**Инженерный вестник Дона**, №7 (2023)
ivdon.ru/ru/magazine/archive/n7y2023/8571

# Поиск персональных данных в неструктурированных текстах с использованием нейронных сетей

*Е.С. Раздьяконов*

*Финансовый университет при Правительстве Российской Федерации, Москва*

**Аннотация:** В данной статье описывается создание гибридной системы для задачи распознавания различных видов персональных данных в неструктурированных текстах. В основу системы легла нейронная сеть архитектуры ELMo-BiLSTM-CRF и регулярные выражения. Для обучения и валидации нейронной сети был использован специализированный русскоязычный набор данных для задачи распознавания именованных сущностей, созданный на основе наборов Nerus и WiNER. Полученная гибридная модель позволит снизить издержки организаций на хранение и обработку текстовых данных, а также сохранить конфиденциальность пользователей в случае утечек.
**Ключевые слова:** персональные данные, обработка естественного языка, распознавание именованных сущностей, условное случайное поле, нейронная сеть, рекуррентная нейронная сеть, регулярное выражение.

## Введение

В последнее время большие данные широко используются для получения знаний во многих сферах деятельности. Однако, это создает очень серьезную проблему безопасности, поскольку подобные данные могут содержать информацию, которая позволяет прямо или косвенно узнать и идентифицировать того или иного человека.

Согласно федеральному закону РФ № 152, такая информация квалифицируется как персональные данные. Они должны надлежащим образом храниться и обрабатываться как юридическими лицами, так и органами государственной власти. Поэтому важно анонимизировать их перед использованием, чтобы убедиться, что это этично и не приведет к утечке частной информации. Под текстовыми данными понимаются наборы документов, записи переговоров, протоколы, журналы и т.п., содержащие разрозненные упоминания персон и идентифицирующие их сведения.

Наиболее важным этапом в процессе анонимизации персональных данных в текстах является их поиск и классификация, который, как правило,

осуществляется путем решения задачи распознавания именованных сущностей.

Распознавание именованных сущностей (named entity recognition, NER) – это задача обнаружения и классификации именованных сущностей в тексте. Под ними можно понимать все, на что можно сослаться с помощью имени, например, организацию, человека или место. Распознавание именованных сущностей выступает в качестве инструмента для многих задач обработки естественного языка, включая системы извлечения информации [1], диалоговые системы [2] и т.д.

Большинство ранних подходов в NER было основано на правилах (rule-based). Такие методы показали свою достаточную эффективность, однако их ограничения заключаются в том, что они достаточно дороги в разработке, специфичны для конкретной области и не переносимы.

Для более гибкого решения задачи распознавания сущностей широко используются методы машинного обучения с учителем. Исследователи в данной области используют различные алгоритмы обучения при разработке NER систем. Примерами таких алгоритмов являются метод опорных векторов (SVM) [3], условные случайные поля (Conditional Random Field, CRF) [4], логистическая регрессия [5].

В последнее время, с развитием технологий глубокого обучения, в задаче NER все чаще стали использоваться нейросетевые модели. Системы описанные в [6], показали, что использование глубокого обучения в целом улучшает точность распознавания сущностей по сравнению с классическими подходами. В системах NER на основе нейросетевых моделей совмещается два этапа: эмбеддинг (embedding), то есть, представление текстовой информации, преимущественно слов, в виде числового вектора, и нейронные сети.

Сравнение точности методов машинного обучения для русского языка было проведено в [7]. Рассматриваемые методы включали в себя 5 классических статистических методов, а также 2 метода на основе нейронных сетей. Для каждого из них рассматривалось два различных эмбеддинга: word2vec и ELMo. В результате исследования, архитектура двунаправленных LSTM-сетей с CRF слоем (BiLSTM-CRF) и эмбеддингом ELMo показала наилучшую точность на тестовой выборке. Данная архитектура и ее преимущества также были описаны в [8].

ELMo (Embeddings from Language Models) – модель текстовых эмбеддингов, которая представляет собой контекстуализированные векторы [9]. Одной из особенностей данной архитектуры является использование двунаправленных LSTM-слоев, благодаря которым модель способна учитывать контекст слова на основе как предшествующих, так и последующих слов.

В задаче распознавания именованных сущностей возможно использование гибридных подходов, которые могут сочетать в себе методы, основанные на обучении, и rule-based методы. Как отмечено в [10], гибридные методы могут демонстрировать лучшие результаты, чем методы, использующие только один из подходов.

Целью данного исследования является создание гибридной модели на основе правил и нейронных сетей. Для создания правил применяются регулярные выражения. В качестве нейросетевой части гибридной модели исследуется архитектура ELMo-BiLSTM-CRF. Распознаваться будут такие персональные данные, как ФИО, адреса, наименования организаций, номера телефонов, серия и номер паспорта, ИНН, СНИЛС, полис ОМС.

### Методы на основе правил

В задаче NER возможно эффективно использовать rule-based методы, если для области существуют заранее известные шаблоны определенных типов именованных сущностей.

В данном случае, четко определенный формат имеют такие виды персональной информации, как номер и серия паспорта, ИНН, СНИЛС, полис ОМС и номер телефона. Также для подобных типов данных отсутствует достаточное количество размеченных наборов данных, что затрудняет использование многих методов машинного обучения и делает использование правил еще более оправданным.

**Номер телефона**

Для поиска номеров телефона в тексте было использовано следующее регулярное выражение:

```
(((8|\+7)[\- \s]?)?(\(?\d{3}\)?[\-\s]?)?[\d\-\s]{7,16})
```

Данное выражение предусматривает поиск российских мобильных и городских номеров и работает с большинством используемых форматов записи.

**Серия и номер паспорта**

Формат серии и номера паспорта может иметь вид (X – серия, любая цифра, Y – номер, любая цифра):

- XX XX YYYYYY
- XXXX YYYYYY
- XXXXYYYYYY

Также возможен разделитель между серией и номером, например:

- XXXX № YYYYYY
- XX XX номер YYYYYY

Для данных форматов записи регулярное выражение имеет вид:

```
((\d{2}\s?\d{2})\D{0,10}(\d{6}))
```

**ИНН**

ИНН представляет собой последовательность 12 цифр для физических лиц и из 10 – для юридических лиц без пробелов. Таким образом, регулярное выражение имеет вид:

```
(([0-9]{12})|([0-9]{10}))
```

**СНИЛС**

Номер СНИЛС состоит из 11 цифр с разделителями. Основные маски ввода имеют вид:

- XXXXXXXXXXX
- XXX-XXX-XXX-XX
- XXX-XXX-XXX XX

Для данного шаблона было составлено следующее регулярное выражение:

```
((\d{3}[\- ]?){3} ?\d{2})
```

**Полис ОМС**

Данный номер состоит из 16 цифр и может иметь разделители. Основные маски имеют вид:

- XXXXXXXXXXXXXXXX
- XXXX XXXX XXXX XXXX

Выражение для масок выглядит следующим образом:

```
((\d{4}[\s-]?){4})
```

## Создание набора данных

Для создания размеченной выборки для обучения и валидации нейронных сетей было использовано два крупнейших датасета для задачи NER на русском языке.

В качестве входных данных будущих моделей необходимы наборы фиксированной длины. Это значит, что данная длина будет максимальным

размером входного предложения. Была выбрана максимальная длина размером 100 токенов, что позволит обрабатывать достаточно длинные предложения с большим количеством знаков пунктуации.

**Nerus** [11] – это крупнейший набор данных для распознавания именованных сущностей на русском языке. Он представляет собой статьи из Lenta.ru с размеченными тегами для NER, морфологией и синтаксисом. Содержит около 700 тысяч новостных статей, что делает его крупнейшим набором данных для NER на русском языке.

Nerus использует стандарт IOB (Inside-Outside-Beginning) для разметки именованных сущностей в тексте. В формате IOB каждый токен в тексте помечается тегом, указывающим его принадлежность к определенному классу сущности. Тег состоит из двух частей: одна часть указывает на тип сущности (например, PER для людей, LOC для мест и ORG для организаций), а другая часть указывает на положение токена внутри сущности. Тег "B" (Beginning) используется для первого токена сущности, тег "I" (Inside) - для всех последующих токенов внутри сущности, а тег "O" (Outside) - для всех токенов, не относящихся к сущности.

Суммарно набор данных содержит более 8 миллионов предложений (более 150 миллионов токенов), которые размечены тегами PER, LOC и ORG.

Длина предложений в датасете варьируется от 1 до 249 токенов. Распределение длин предложений в данной наборе изображено на рисунке 1.

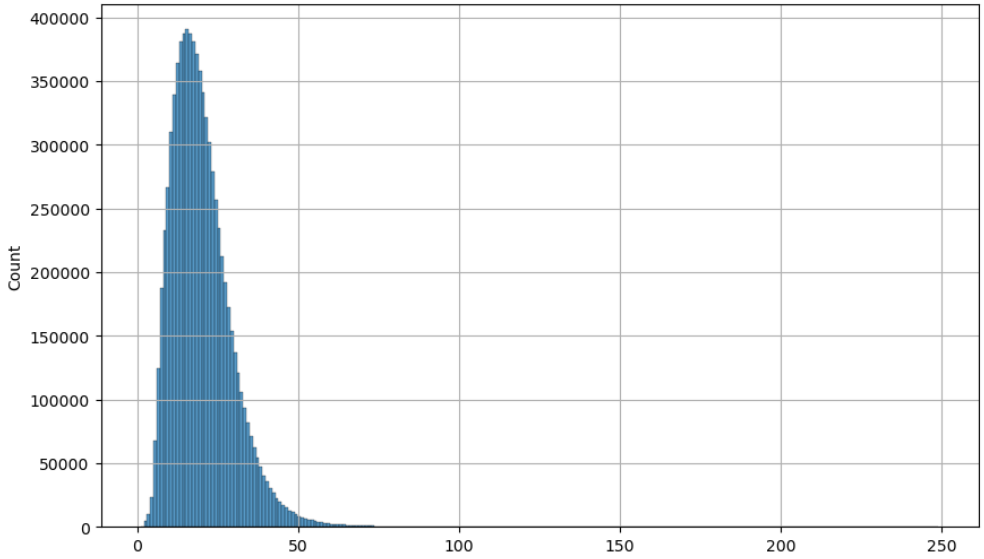


Рис. 1. – Гистограмма распределения длин предложений в Nerus

Доля предложений длиной более 100 незначительна по сравнению с объемом всей выборки и составляет 0.02% (2029 предложений), следовательно, их можно отбросить без значительных потерь. Также было исключено 663 предложения длиной в 1 токен, который являлся пустой строкой, либо символом переноса строки.

**WiNER** [12] – набор, созданный путем автоматической разметки статей Википедии для нескольких языков, включая русский. Набор содержит четыре вида сущностей, размеченных в формате IOB: люди, локации, организации и «прочее». Объем для русского языка составляет порядка 200 тысяч предложений. Помимо тегов O, PER, ORG и LOC использовался также и тег MISC для обозначения прочих сущностей.

Длина предложений варьировалась от 1 до 219 токенов. Распределение длин предложений представлено на рисунке 2.

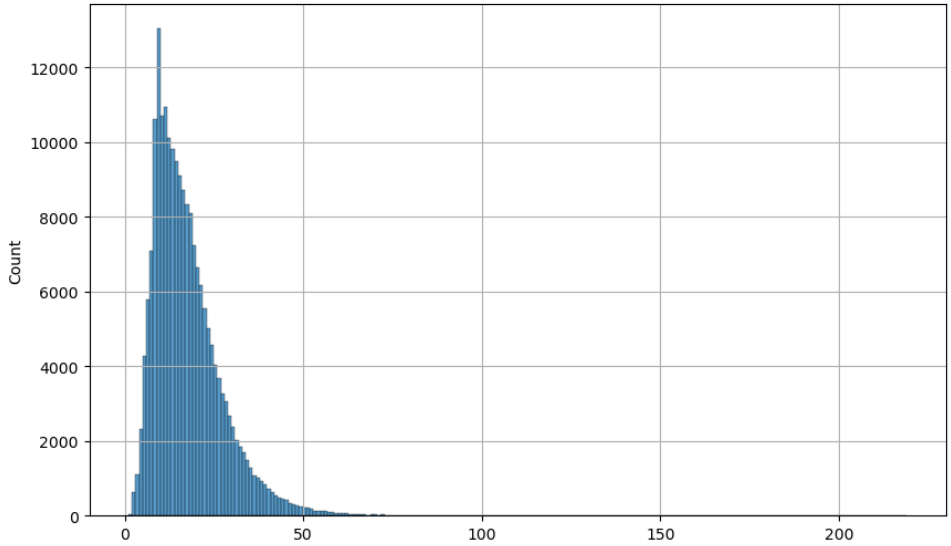


Рис. 2. – Гистограмма распределения длин предложений в WiNER

Как и в случае с Nerus, были отброшены предложения длиной больше 100 или равных 1. Доля отброшенных предложений составила 0.04% от общего объема.

Однако разметка в WiNER с учетом положения токена внутри сущности редко учитывала начало сущности, и большинство начальных тегов было помечено, как внутренние. Для улучшения качества разметки была произведена замена внутреннего тега на начальный, если перед ним не следовало тега данного типа сущности. Изменение количества тегов в результате замены и пример преобразования показаны в таблицах 1 и 2.

Таблица № 1

Количество тегов до и после предобработки

| Тег | Исходное количество | Итоговое количество |
|---|---|---|
| B-PER | 161 | 71626 |
| I-PER | 136345 | 64880 |
| B-ORG | 51 | 37119 |
| I-ORG | 63666 | 26598 |
| B-LOC | 1457 | 144379 |
| I-LOC | 191820 | 48898 |

Таблица № 2

Предобработка тегов в WiNER

| Токен | Исходный тег | Итоговый тег |
|---|---|---|
| 8 | O | O |
| апреля | O | O |
| 1986 | O | O |
| года | O | O |
| состоялся | O | O |
| визит | O | O |
| М. | I-PER | B-PER |
| С. | I-PER | I-PER |
| Горбачёва | I-PER | I-PER |
| в | O | O |
| Тольятти | I-LOC | B-LOC |
| , | O | O |
| где | O | O |
| он | O | O |
| посетил | O | O |
| Волжский | I-ORG | B-ORG |
| Автозавод | I-ORG | I-ORG |
| . | O | O |

## Разработка нейронной сети

Сеть ELMo-BiLSTM-CRF представляет собой архитектуру BiLSTM-CRF, использующую эмбеддинги ELMo в качестве входа.

Для эмбеддинга использована модель ELMo из проекта deeppavlov, предобученная на текстах русскоязычных новостей. Эмбеддинг одного токена представляет собой вектор из 1024 компонент. Эмбеддинги для набора данных были вычислены заранее. Таким образом, каждое предложение было представлено в виде массива размером (100, 1024). В данном исследовании всего было использовано 100 000 предложений, по 50 000 предложений из каждого набора данных. Валидационная выборка составила 0.2 от общего объема.

Нейронная сеть разрабатывалась на языке Python с использованием интерфейса Keras библиотеки Tensorflow. Обучение проходило с использованием графического процессора Nvidia GeForce RTX 4080.

Для предотвращения переобучения, на слоях использовалась L1-регуляризация, а также после каждого слоя использовался Dropout-слой.

Архитектура сети выглядит следующим образом:

- BiLSTM(1024*2)
- Dropout(0.3)
- LSTM(1024, ReLU)
- Dropout(0.3)
- TimeDistributed(Dense(512, ReLU))
- Dropout(0.5)
- CRF

Обучение проходило с размером батча, равным 400, максимальным количеством эпох, равным 80 и ранней остановкой при отсутствии прогресса по метрике macro-f1 в течение 8 эпох. В качестве алгоритма оптимизации использовался алгоритм Adam со скоростью обучения, равной 1e-3.

Лучшие результаты были достигнуты на 17 эпохе с точностью по метрике macro-F1, равной 91.36% на валидационной выборке. Динамика точности в процессе обучения модели представлена на рисунке 3.


Рис. 3. – Динамика точности модели ELMo-BiLSTM-CRF

Метрики точности модели на различных классах и матрица ошибок представлены на рисунках 4 и 5.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| O | 0.9949 | 0.9945 | 0.9947 | 325441 |
| B-PER | 0.9627 | 0.9503 | 0.9565 | 7524 |
| I-PER | 0.9760 | 0.9617 | 0.9688 | 5826 |
| B-ORG | 0.8506 | 0.8891 | 0.8694 | 6059 |
| I-ORG | 0.8183 | 0.8721 | 0.8443 | 4808 |
| B-LOC | 0.9451 | 0.9404 | 0.9427 | 11236 |
| I-LOC | 0.8603 | 0.7807 | 0.8186 | 2627 |
| micro avg | 0.9865 | 0.9865 | 0.9865 | 363521 |
| macro avg | 0.9154 | 0.9127 | 0.9136 | 363521 |
| weighted avg | 0.9866 | 0.9865 | 0.9865 | 363521 |

Рис. 4. – Показатели точности модели для отдельных классов

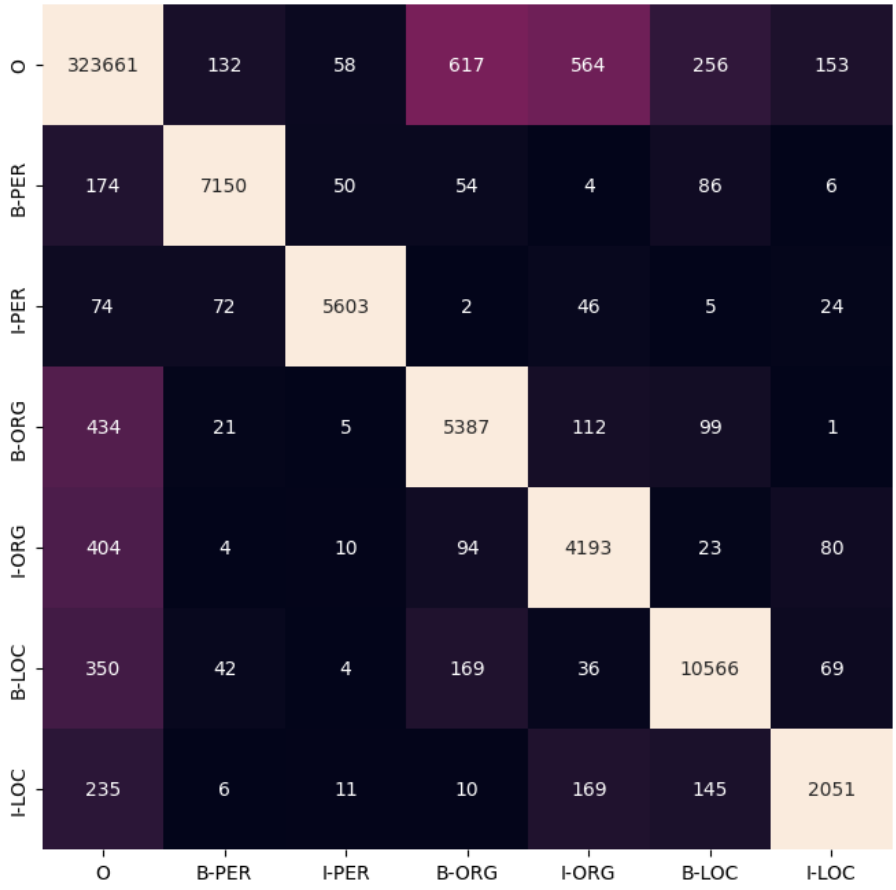


Рис. 4. – Матрица ошибок модели

Как видно из рисунков 3 и 4, классы B-ORG, I-ORG и I-LOC распознаются заметно хуже остальных. Судя по матрице ошибок, чаще всего они неверно классифицируются, как тег «O». Лучше всех распознаются теги сущностей B-PER и I-PER, связанные с именами людей.

## Гибридная модель

Гибридная модель основана на вышеописанной нейронной сети и правилах на основе регулярных выражений. Для разделения текста на предложения и токенизации используется Python-библиотека razdel, созданная специально для русского языка. Для эмбеддинга токенов используется загруженная модель ELMo.

Несмотря на то, что нейросетевая модель способна достаточно точно распознавать названия городов и улиц, она не распознает численные сущности, например, номера домов и квартир. Для решения данной проблемы, в выходе нейронной сети дополнительно отмечались токены, содержащие цифры и идущие на небольшом расстоянии после сущностей LOC.

Таким образом, процесс распознавания сущностей состоит из следующих этапов:

1. Разбиение текста на предложения методом razdel.sentenize
2. Применение регулярных выражений для поиска структурированных сущностей и их позиций в предложениях
3. Токенизация предложений методом razdel.tokenize
4. Эмбеддинг токенов в предложениях с помощью ELMo
5. Разметка токенов моделью BiLSTM-CRF
6. Распознавание номеров в адресах
7. Вывод результатов на основе выходов двух моделей

## Заключение

В данной статье была описана реализация нейронной сети архитектуры ELMo-BiLSTM-CRF для распознавания именованных сущностей, ее обучение и тестирование на созданном наборе данных, а также разработка rule-based - методов на основе регулярных выражений для определения отдельных видов сущностей.

В результате была получена гибридная модель распознавания и классификации личных данных в неструктурированных текстах, которая может лечь в основу системы автоматической анонимизации персональных данных.

У полученной модели существует несколько возможных вариантов улучшения:

**Использование больших моделей глубокого обучения.** Для улучшения качества распознавания сущностей могут быть использованы нейросетевые модели схожей архитектуры, но большего размера, обученные на больших выборках (например, на всем объеме набора Nerus) мощными вычислительными кластерами.

**Распознавание адресов с использованием ГАР.** Точность распознавания данных можно улучшить, используя модуль, сверяющий адреса в текстах с государственным адресным реестром (ГАР), что может заметно увеличить гибкость анонимизации (например, возможность исключать только адреса жилых домов).

## Литература

1. Hoffmann R., Zhang C., Ling X., Zettlemoyer L., Weld D.S. Knowledge-based weak supervision for information extraction of overlapping relations. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2011. pp. 541–550.

2. Маслова М.А., Бажутова Д.А., Дмитриев А.С. Алгоритмы работы чат-бота для поиска товаров // Инженерный вестник Дона. 2021, №4. URL: ivdon.ru/ru/magazine/archive/n4y2021/6921/.

3. Saha S.K., Narayan S., Sarkar S., Mitra P. A composite kernel for named entity recognition. Pattern Recognition Letters, 2010, 31 (12). pp. 1591-1597.

4. Majumder M., Barman U., Prasad R., Saurabh K., Saha S.K. A novel technique for name identification from homeopathy diagnosis discussion forum. Procedia Technology, 2012, 6. pp. 379–386.

5. Ek T., Kirkegaard C., Jonsson H., Nugues P. Named entity recognition for short text messages. Procedia - Social and Behavioral Sciences, 2011, 27. pp. 178-187.

6. Lample G., Ballesteros M., Subramanian S., Kawakami K., Dyer C. Neural architectures for named entity recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2016. pp. 260-270.

7. Gultiaev A.A, Domashova J.V. Developing a named entity recognition model for text documents in Russian to detect personal data using machine learning methods. Procedia Computer Science, 2022, 213. pp. 127-135.

8. Маслова М.А., Дмитриев А.С., Холкин Д.О. Методы распознавания именованных сущностей в русском языке // Инженерный вестник Дона, 2021. №7. URL: ivdon.ru/ru/magazine/archive/n7y2021/7066/.

9. Peters M., Neumann M., Iyyer M., Gardner M., Clark C., Lee K., Zettlemoyer L. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2018. pp. 2227–2237.

10. Jehangir B., Radhakrishnan S., Agarwal R. A survey on Named Entity Recognition — datasets, tools, and methodologies. Natural Language Processing Journal, 2023, 3. URL: sciencedirect.com/science/article/pii/S2949719122000036/.

11. Nerus: Large silver standart Russian corpus with NER, morphology and syntax markup. URL: github.com/natasha/nerus (date assessed: 10.06.2023).

12. Ghaddar A., Langlais P. WiNER: A Wikipedia Annotated Corpus for Named Entity Recognition. Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1), Asian Federation of Natural Language Processing, 2017. pp. 413–422.

## References

1. Hoffmann R., Zhang C., Ling X., Zettlemoyer L., Weld D.S. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2011. pp. 541–550.

2. Maslova M.A., Bazhutova D.A., Dmitriev A.S. Inzhenernyj vestnik Dona. 2021. №4. URL: ivdon.ru/ru/magazine/archive/n4y2021/6921/.

3. Saha S.K., Narayan S., Sarkar S., Mitra P. Pattern Recognition Letters, 2010, 31 (12). pp. 1591-1597.

4. Majumder M., Barman U., Prasad R., Saurabh K., Saha S.K. Procedia Technology, 2012, 6. pp. 379–386.

5. Ek T., Kirkegaard C., Jonsson H., Nugues P. Procedia - Social and Behavioral Sciences, 2011, 27. pp. 178-187.

6. Lample G., Ballesteros M., Subramanian S., Kawakami K., Dyer C. 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2016. pp. 260-270.

7. Gultiaev A.A, Domashova J.V. Procedia Computer Science, 2022, 213. pp. 127-135.

8. Maslova M.A., Dmitriev A.S., Kholkin D.O. Inzhenernyj vestnik Dona. 2021. №7. URL: ivdon.ru/ru/magazine/archive/n7y2021/7066/.

9. Peters M., Neumann M., Iyyer M., Gardner M., Clark C., Lee K., Zettlemoyer L. 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2018. pp. 2227–2237.

10. Jehangir B., Radhakrishnan S., Agarwal R. Natural Language Processing Journal, 2023, 3. URL: sciencedirect.com/science/article/pii/S2949719122000036/.

11. Nerus: Large silver standart Russian corpus with NER, morphology and syntax markup. URL: github.com/natasha/nerus (accessed 06/10/23).

12. Ghaddar A., Langlais P. Eighth International Joint Conference on Natural Language Processing (Volume 1), Asian Federation of Natural Language Processing, 2017. pp. 413–422.