IEEE
TÜRKİYE SECTION

UBMK'25

Bildiriler Kitabı
Proceedings

Editör Eşref ADALI

10. Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı

10th International Conference on Computer Science and Engineering

17-18-19 Eylül (September) 2025 İstanbul - Türkiye

IEEE TÜRKİYE SECTION

UBMK'25

Bildiriler Kitabı
Proceedings

Editor Eşref ADALI

10. Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı

10th International Conference on Computer Science and Engineering

17-18-19 Eylül (September) 2025 İstanbul - Türkiye

Media type	Part Number	ISBN	Online ISSN
XPLORE COMPLIANT	CFP25L97-ART	979-8-3315-9975-1	2521-1641
CD-ROM	CFP25L97-CDR	979-8-3315-9974-4	

10. Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı (UBMK'2025)

10th International Conference on Computer Science and Engineering

17-18-19 Eylül 2025 -İstanbul-Türkiye
17-18-19 September 2025 - İstanbul-Türkiye

IEEE Catalog Number : CFP25L97-ART

ISBN : 978-8-3315-9975-1

Additional copies mey be ordered from:
Curran Associates, Inc
57 Morehouse Lane Red Hook, NY 12571 USA
Phone: (845) 758 0400
Fax: (845) 758 2633
E-mail: curran@proceeding.com

UBMK’2025’ye Hoşgeldiniz
Welcome to UBMK’2025

Sevgili Katılımcılar:

UBMK uluslararası nitelikli konferans serisi, 1990 yılından beri düzenli olarak yapılmakta olan Bilgisayar Mühendisliği Bölüm Başkanları toplantılarında alınan bir kararla on yıl önce başlamıştır. Konferansın 10.su IEEE-UBMK-2025 bu yıl 17-18-19 Eylül, 2025 günlerinde İstanbul Teknik Üniversitesinin ev sahipliğinde düzenlemiştir.

IEEE-UBMK-2025 konferansına bu yıl Almanya, Amerika Birleşik Devletleri, Azerbaycan, Fransa, Irak, İngiltere, İsveç, İtalya, Kanada, Kazakistan, Kırım, Kırgızistan, Rusya, Özbekistan, Tataristan, Taylant, Ürdün ve Türkiye’den 610 dolayında bildiri gönderilmiş ve bu bildiriler Türk ve yabancı 250 hakem tarafından değerlendirilmiştir.

Her bildiri en az iki hakem tarafından incelenmiş ve uzlaşma olmadığı durumlarda üçüncü bir hakemin değerlendirmesine başvurulmuştur. Bildiri başına düşen ortalama hakemlik 2,3 olmuştur. Bu değerlendirmelerin sonunda 327 bildirinin sözlü olarak sunulması uygun bulunmuştur. Kabul edilen ve sunulan bildiriler içerik ve kalite ölçünlerini sağlaması durumunda IEEE Xplore’da yayımlanacaktır.

Konferans çalışmalarında, Bilgisayar Mühendisliği Bölüm Başkanları Danışma Kurulu olarak görev almışlardır. Bildirilerin değerlendirilmesi Bilim Kurulu üyeleri tarafından yapılmıştır. Konferansın düzenlenmesi ise Yürütme Kurulunun önerileri doğrultusunda, Düzenleme Kurulu tarafından yapılmıştır.

Son olarak, konferansın başarılı bir şekilde yürütülmesi için tüm olanaklarını sunan İstanbul Teknik Üniversitesi Rektörü Sayın Prof. Dr. Hasan Mandal’a teşekkür ediyoruz. Ayrıca Düzenleme Kuruluna, bildirileri titizlikle değerlendiren Bilim Kurulu Üyelerine ve değerli araştırmalarının sonuçlarını bilişim camiası ile paylaşan bildiri sahiplerine teşekkürlerimizi iletiriz.

Prof. Dr. Eşref ADALI
UBMK-2025 Konferans Başkanı ve Bildiri Kitabı Editörü

Dear Participants:

The UBMK international conference series started nine years ago with a decision taken at the Computer Engineering Department Heads (BMBB) meetings, which have been held regularly since 1990. The 10th edition of the conference, UBMK'25, was held this year on October 17-18-19, 2025, hosted by İstanbul Technichal University.

This year, approximately 610 papers were submitted to the IEEE-UBMK-2025 conference from Germany, the United States, Azerbaijan, France, Iraq, the United Kingdom, Sweden, Italy, Canada, Kazakhstan, Crimea, Kyrgyzstan, Russia, Uzbekistan, Tatarstan, Thailand, Jordan, and Turkey, and these papers were evaluated by 250 Turkish and foreign referees.

Each paper was evaluated at least by two referees, and in cases where there was no consensus, a third referee was consulted. At the end of these evaluations, 327 papers were accepted for oral presentation. Accepted and presented papers will be submitted for inclusion into IEEE Xplore subject to meeting IEEE Xplore's scope and quality requirements.

During the conference, Heads of Information Engineering Departments took part in the Advisory Board. The evaluation of the papers was made by the members of the Scientific Committee. The conference was organized by the Organizing Committee in line with the recommendations of the Executive Committee.

Finally, we would like to thank İstanbul Technical University Rector Prof. Dr. Hasan Mandal for his continued support for the success of the conference. In addition, we would like to thank the Organizing Committee, the Scientific Committee Members who carefully evaluated the papers, and the owners of the papers who shared the results of their valuable research with the informatics community.

Prof. Dr. Esref ADALI
UBMK'25 Conference Chair and Proceedings Editor

Düzenleyenler / Organizer

Destekleyenler / Sponsors

Morphotactic Models and Algorithms of the Uzbek Language

Khamroeva Shahlo Mirdjonovna
Dept.of Computational Linguistics and Digital Technologies
Tashkent State University of Uzbek Language and Literature named Alisher Navo'i
Tashkent, Uzbekistan
shaxlo.xamrayeva@navoiy-uni.uz

Elov Botir Boltayevich
D*ept.of Computational Linguistics and Digital Technologies*
Tashkent State University of Uzbek Language and Literature named Alisher Navo'i
Tashkent, Uzbekistan
elov@navoiy-uni.uz

Alaev Ruhillo Habibovich
Department of information Security
National University of Uzbekistan named after Mirzo Ulugbek
Tashkent, Uzbekistan
alayev_r@nuu.uz

Saydullayeva Dilbar Farhod qizi
Department of Uzbek Linguistics
National university of Uzbekistan,
Tashkent, Uzbekistan
dilbarsaydullayeva069@gmail.com

***Abstract*—This article presents a comprehensive study of morphotactic models and algorithms developed for the Uzbek language. Given the agglutinative nature of Uzbek, morphotactics—rules governing the internal structure and sequence of morphemes within a word—play a crucial role in accurate morphological analysis and generation. The paper outlines rule-based and data-driven approaches to modeling Uzbek morphotactics, emphasizing the integration of grammatical constraints and affix ordering principles. It further discusses the implementation of finite-state transducers (FSTs) and machine learning techniques for automatic morphological tagging and generation. The proposed models aim to enhance the performance of natural language processing (NLP) applications such as part-of-speech tagging, lemmatization, and syntactic parsing for Uzbek. Evaluation results demonstrate that the developed algorithms achieve high accuracy in capturing the complex morphological patterns of the language. This research contributes to the advancement of computational resources for low-resourced Turkic languages.**

Keywords—Universal Dependencies; morphological analysis; morphotactic model; FEATS mapping; deterministic finite automaton; lemmatization; affixal segmentation; parts of speech; UD tagging; UD format; morphological tagging algorithm; grammar formalism; agglutinative language; natural language processing (NLP).

I. INTRODUCTION

Morphotactic modelling constitutes a pivotal component of contemporary research on morphologically rich, agglutinative languages, where the combinatorial range of affixes demands precise formalisation to guarantee reliable computational processing. Uzbek—an Eastern Turkic language spoken by more than 35 million people—exemplifies such complexity: its words are generated through the sequential concatenation of derivational and inflectional morphemes whose ordering is governed by rigid grammatical constraints. A faithful computational representation of these constraints is indispensable for a broad spectrum of downstream natural-language-processing (NLP) tasks, including lemmatisation, syntactic parsing, information retrieval, and machine translation. Despite substantial advances in morphotactic modelling for better-resourced languages, Uzbek remains comparatively under-represented in the computational-linguistics literature. Existing resources are fragmented, often limited to small-scale rule sets or ad-hoc data-driven heuristics, which restrict their portability and scalability. The absence of a unified, linguistically grounded morphotactic framework hinders both the accuracy of current Uzbek NLP pipelines and their integration into multilingual systems.

This article seeks to address that lacuna by presenting a rigorous account of Uzbek morphotactics and by proposing algorithms that operationalise these insights within a computationally efficient architecture. We pursue two complementary lines of inquiry. First, we formalise Uzbek affix ordering through finite-state transducer (FST) technology, which encodes morphotactic constraints as deterministic state transitions and thus supports both exhaustive morphological analysis and robust generation. Secondly, we explore data-driven refinement strategies—most notably conditional random fields (CRFs) and neural sequence models—to capture residual variation and to disambiguate analyses in context. The synergy between rule-based precision and statistical disambiguation yields a hybrid system capable of achieving state-of-the-art accuracy while preserving linguistic interpretability. By providing an extensible FST grammar, publicly available training corpora, and reproducible evaluation benchmarks, the present study delivers both theoretical and practical contributions. Theoretically, it delineates the morphotactic architecture of Uzbek with greater explicitness than has previously been attempted; practically, it furnishes the NLP community with modular resources that can be seamlessly incorporated into existing toolchains. In doing so, the work advances the broader agenda of rendering low-resource Turkic languages first-class citizens in multilingual language-technology ecosystems.

II. LITERATURE REVIEW

The study of morphotactics in natural language processing has received significant attention, particularly in the context of agglutinative languages, where complex morphological structures necessitate robust formal representations. Foundational work in the field of computational morphology, such as [5] introduction of two-level morphology and [6] development of finite-state transducer (FST) architectures, has laid the groundwork for morphotactic modelling in numerous languages. These approaches have been effectively applied to Turkish, Finnish, and Hungarian, where morphological complexity is analogous to that of Uzbek. The success of these systems has demonstrated the efficacy of FSTs in encoding morphotactic constraints while enabling efficient parsing and generation. In the Turkic language family, which includes Uzbek, Turkish has been the primary focus of morphotactic research. K.Oflazer [2] introduced a comprehensive two-level morphological analyzer for Turkish using finite-state methods, later refined with more sophisticated rule-based and statistical approaches [1]. Azerbaijani and Kazakh have also begun to attract attention in recent years, particularly through the Universal Dependencies (UD) project and related annotation efforts. However, morphotactic modelling for Uzbek remains limited in both depth and scope. Existing studies on Uzbek morphology are primarily descriptive or pedagogical, focusing on the grammatical rules of suffixation, parts of speech, and morphological categories such as tense, aspect, mood, and person-number agreement. Some efforts have been made to build computational tools, including part-of-speech taggers and stemmers, often employing rule-based techniques or shallow machine learning algorithms. Notable among these is the work of [4], who developed a basic Uzbek morphological analyzer using rule-based FSTs. However, these early systems suffer from limited lexical coverage and insufficient handling of morphotactic ambiguity, particularly in derivational morphology and clitic attachment.

Recent advances in low-resource language modelling have also opened possibilities for Uzbek through the use of neural approaches, such as encoder-decoder architectures and sequence labelling models. While these techniques show promise, they are often constrained by the scarcity of large, annotated Uzbek corpora. Initiatives such as the Uzbek UD Treebank [8] provide a foundational resource, but further expansion is necessary to support robust model training and evaluation. Furthermore, morphotactic modelling for Uzbek has not yet fully benefitted from hybrid approaches that combine the rule-governed precision of finite-state models with the adaptive capabilities of machine learning. Such integrative frameworks have shown promising results in other morphologically complex languages and are well-suited to address the specific challenges posed by Uzbek, including vowel harmony, affix ordering, and the interaction of inflectional and derivational morphemes. A 2021 study proposes a finite state machine-based morphological analyzer specifically designed to address the complexities of the agglutinative structure of the Uzbek language, offering a systematic and formal approach to morphological processing [4]. Sharipov and Salaev [7], in their work, specifically detail a morphological analyzer for Uzbek using a finite state machine, focusing on an affix-stripping approach without relying on a lexicon. This contrasts with other discussions at Turklang, such as the comprehensive overview by Washington et al. [8] who, while presenting open-source morphological transducers for numerous Turkic languages (including Uzbek), emphasize hand-crafted finite-state transducers for accuracy and completeness, aiming to avoid under- and over-generation of forms often seen in statistical or affix-stripping methods.

While the theoretical basis for Uzbek morphotactics is well-documented in traditional linguistic literature, its computational implementation remains underdeveloped. There is a clear need for comprehensive, scalable models that integrate linguistic knowledge with data-driven refinement. The present study seeks to fill this gap by developing formal morphotactic algorithms tailored to the Uzbek language, informed by both linguistic theory and current trends in computational morphology.

III. MORPHOTACTIC MODELS

A. Morphotactic Mathematical Models

Based on the strict morphotactic skeleton of each part of speech in the Uzbek language presented above (Table 2), the set of deterministic finite automaton (DFA) graphs (Table 3), and the morphological features according to the Universal Dependencies (UD) standard (FEATS, Table 4), we develop morphotactic mathematical models for each part of speech in the form of formal grammar + deterministic finite automaton.

Noun

The noun paradigm describes the language of word forms that generate the following sequence:

$$\mathcal{L} = \{x = r y_1 y_2 y_3 y_4 y_5 y_6 y_7 \mid r \in R, y_i \in \Gamma_i, (y_1 \dots y_7)\ slot\ tartibida\ keladi\}$$

Examples are given in Table I.

TABLE I. NOUN

i-slot	Lebel	Set of affixes (Γ_i)	Note
0	**R**	**lemmas** (root of noun)	lug‘at
1	**OY**	*-lik, -chi, -zor, -xona, …*	Noun forming
2	**DIM**	*-cha, -gina, -jon, …*	diminutivization
3	**PL**	*-lar/-lar-i*	plural
4	**POS**	*-im, -ing, -i, -miz, …*	possesing
5	**CASE**	*Ø, -ni, -ga, -da, -dan, -ning, -gacha, -cha*	case
6	**REL**	*-day/-dek, -sifatida, …*	relationship and interaction
7	**PART**	*-ku, -mi, -chi, -da, -a/-ya*	postposition

For the noun part of speech, slots 1, 2, 6, and 7 are optional; slots 3 to 5 are grammatical; the order is fixed.

$$Ot = R(OY)?\,(DIM)?\,(PL)?\,(POS)?\,(CASE)?\,(REL)?\,(PART)?$$

For example, the word “kitoblarimizdan”:

R=“kitob”, PL=”-lar” , POS=”-imiz” , CASE=”-dan”

The morphological tagging model of the Uzbek language is represented in the form of a deterministic finite automaton (DFA) as follows:

$$M = \{Q, \Sigma, \delta, q_0, F\}$$

Here:

- $\boldsymbol{Q}$ – the set of states of the automaton.
- $\boldsymbol{\Sigma}$ – the alphabet (the set of all morphological units – roots and affixes – involved in word formation in Uzbek).
- $\boldsymbol{\delta}$ – the transition function between states:
- $\delta: Q \times \Sigma \to Q$
- $\boldsymbol{q_0}$ – the initial state (root or lemma).
- $\boldsymbol{F \subseteq Q}$ – the set of final (accepting) states (states where the analysis is successfully completed).

The morphological model consists of the following components:

Word root (lemma):

$$Asos = \{lemma \mid lemma \in Lug'at\}$$

Set of affixes (Σ):

Affixes in the Uzbek language are divided into the following groups:

$$\Sigma = Affikslar = Kelishik \cup Ko'plik \cup Egalik \cup Nisbat \cup Inkor \cup Zamon \cup Mayl \cup ShaxsSon \cup RavishYasovchi \cup Daraja \cup Yuklama$$

Each group of affixes is denoted as follows (Table II):

TABLE II. EACH GROUP OF AFFIXES

Affix Group	Notation	Elements (Examples)
Case	Case	{−ni,−ning,−ga,−da,−dan,−gacha,Ø}
Plural	Plural	{−lar}
Possessive	Poss	{−im,−ing,−i,−miz,−ngiz,−lari}
Voice	Voice	{−il,−in,−ish,−tir}
Negation	Polarity	{−ma,−mas,−may}
Tense	Tense	{−di,−gan,−moqda,−yapti,−ar,−adi,−ay otgan}
Mood	Mood	{−sa,−moqchi,Ø}
Person-number	Person/ Number	{−man,−san,−miz,−siz,−lar,Ø}
Adverbializer	AdvDeriv	{−cha,−lab,−larcha}
Degree (in adjectives and adverbs	Degree	{−roq,Ø}
Postposition	Part	{−mi,−gina,−ku}

The set of automaton states (Q) and the initial state ($q0$), where the initial state of the automaton corresponds to the stem (lemma):

$q0=Asos$

During the generation (or analysis) of a word form, the automaton transitions to the next states based on the following morphotactic rules. Each state indicates the type of the next grammatical affix of the word. The set of states is represented as follows:

$$Q = \{Asos, Ko'plik, Egalik, Kelishik, Nisbat, Inkor, Zamon, Mayl, ShaxsSon, RavishYasovchi, Daraja, Yuklama, FINAL\}$$

The set of final states: $F = \{FINAL\}$

The state transition function (δ) is based on morphotactic rules. Below, the transitions for each part of speech are presented.

B. Mathematical operation principle of the model

A word w ($w \in \Sigma$) is accepted by the automaton (i.e., morphologically correctly analyzed) if and only if the automaton, starting from the initial state $q0$, sequentially accepts all morphological elements of the word (the stem and affixes) and reaches a final state. Only in this case can the morphological analysis be considered correct. Formally: A word $w=a1a2...an$, $ai \in \Sigma$, is accepted by the automaton if the following conditions are met:

- $\delta(q_0, a_1) = q_1$
- $\delta(q_1, a_2) = q_2$

…

- $\delta(q_{n-1}, a_n) = q_n, \quad q_n \in F$

In other ways:

$$w \in L(M) \Leftrightarrow \delta(q_0, w) \in F$$

Here, $L(M)$ denotes the language of all words accepted by the automaton. The deterministic finite automaton for the noun part of speech is constructed as follows:

- **Condition**, $Q = \{q_0, q_{OY}, q_{DIM}, q_{PL}, q_{POS}, q_{CASE}, q_{REL}, q_{PART}, q_f\}$
- **Starter**, q_0 – after the lexical stem is accepted
- **Accepting state**, q_f
- **Alphabet,** $\Sigma = R \cup \bigcup_{i=1}^{7} \Gamma_i$
- **Transitions (ε –** epsilon transitions**)**

$$\delta(q_0, OY) = q_{OY} \mid Yoki\ OY\ bo`lmasa\ q_0 \to q_{DIM}$$
$$\delta(q_{OY}, DIM) = q_{DIM}$$
$$\delta(q_{DIM}, PL) = q_{PL}$$
$$\delta(q_{PL}, POS) = q_{POS}$$
$$\delta(q_{POS}, CASE) = q_{CASE}$$
$$\delta(q_{CASE}, REL) = q_{REL}$$
$$\delta(q_{REL}, PART) = q_{PART}$$
$$\delta(q_{PART}, \varepsilon) = q_f$$

For a word $\boldsymbol{w}$ in the noun form, the following chain is recorded in the automaton M:

$$\delta^*(q_0, w) = q_f$$

Thus, it can be proven that the language of words w conforming to the morphotactic rules is equal to L(M)=L.

Map of UD-FEATS

$\Phi: \Gamma_i \to UD\ Feats$

$\Phi(-lar) = Number = Plur,$

$\Phi(-imiz) = Person[psor] = 1 \mid Number[psor] = Plur,$

$\Phi(-ni) = Case = Acc,$

$\Phi(-mi) = token\ \ UPOS = PART, \ldots$

Resulting UD lebel – $\bigcup_i \Phi_i$

Use of the Model

1. The segmentation algorithm checks from the end of the word: $\Gamma_7 \to \Gamma_6$
2. If each separated affix $\boldsymbol{y_i}$ does not correspond to the slot → it is rejected.
3. If the remaining part $r \in R$ (the noun lemma in the dictionary), it is accepted; otherwise, it is marked as unknown or erroneous (Table III).

TABLE III. EXAMPLES

Word	Segmentation	State transitions	UD-FEATS
kitob-lar-imiz-dan	R=kitob, PL=−lar, POS=−$imiz$, $CASE$=−dan	$q_0 \rightarrow q_{PL}$ $\rightarrow q_{POS}$ $\rightarrow q_{CASE}$ $\rightarrow q_f$	Number=Plur | Person[psor]=1 | Number[psor]=Plur | Case=Abl
bola-cha-ngiz-ga	R=bola, DIM=−cha, POS=−$ngiz$, $CASE$=−ga	$q_0 \rightarrow q_{DIM}$ $\rightarrow q_{POS}$ $\rightarrow q_{CASE}$ $\rightarrow q_f$	Derivation=Diminutive | Person[psor]=2 | Number[psor]=Plur | Case=Dat

The model determines the affix-ordering rules of Uzbek nouns at the level of a regular language, and the finite automaton implementation serves as the core verifier of the Uzbek morphological analyzer.

Verb

The verbal form of the verb consists of the following strict sequential affix blocks (Table IV):

$$V=[r+VD+VO+NEG+TNS+AGR+MOD+PRT],$$

TABLE IV. VERB EXAMPLES

i-slot	Lebel	Set of affixes (Γ_i)	Examples
0	**R**	Root of verb (lemma)	*kel, bor, o‘qi*
1	**VD**	Verb forming	*-la* (*bo‘y-la*), *-ar* (*yash-ar*), *-ish* (reciprocal)
2	**VO**	Voice	*-dir* (caus), *-il* (pass), *-in* (refl), *-ish* (rcp)
3	**NEG**	Negative	*-ma/-mas/-may*
4	**TNS**	Tense	*-di* (Past), *-moqda / -yap* (Prog Pres), *-ar/-adi* (Fut/Hab)
5	**AGR**	Person-number	*-man, -san, -miz, -ngiz, -lar, Ø*
6	**MOD**	Mood	*-sa* (Cond), *-moqchi* (Prp), *-sin/-ing* (Imp), Ø (Ind)
7	**PRT**	Postposition	*-mi, -ku, -da, -a/-ya*

$$Fe'l = R(VD)?\,(VO)?\,(NEG)?\,(TNS)?\,(AGR)?\,(MOD)?\,(PRT)?$$

TABLE V. EXAMPLES TO THE PART OF SPEECH VERB

Word	Segmentation	State transitions	UD-FEATS
yoz-dir-ma-di	R=yoz, VO=−dir, NEG=−ma, TNS=−di	$q_0 \rightarrow q_{VO}$ $\rightarrow q_{NEG}$ $\rightarrow q_{TNS}$ $\rightarrow q_f$	Voice=Cau | Polarity=Neg | Tense=Past | Person=3 | Number=Sing
bor-ayotgan-miz	R=bor, TNS=−$ayotgan$, AGR=−miz	$q_0 \rightarrow q_{TNS}$ $\rightarrow q_{AGR}$ $\rightarrow q_f$	Aspect=Prog | Tense=Pres | Person=1 | Number=Plur
kel-ma-sin	R=kel, NEG=−ma, MOD=−sin	$q_0 \rightarrow q_{NEG}$ $\rightarrow q_{MOD}$ $\rightarrow q_f$	Polarity=Neg | Mood=Imp | Person=3 | Number=Sing

Similar to the noun, deterministic finite automaton, states, transition functions, accepting states, and UD feature functions are constructed for adjectives, numerals, adverbs, pronouns, auxiliaries, conjunctions, predicates, modal words, onomatopoeic words, and interjections.

The presented mathematical model (based on a DFA) enables the automatic morphological segmentation and grammatical tagging of words in the Uzbek language. This formal model fully represents the morphological structure of Uzbek from a mathematical standpoint and allows adaptation to the Universal Dependencies (UD) standard. The model can be practically applied in automatic tagging systems, machine translation, NLP, and other linguistic technologies development.

C. Morphological Segmentation and Analysis Algorithm

Based on the formal model described above, a practical algorithm has been developed. The algorithm segments the input word into its constituent parts and assigns the relevant grammatical information to each segment (segmentation and tagging). The operational stages of the algorithm are explained as follows:

Stage 1. *Input and Preparation:* Initially, the word to be analyzed is accepted as input. The word is pre-normalized: uppercase and lowercase letters are unified into a single form, and Uzbek Latin and Cyrillic alphabets are converted into a consistent alphabet (e.g., Latin). Characters such as apostrophes and hyphens are treated separately.

Stage 2. *Lexical Analysis (Dictionary-based):* A lexical check is performed to identify the possible stem of the word. If the word is fully present in the dictionary (for example, as an independent standalone word), it may not be complex. However, in agglutinative languages, often only the stem (lemma) of a word is stored in the dictionary rather than its full forms. Therefore, the algorithm considers multiple possible stem candidates for the word. For instance, for the input “*kitoblarimizdan*” the stem “*kitob*” might be found in the dictionary. In some cases, reliance on the dictionary might be insufficient. For example, to analyze unknown words, the algorithm must identify the stem directly by segmenting the affixes.

Stage 3. *Rule-based Recursive Analysis:* At this stage, segmentation is performed starting from the end of the word according to the affix rules of the formal model. The algorithm operates recursively or by iterative backward search:

```
funksiya Tahlil(so‘z):
  agar so‘z bo‘sh bo‘lsa:
  return [[]]  #bo‘sh segmentlar ro‘yxati (tahlil yakunlandi)
  natijalar = []
  for har bir affiks in affikslar_to‘plami:
    agar so‘z oxiri affiksga mos kelsa:
        asos_qismi = so‘z boshidan (so‘z uzunligi - affiks uzunligi) gacha
        qolgan_qism = so‘zning affiksdan oldingi qismi
        for har bir sub_tahlil in Tahlil(qolgan_qism):
          natijalar.append( sub_tahlil + [affiks] )
  return natijalar
```

The *Tahlil (so‘z)* (Analyze(word) function in the above pseudocode recursively finds all possible affix analyses for the given word. The algorithm searches for affixes matching the end of the word, and upon finding one, separates it from the word and recursively calls itself on the remaining part. When the recursion reaches the base case, when only the stem (or empty part) remains, the sequence of found affixes is

returned as the result. For example, the algorithm works as follows for the word "*kitoblarimizdan*":

It identifies the suffix *-dan* (ablative case) at the end, leaving the remainder as "*kitoblarimiz*".

From "*kitoblarimiz*", it finds the suffix *-miz* (1st person plural possessive), leaving "*kitoblari*".

From "*kitoblari*", it finds the suffix *-i* (3rd person singular possessive), leaving "*kitoblar*".

From "kitoblar", it finds the suffix -lar (plural), leaving "kitob".

The stem "kitob" exists in the dictionary, so the recursion stops.

The algorithm stores the sequence of identified affixes and annotates them with grammatical tags.

Stage 4. *Grammatical Verification and Filtering:* The obtained segmentation variants are checked for compliance with the formal model's rules. Any analysis with affix orders or combinations that contradict the model is rejected. This stage corresponds to the accepting states of the deterministic automaton. Analyses leading to invalid paths are discarded. Typically, only one correct analysis remains, although sometimes multiple ambiguous analyses can occur. Although disambiguation is beyond the scope of this work, in some cases it may not be necessary based on rules (e.g., the word "*yuz*" can be either a noun or a verb, with context resolving the ambiguity).

Stage 5. *Lemmatization:* The stem of the word (lemma) separated during segmentation is determined. If the stem is not found in the dictionary, the segmentation may be considered incorrect or the stem may be output as an unknown lemma. Usually, dictionary checks occur within the recursive algorithm's internal stages; the above pseudocode simplifies this process. Lemmas are normalized using an explanatory dictionary and rules: for example, the stem of "*borayotgan*" is taken as "*bor-*" and the lemma is output in its base form as "*bormoq*". When identifying lemmas, cases causing ambiguity such as neologisms, named entities (NERs), abbreviations, dialectal words, measurement units, hyphenated dates, conditional abbreviations, and homonyms must be resolved by a separate algorithm.

Stage 6. *Output Formatting:* In the final stage, the segmentation and grammatical tags are formatted according to the Universal Dependencies (UD) standard. According to UD requirements, each word must include its UPOS (Universal Part of Speech) and FEATS (morphological features). In our case: UPOS is first determined based on the part-of-speech rules applied during analysis. For example, if the model used noun rules, then N – NOUN; if it used verb rules, then V – VERB; and so on. For the FEATS column, the following mapping is applied:

Each identified affix corresponds to a specific UD grammatical category and value. For instance, if the plural suffix *-lar* is detected, the feature *Number=Plur* is added to FEATS. If no plural suffix is found, then the *Number* feature is omitted, as UD conventionally does not specify singular explicitly.

If a case suffix is identified, the Case feature is added: for *-ni Case=Acc*, *-ning Case=Gen*, *-ga Case=Dat*, *-da Case=Loc*, *-dan Case=Abl*, if none is present (zero suffix), *Case=Nom* may be inferred, though UD convention sometimes omits nominative for clarity. The mapping between Uzbek case suffixes and UD Case values is presented in Table VI below:

TABLE VI. MAPPING OF UZBEK CASE SUFFIXES → UD FEATS

№	Traditional name (uz)	Typical suffix(es)	UD Case= value	Explanation
1.	Nominative (basic case)	**Ø** (nol)	Nom	Subject in the sentence, copular noun (kitob bor)
2.	Accusative	**-ni**	Acc	Direct object (*kitob-ni oldim*)
3.	Dative / Directional	**-ga, -qa, -ka**	Dat	Direction/ purpose (*uy-ga, dalaga*)
4.	Locative (place-time case)	**-da, -ta**	Loc	Locative (place,time) (*stol-da, yoz-da*)
5.	Ablative	**-dan, -tan, -din**	Abl	Source/point of departure (*uy-dan ketdi*)
6.	Genitive	**-ning**	Gen	Possessive relationship (*kitob-ning rangi*)

If a **possessive suffix** is identified, it is typically marked in the UD format using the Possessive feature or by specifying the person and number of the possessor. For instance, if the suffix -im (indicating my, 1st person singular possessor) is present, the feature FEATS Person[psor]=1|Number[psor]=Sing is added. For third person possessive forms, (e.g., -lari), then Number[psor]=Plur is used. In some UD corpora, a simpler annotation such as Poss=Yes is used instead. However, this does not provide information about the person of the possessor, which limits the level of granularity. In our model, the goal is to annotate possessive features as precisely as possible.

The mapping of possessive suffixes in Uzbek to →UD Case and related morphological features is presented in Table 6 below.

The following Table VIII presents the results of morphological analysis in UD format for words belonging to different parts of speech:

The complete UD-based morphological tagging model provides an essential foundation for comparing Uzbek with other languages in international linguistic research, as well as for evaluating multilingual models. For instance, in cross-lingual transfer learning using UD corpora, such morphological analysis is crucial for accurately interpreting Uzbek syntactic structures, such as conjunctions and clitics. Moreover, the model can serve as a linguistic backbone for developing grammar exercises or intelligent tutoring systems in Uzbek language education (Table VII, VIII).

TABLE VII. MAPPING OF UZBEK POSSESSIVE SUFFIXES TO →UD CASE

№	TRADITIONAL CLASSIFICATION	TYPICAL SUFFIX FORMS *	UD FEATS VALUE	EXAMPLE	UD TEG (LEBEL)
1.	**I-PERSON SINGULAR** ("MENING …-IM")	-M, -IM/-UM/-EM, -AM	PERSON[PSOR]=1 | NUMBER[PSOR]=SING	*KITOB-IM*	PERSON[PSOR]=1
2.	**II- PERSON SINGULAR** ("SENING …-ING")	-NG, -ING/-UNG/-ENG, -ANG	PERSON[PSOR]=2 | NUMBER[PSOR]=SING	*UY-ING*	PERSON[PSOR]=2
3.	**III- PERSON SINGULAR** ("UNING …-I/-SI")	-I/-Y, -SI	PERSON[PSOR]=3 | NUMBER[PSOR]=SING	*DARS-I*	PERSON[PSOR]=3
4.	**I- PERSON PLURAL** ("BIZNING …-IMIZ")	-MIZ, -IMIZ/-UMUZ/-IMIZ	PERSON[PSOR]=1 | NUMBER[PSOR]=PLUR	*KITOB-IMIZ*	PERSON[PSOR]=1
5.	**II- PERSON PLURAL** ("SIZNING …-INGIZ")	-NGIZ, -INGIZ/-UNGIZ/-INGIZ	PERSON[PSOR]=2 | NUMBER[PSOR]=PLUR	*UY-INGIZ*	PERSON[PSOR]=2
6.	**III- PERSON PLURAL** ("ULARNING …-LARI")	-LARI (-LAR + -I)	PERSON[PSOR]=3 | NUMBER[PSOR]=PLUR	*DARS-LARI*	PERSON[PSOR]=3

TABLE VIII. UD-BASED MORPHOLOGICAL ANALYSIS RESULTS FOR WORDS FROM DIFFERENT PARTS OF SPEECH

№	Word	Lemma	UPOS	FEATS
1.	kitoblarimizdan	kitob	N	Number=Plur | Person[psor]=1 | Number[psor]=Plur | Case=Abl
2.	yozdirilmayapti	yozmoq	VB	VerbForm=Fin | Tense=Pres | Aspect=Prog | Voice=Pass | Polarity=Neg | Person=3 | Number=Sing
3.	yaxshiroq	yaxshi	JJ	Degree=Cmp
4.	uchinchi	uch	NUM	NumType=Ord
5.	tezroq	tez	RR	Degree=Cmp
6.	sizlarga	siz	P	PronType=Prs | Number=Plur | Case=Dat
7.	bilan	bilan	II	AdpType=Post
8.	va	va	C	ConjType=Coord
9.	mi	mi	Prt	PartType=Ques
10.	kerak	kerak	MD	Mood=Pot
11.	sharaqlaridan	sharaq	IM	Onomat=Yes | Number=Plur | Person[psor]=3 | Number[psor]=Plur | Case=Abl
12.	voy	voy	UH	-

IV. CONCLUSION

In conclusion, it is worth emphasizing that formalizing a morphologically rich and complex language like Uzbek is a challenging but highly rewarding endeavor.

The proposed mathematical model and algorithm not only function as practical computational tools but also offer deep insights into the internal structure of the language.

Continuing this work will contribute to the broader integration of Uzbek into the digital world, the advancement of language technologies, and the global recognition of our language.

REFERENCES

[1] D. Yuret, F. Türe. 2006. Learning Morphological Disambiguation Rules for Turkish. In Proceedings of the Human Language Technology Conference of the NAACL, Main Conference, pages 328–334, New York City, USA. Association for Computational Linguistics.

[2] K. Oflazer, I. Kuruoz. 1994. Tagging and Morphological Disambiguation of Turkish Text. In Fourth Conference on Applied Natural Language Processing, pages 144–149, Stuttgart, Germany. Association for Computational Linguistics.

[3] S. Ivanova, J. Washington, F. Tyers. 2022. A Free/Open-Source Morphological Analyser and Generator for Sakha. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 5137–5142, Marseille, France. European Language Resources Association.

[4] K. S. Mirdjanovna, "Finite State Machine Model for Uzbek Language Morphological Analyzer," 2021 6th International Conference on Computer Science and Engineering (UBMK), Ankara, Turkey, 2021, pp. 395-400.

[5] K. Koskenniemi Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production. Helsinki: University of Helsinki. Department of General Linguistics, 1983. – 160 p. (Publications; 11).

[6] K.Beesley, L. Karttunen, (2003). Finite-State Morphology. Bibliovault OAI Repository, the University of Chicago Press.

[7] M. Sharipov, U. Salaev, G. Matlatipov. (2023). Morphological analyzer of Uzbek language using truncating methods. 189-193.

[8] M. Gökırmak, F. Tyers, J. Washington. 2019. A free/open-source rule-based machine translation system for Crimean Tatar to Turkish. In Proceedings of the 2nd Workshop on Technologies for MT of Low Resource Languages, pages 24–31, Dublin, Ireland. European Association for Machine Translation.