

UDC 004.8:81'322

**AUTOMATED CONSTRUCTION OF A PUNCTUATION-TAGGED CORPUS
FOR TRANSFORMER-BILSTM-BASED PUNCTUATION RESTORATION IN UZBEK**

Sharipov Maksud Siddiqovich – *Candidate of Technical Sciences*

maqsbek72@gmail.com

Tashkent University of Information Technologies named after Muhammad al-Khwarizmi

Adinayev Hushnodbek Saylboyevich – *senior teacher*

hushnodbek.adinaev@gmail.com

Urgench State University

**O'ZBEK TILIDA TRANSFORMER-BILSTM ASOSIDA PUNKTUATSIYANI TIKLASH UCHUN
PUNKTUATION TEGLANGAN KORPUSNI AVTOMATIK SHAKLLANTIRISH**

Sharipov Maqsud Siddiqovich – *texnika fanlari nomzodi*

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti

Adinayev Xushnodbek Saylboyevich – *katta o'qituvchi*

Urganch davlat universiteti

**АВТОМАТИЧЕСКОЕ ФОРМИРОВАНИЕ ПУНКТУАЦИОННО РАЗМЕЧЕННОГО
КОРПУСА ДЛЯ ВОССТАНОВЛЕНИЯ ПУНКТУАЦИИ НА ОСНОВЕ
TRANSFORMER-BILSTM В УЗБЕКСКОМ ЯЗЫКЕ**

Шарипов Максуд Сиддиқович – *кандидат технических наук*

Ташкентский университет информационных технологий имени Мухаммада ал-Хорезми

Адинаев Хушнудбек Сайлбоевич – *старший преподаватель*

Ургенчский государственный университет

Tayanch so'zlar: punktuatsiyani tiklash, o'zbek tili, Transformer, BiLSTM, tabiiy tilni qayta ishlash.

Rezyume. Ushbu tadqiqotda o'zbek tili uchun Transformer-BiLSTM modellarini o'qitishga mo'ljallangan punktuatsion teglangan ma'lumotlar to'plamini avtomatik shakllantirish algoritmi taklif etiladi. Jarayon matnni normallashtirish, punktuatsiyaga sezgir tokenizatsiya, token darajasida teglash, ma'lumotlarni ajratish va CoNLL formatiga o'tkazishni o'z ichiga oladi. Jami 11 ta teg qo'llanilib, 11 888 213 ta tokenli korpus shakllantirilgan. Taklif etilgan usul punktuatsiyani tiklash uchun katta hajmdagi o'zbekcha teglangan ma'lumotlarni tayyorlashga xizmat qiladi.

Ключевые слова: восстановление пунктуации, узбекский язык, Transformer, BiLSTM, обработка естественного языка.

Резюме. В данном исследовании предлагается алгоритм автоматического формирования пунктуационно размеченного набора данных для обучения моделей Transformer-BiLSTM на узбекском языке. Процесс включает нормализацию текста, токенизацию с учетом пунктуации, разметку на уровне токенов, разделение данных и преобразование в формат CoNLL. Используется 11 меток, а сформированный корпус содержит 11 888 213 токенов. Предлагаемый метод предназначен для подготовки крупномасштабных размеченных данных на узбекском языке для задачи восстановления пунктуации.

Key words: punctuation restoration, Uzbek language, Transformer, BiLSTM, natural language processing.

Summary. This study proposes an automated algorithm for constructing a punctuation-annotated dataset for training Transformer-BiLSTM models in Uzbek. The process includes text normalization, punctuation-aware tokenization, token-level labeling, dataset partitioning, and conversion into CoNLL format. A total of 11 labels are used, and the resulting corpus contains 11,888,213 tokens. The proposed method is intended for preparing large-scale annotated Uzbek data for punctuation restoration.

Introduction. Punctuation marks are essential for expressing the syntactic structure, semantic meaning, and readability of written text. Their omission or incorrect placement may distort sentence structure and meaning, making automatic punctuation restoration an important task in speech recognition, text editing, machine translation, and other NLP applications.

Automatic punctuation restoration is commonly formulated as a sequence-labeling task. Earlier approaches relied on linguistic rules and statistical models, whereas later studies adopted LSTM, BiLSTM, and attention-based architectures [1], [2]. More recently, Transformer-based models have demonstrated strong performance by capturing long-range contextual dependencies. Courtland et al. [3] applied a bidirectional Transformer to TED Talks data, while Alam et al. [4] showed the effectiveness of

Transformers for both high- and low-resource languages. BERT-based approaches have also proven effective for punctuation restoration [5].

However, large-scale token-level punctuation datasets for Uzbek remain limited. Therefore, this study proposes an automated algorithm for transforming naturally punctuated Uzbek texts into a standardized dataset for Transformer-BiLSTM models. The proposed pipeline includes text normalization, punctuation-aware tokenization, token-level label assignment, train/validation/test splitting, and conversion into CoNLL format.

Related Work. Studies on automatic punctuation restoration can generally be divided into rule-based and statistical approaches, recurrent neural networks, and Transformer-based models. LSTM and BiLSTM architectures enabled bidirectional sequence modelling [1],

[2], while Transformer models improved the representation of long-range contextual dependencies through self-attention [3], [5].

Early research on Uzbek punctuation mainly relied on linguistic rules. Sharipov, Adinaev and Kuriyozov [6] proposed more than 30 rules based on the morphological and syntactic characteristics of Uzbek and achieved approximately 84% accuracy on a 30,000-word corpus from 30 domains. Sharipov and Adinaev [7] later developed functional and information models for punctuation analysis using IDEF0 and IDEF1X methodologies together with a rule-based system.

Statistical approaches were subsequently explored. Adinaev [8] proposed an N-gram model evaluated on 137,848 sentences and 1,642,860 words, achieving F1-scores of up to 87% for several punctuation classes and an overall performance of approximately 81%. Sharipov, Adinayev and Yusupova [9] also developed an Uzbek punctuation corpus containing 137,848 sentences and 1,642,860 words from 30 domains. In addition, Sharipov and Adinayev [10] proposed a rule-based method for detecting interrogative sentences, reporting 87.9% performance.

For sequence labeling, Sharipov and Adinayev [11] applied a Conditional Random Fields (CRF) model to an IT-domain corpus containing more than 1.5 million words and 295,012 punctuation marks, achieving approximately 87.5% accuracy. The scalability of punctuation processing was also investigated using parallel and Apache Spark-based distributed computing, which considerably reduced processing time for larger text collections [12].

Overall, previous studies have addressed rule-based, statistical, corpus-based, and sequence-labeling approaches for Uzbek punctuation. However, the automatic transformation of large-scale naturally punctuated Uzbek text into a standardized dataset through normalization, punctuation-aware tokenization, 11-class token-level annotation, train/validation/test splitting, and CoNLL-format generation has not been fully integrated into a single pipeline. The present study addresses this gap by providing a standardized data preparation procedure for Transformer and Transformer-BiLSTM models.

Materials and Methods.

Problem formulation. The given sequence of text tokens is represented as: $X=(x_1, x_2, \dots, x_n)$.

For each token x_i , the corresponding punctuation class $y_i \in Z$ is determined.

The set of classes is defined as:

$Z=\{O, \text{PERIOD}, \text{COMMA}, \text{QUESTION}, \text{EXCLAMATION}, \text{COLON}, \text{SEMICOLON}, \text{DASH}, \text{ELLIPSIS}, \text{QUOTE}, \text{PAREN}\}$.

Here, **O** indicates that no punctuation mark follows the token.

The subsequent task of the model can be expressed as: $\bar{y}_i = \operatorname{argmax} P(y_i = c | X)$ $c \in Z$ where y_i denotes the punctuation class predicted by the model for the i -th token.

An important aspect is that a punctuation mark is not represented as an independent target token; instead, it is assigned as the class label of the preceding word.

Text normalization. At the initial stage of the algorithm, the raw text is read from a file and standardized.

The normalization function can be expressed as:

$T'=N(T)$ It can be expressed as follows.

Here, T denotes the raw text, N is the normalization function, and T' represents the standardized text.

For the Uzbek language, it is particularly important to normalize the following characters into a unified form:

- o', o';
- g', g';
- various quotation marks;
- dash and hyphen variants;
- multiple consecutive spaces;
- incorrect line breaks.

This stage is particularly important because multiple Unicode representations of the same linguistic symbol may be interpreted by the tokenizer as different tokens.

Punctuation-aware tokenization. The normalized text is divided into a sequence of lines: $T'=\{L_1, L_2, \dots, L_m\}$

For each line, the following tokenization operation is performed:

$K_j = \text{Tokenize}(L_j)$

For example, the sentence:

Bugun havo yaxshi, ertaga esa yomg'ir yog'adi. quyidagi ketma-ketlikka aylantiriladi:

is transformed into the following sequence:

[Bugun, havo, yaxshi, ",", ertaga, esa, yomg'ir, yog'adi, "."]

Assigning punctuation labels. The main part of the algorithm consists of converting punctuation tokens into class labels.

For each regular token a (x_i, O) pair is created.

If the next element is a punctuation mark pp , the **O** label of the preceding token is replaced with the corresponding punctuation label according to $y_i = M(p)$ where $M(p)$ denotes the mapping function that assigns a punctuation mark p to its corresponding class label.

Here, M is a mapping function that maps punctuation marks to their corresponding classes:

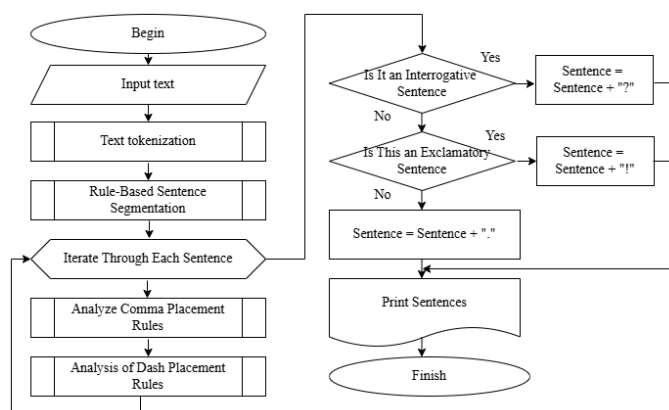
$M(.)=\text{PERIOD}$, $M(,)=\text{COMMA}$, $M(?)=\text{QUESTION}$, $M(!)=\text{EXCLAMATION}$ etc.

For example, the sentence "*Men keldim, u ketdi.*" is transformed as follows:

This approach is highly suitable for **Transformer-based token classification tasks**.

Algorithm.

Figure 1. Flowchart of the Algorithm for Constructing a Punctuation-Annotated Dataset for the Transformer-BiLSTM Hybrid Model



Block diagram of the proposed approach. An algorithm was developed for the automatic construction of a punctuation-annotated dataset for training Transformer–BiLSTM hybrid models. The algorithm includes the following stages: reading and normalizing raw text, punctuation-aware tokenization, distinguishing lexical and punctuation units, assigning punctuation marks to preceding tokens as class labels, validating the generated sequences, and dividing them into training, validation, and test subsets. The final data are stored in CoNLL format suitable for sequence-labeling models. The execution flow of the proposed algorithm is illustrated in **Figure 1**.

Results and Discussion

Constructed dataset. Using the proposed algorithm, a total of 11,888,213 annotated tokens were generated.

Of these, 9,003,523 tokens belong to the O class, while the remaining 2,884,690 tokens belong to the punctuation classes.

This corresponds to approximately:

75.73% – O class;

24.27% – punctuation classes.

Model natijalari bo'yicha juda muhim masala. The experimental evaluation was conducted using eleven classes: the non-punctuation class O and ten punctuation classes, namely PERIOD, COMMA, QUESTION, EXCLAMATION, COLON, SEMICOLON, DASH, ELLIPSIS, QUOTE, and PAREN. All models were evaluated under the same corpus partitioning and evaluation protocol, ensuring a consistent comparison across CRF, BiLSTM, Transformer-based, and Transformer–BiLSTM architectures.

Table 1. Test results

Class	Precision	Recall	F1-score
O	0,8924	0,9803	0,9343
PERIOD	0,8076	0,7933	0,8004
COMMA	0,6665	0,7249	0,6945
QUESTION	0,9248	0,5523	0,6916
EXCLAMATION	0,3555	0,1011	0,1574
COLON	0,7893	0,4797	0,5967
SEMICOLON	0,7707	0,2787	0,4094
DASH	0,7066	0,2135	0,3279
ELLIPSIS	0,6695	0,2815	0,3963
QUOTE	0,8357	0,2601	0,3967
PAREN	0,7228	0,2604	0,3829

Conclusion. This study proposed an automated algorithm for constructing punctuation-tagged datasets for Transformer–BiLSTM-based punctuation restoration models in Uzbek. The proposed pipeline integrates text normalization, punctuation-aware tokenization, token-level label assignment, sequence validation, dataset partitioning, and CoNLL-format generation within a unified procedure. Using the developed algorithm, a corpus containing 11,888,213 labeled tokens was generated across 11 classes, including ten punctuation categories and the non-punctuation class. Analysis of the resulting corpus revealed considerable class imbalance, emphasizing the importance of class-wise and macro-averaged evaluation measures. The generated dataset provides a standardized data representation suitable for Transformer, recurrent, and hybrid sequence-labeling architectures. Future work will focus on handling composite punctuation structures, balancing rare punctuation categories, and evaluating the dataset using multilingual Transformer–BiLSTM architectures across different text domains.

References

1. Tilk O., Alumäe T. LSTM for punctuation restoration in speech transcripts. Proceedings of Interspeech 2015.
 2. Tilk O., Alumäe T. Bidirectional recurrent neural network with attention mechanism for punctuation restoration. // Proceedings of Interspeech 2016. – P. 3047-3051.
 3. Courtland M., Faulkner, A., McElvain, G. Efficient automatic punctuation restoration using bidirectional Transformers with robust inference. 2020, Proceedings of the 17th International Conference on Spoken Language Translation. – P. 272-279.
 4. Alam T., Khan A., Alam, F. Punctuation restoration using Transformer models for high – and low-resource languages. Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020).
 5. Nagy A., Bial B., Ács J. Automatic punctuation restoration with BERT models. arxiv preprint arxiv: 2101.07343.
 6. Sharipov M.S., Adinaev H.S., Kuriyozov E.R. Rule-Based Punctuation Algorithm for the Uzbek Language. 2024 IEEE 25th International Conference of Young Professionals in Electron Devices and Materials (EDM), 2024. – P. 2410-2414.
 7. Sharipov M.S., Adinaev H.S. Development of Models for Punctuation Analysis of Uzbek Language Texts. // Bulletin of TUIT: Management and Communication Technologies, 2025, 1(4).
 8. Adinaev H.S. Punctuation Analysis of Uzbek Texts Based on the N-Gram Model. // Electronic Journal of Actual Problems of Modern Science, Education and Training, 2025, № 2. – P. 104-10.
 9. Sharipov M.S., Adinayev H.S., Yusupova M.M. O'zbek tili matnlarida punktuatsion tahlil qilish uchun korpus yaratish. // Development of Science, 2025, Vol. 2, №3, 128-133-b.
 10. Sharipov M., Adinayev H. O'zbek tili matnlarida so'roq gaplarni aniqlashning qoidaga asoslangan algoritmini ishlab chiqish. // Management and Future Technologies, 2025, Vol. 2, Issue 1, 182-89-b.
 11. Sharipov M.S., Adinayev H.S. Shartli tasodifiy maydonlar modeli asosida o'zbek tili matnlarini punktuatsion tahlil qilish. // Al-Farg'oniy avlodlari, 2025, Vol. 1, Issue 2.
- Sharipov M.S., Adinayev H.S. Parallel hisoblash va Apache Spark asosida katta hajmdagi matnlarni punktuatsion tahlil qilish. // O'zbekiston milliy axborot agentligi – O'za ilm-fan bo'limi (elektron jurnal), Vol. 11(73), November 2025, 243-247-b.