘qitishda yordam berish uchun fastText dan olingan o‘zbekcha so‘z vektorlaridan foydalanildi. 150 ta jumla qo‘lda bog‘lanib bo‘lgach, Stanza tarkibidagi parser modelini shu ma’lumotlar asosida dastlabki o‘qitdik. Ushbu dastlabki model keyin yana bir qancha jumlani avtomatik tahlil qilib berdi, ya’ni har bir so‘zga HEAD va DEPREL qiymatlarini belgilab chiqdi. Albatta, bu avtomatik natijalar to‘liq ishonchli bo‘lmagan – annotatorlar tomonidan ushbu jumlalarning har biri diqqat bilan tekshirilib, avtomatik tahlil xatolari tuzatildi. Tuzatilgan

jumlalar qo‘shimcha tarzda dasturga qayta o‘rgatilib, sintaktik analizning ikkinchi modeli hosil qilindi. Nihoyat, qolgan jumlalar ham shu ikkinchi model yordamida avtomatik tahlil qilindi va ular ham mutaxassis tomonidan qo‘lda tekshirilib, zaruriy joylarda tuzatildi. Jarayon yakunida sintaktik bog‘lanishlarni belgilovchi model uchinchi bor takomillashtirilib o‘qitildi (1-bosqichdan 3-bosqichgacha model sifat ko‘rsatkichlari oshib bordi — F1 o‘lchovi birinchi yurgizishda 46%, oxirgi yurgizishda 52% gacha ko‘tarildi). Oxir-oqibat, korpusdagi barcha jumlalarning izchil sintaktik daraxtlari (bog‘lanishlari) tayyor bo‘ldi.

Annotatsiya jarayonining ushbu bosqichma-bosqich yondashuvi jamoaga ishni ma’lum darajada avtomatlashtirib, vaqt va kuchni tejashga imkon berdi. Yakuniy bosqichda esa barcha qatlamlarning mutanosibligini yana bir bor sinchiklab tekshirish, yagona formatga keltirish va CoNLL-U fayllariga eksport qilish amalga oshirildi.

ADABIYOTLAR RO‘YXATI

1. Bruno Guillaume. 2021. Graph Matching and Graph Rewriting: GREW tools for corpus exploration, maintenance and conversion. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 168–175, Online. Association for Computational Linguistics.

2. P Qi, Y Zhang, Y Zhang, J Bolton, Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. CD Manning. Association of Computational Linguistics (ACL) System Demonstrations.

3. https://stanfordnlp.github.io/stanza/

4. Salaev U. UzMorphAnalyser: A morphological analysis model for the Uzbek language using inflectional endings //AIP Conference Proceedings. – AIP Publishing, 2024. – T. 3244. – №. 1

5. Stefanie Dipper, Cora Haiber, Anna Maria Schröter, Alexandra Wiemann, and Maike Brinkschulte. 2024. Universal Dependencies: Extensions for Modern and Historical German. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17101–17111, Torino, Italia. ELRA and ICCL.

6. https://www.nltk.org/api/nltk.tokenize.html

7. https://huggingface.co/datasets/tahrirchi/uz-crawl