НЕЙРОННОЕ ПРЕДСТАВЛЕНИЕ СЕМАНТИЧЕСКОГО ПОЛЯ

© Р. Г. Мифтахова*, Е. А. Морозкина

Башкирский государственный университет

Россия, Республика Башкортостан, 450076 г. Уфа, ул. Коммунистическая, 19.

Тел.: +7 (927) 332 66 51.

*Email: miftahovar@yandex.ru

Важной проблемой при обработке естественного языка является многозначность, требующая от системы понимания контекста и предметной сферы, в которой употребляется слово. Такие сведения могут быть представлены в так называемой семантической сети, используемой в нейронном машинном переводе. Это специальные базы данных, представляющие сведения о лексике в виде понятий и связей между ними. Эти сведения образуют семантическую иерархию. В сфере обработки естественного языка сегодня самой популярной семантической сетью для английского языка является семантическая сеть WordNet. Она широко применяется в системах машинного перевода. Создание таких тэгированных баз для других языков – это длительный и трудоемкий процесс, поэтому ученые уделяют большое внимание автоматическому построению семантических сетей. В данной работе представлены алгоритмы обработки текста на высокофлективном языке для выявления семантического поля лексем. Проведен также сравнительный анализ примененных алгоритмов для аналитического английского языка.

Ключевые слова: обработка естественного языка, семантическое поле, модели и алгоритмы, флективные и аналитические языки.

Наш мир неотвратимо и непрерывно меняется, достижения в разных областях науки, особенно в области информационных технологий, в корне меняют то, как люди взаимодействуют в повседневной жизни, ведут бизнес, управляют разными сферами деятельности. Движущей силой этих изменений является количественная лингвистика. Обработка естественного языка (ОЕЯ) – это «развивающееся направление искусственного интеллекта» [1], которое переросло из чисто филологической области в другие области научных знаний, связанных «с применением нейронных сетей» [7]. Технологические инновации позволяют развивать современные теории искусственного интеллекта и предлагать все более эффективные методы ОЕЯ. В рамках данной статьи рассматриваются проблемы, с которыми приходится сталкиваться при выявлении семантического поля лексем в системах нейронного машинного перевода с опорой на модель «word2vec». В статье исследованы также приемы выявления семантического поля для высокофлективных языков.

Понятие «флексия» является неотъемлемой частью морфологии многих естественных языков, однако в последние годы было предпринято недостаточно исследований с целью унификации вычислительных методов для флективных языков.

Флексия – это способ преобразования слова за пределами основы слова, на уровне морфологии, для выражения синтаксических отношений между словами в словосочетании и предложении или для дополнения значения слова без его изменения. В естественных языках флексии подвергаются в основном такие части речи, как существительные, прилагательные и глаголы, а все возможные формы, которые может приобретать слово, представ-

ляют собой флексионную парадигму, где лемма является канонической формой слова [2].

Согласно морфологии, языки делятся на четыре типа: изолирующие (морфем мало или вовсе отсутствуют, напр., китайский), аналитический (схож с изолирующим, но морфологических процессов больше, напр., английский) и синтетический. Последний может быть разделен на фузионный и агглютинативный. Русский язык относится к фузионным или флективным языкам.

Онлайн ежедневно генерируется огромное количество данных, большая часть которых представлена в текстовом виде. Цель ОЕЯ заключается в том, чтобы обработать эти данные и получить новую информацию. Компьютеры могут легко обрабатывать структурированные данные. Но данные на естественном языке (тексте) неструктурированы. Для их обработки необходимы специальные подходы, модели и алгоритмы. ОЕЯ, являясь частью искусственного интеллекта, разрабатывает различные способы обработки текста, которые бы позволили компьютеру «понимать» язык, подобно человеку. ОЕЯ имеет уже 20–30-летнюю историю активных исследований, но многие проблемы остаются нерешенными. До сих пор нет понимания, как человек обрабатывает язык. Несмотря на это, сегодня ОЕЯ решает много важных задач, среди которых генерирование краткого содержания текста, автоматическое распознавание речи, преобразование текста в речь, машинный перевод, выявление тональности текста, семантический анализ, генерирование текста из структурированных данных и многие другие.

В рамках данной статьи было проведено исследование, направленное на сравнительный анализ

полученных результатов после обработки аналитического английского языка и высоко флективного русского языка, и предложены модели и алгоритмы, которые позволяют получить для флективного языка результаты, сравнимые с результатами для аналитического языка. Разница в полученных данных объясняется тем, что русский язык имеет развитый процесс формирования слов в предложении и словосочетании с помощью формообразующей морфемы в отличие от английского языка, где слова меньше подвержены изменениям, а связь слов в предложении зависит в большей степени от места, а не от формы самого слова.

Первая стадия исследования заключалась в выявлении семантического поля слов и анализе полученных результатов для русского и английского языков.

Для этой цели были созданы два корпуса из текстов средств массовой информации, один корпус англоязычный, второй – русскоязычный, по 20 000 слов в каждом за период с марта по апрель 2021 г. Часть исследования основывается на корпусе, собранном в апреле–мае 2020 г. Для корректного представления в терминале файлы были преобразованы в кодировку UTF-8 и обработаны модулем PlaintextCorpusReader из ресурса NLTK.

```
>>> direct= 'D:/rus_corpus'
>>> corpus=PlaintextCorpusReader(direct, '.*')
```

Далее корпус был поделен на токены. В зависимости от поставленной цели токены могут представлять собой лингвистическую единицу в виде алфавитных символов, слов, предложений и даже целого текста. На этом этапе токенами стали слова и знаки препинания.

```
>>> corpus_words= corpus.words()
>>> corpus_words[:50]
```

```
['Минтранс', 'отрицает', 'подорожание', 'авиабилетов', '•', '•', '10', '•', '•', '•', '•', '•', 'Минтранс', 'России', 'не', 'подтвердил', 'информацию', 'о', 'росте', 'цен', 'на', 'авиабилеты', '•', 'о', 'котором', 'ранее', 'сообщили', '«', 'Известия', '»', 'со', 'ссылкой', 'на', 'данные', 'агрегаторов'...]
```

Семантическое поле слова выявлялось на основе n-грамм, т.е. контекста, поэтому для получения более точных результатов токены в виде пунктуации были удалены, кроме того, весь текст был преобразован в строчный, потому что программа капитализированные и строчные слова рассматривает как разные, что значительно влияет на полученные показатели.

```
>>> corpuslower=[w.lower() for w in corpus_words if w.isalpha()]
>>> corpuslower[:50]
['минтранс', 'отрицает', 'подорожание', 'авиабилетов', 'минтранс', 'россии', 'не', 'подтвердил', 'информацию', 'о', 'росте', 'цен', 'на', 'авиабилеты', 'о', 'котором', 'ранее', 'сообщили', 'известия', 'со', 'ссылкой', 'на', 'данные', 'агрегаторов', 'для', 'путешественников', 'пресс', 'служба', 'мини-
```

```
стерства', 'уточнила', 'что', 'по', 'некоторым', 'направлениям', 'цены', 'на', 'билеты', 'снизились', 'информация', 'представленная', 'рядом', 'агрегаторов', 'о', 'якобы', 'росте', 'цен', 'на', 'авиабилеты', 'по', 'сравнению']
```

И далее полученный корпус был отформатирован на уровне текста и была применена команда выявления семантики.

```
>>> simtext=nltk.Text(corpuslower)
>>> simtext=nltk.Text(corpuslower)
>>> simtext.similar('суэцкий')
```

году через весь перегородив судоходный проходя

Как видно из полученного результата, «суэцкий» ассоциируется со словами *весь, перегородив, судоходный, проходя*. Однако можно предположить, что эти данные не полные, т.к. система рассматривает все склонения слова «суэцкий», напр. «суэцкого», не как другие формы слова, а как другие слова. Правомочность такого предположения была выявлена через лемматизацию. Лемма (каноническая форма слова) и определения ее грамматических характеристик представлены в книге В. В. Бочарова, О. В. Митренина «Компьютерная морфология».

```
>>> morph = pymorphy2.MorphAnalyzer()
>>> digits = [str(i) for i in range(10)]
>>> def preprocess(tokens):
    return [morph.normal_forms(word)[0]
            for word in tokens
            if (word[0] not in digits)]
```

```
>>> gr = preprocess(corpus_words)
```

После лемматизации корпус приобрел следующий вид

```
>>> gr[:50]
['минтранс', 'отрицать', 'подорожание', 'авиабилет', 'минтранс', 'россия', 'не', 'подтвердить', 'информация', 'о', 'рост', 'цена', 'на', 'авиабилет', 'о', 'который', 'ранее', 'сообщить', 'известие', 'с', 'ссылка', 'на', 'дать', 'агрегатор', 'для', 'путешественник', 'пресс', 'служба', 'министерство', 'уточнить', 'что', 'по', 'некоторый', 'направление', 'цена', 'на', 'билет', 'снизиться', 'информация', 'представить', 'рядом', 'агрегатор', 'о', 'якобы', 'рост', 'цена', 'на', 'авиабилет', 'по', 'сравнение']
```

Теперь команду выявления семантического поля слова «суэцкий» применим к лемматизированному корпусу.

```
grtext.similar('Суэцкий')
```

*по администрация блокировка управление через год весь сайт от они
проходить сам telegram канал ежедневно
движение если эфир открытие
история*

Полученный результат показывает правильность такого подхода для флективных языков, он стал точнее и полнее. Если в оригинальном тексте слово ассоциировалось с шестью словами, то после обработки видно, что их стало двадцать.

Для аналитического английского языка такая проблема не стоит в связи с тем, что «Suez» практически не склоняется:

```
>>> corp_eng.similar('suez')
canal vessel waterway blockage crew world incident ships company sca
delays boat bbc operation authority stern claim decision bow route
```

Видно, что результаты вывода полные.

Еще одним способом решения проблем с флективными языками при выявлении семантического поля слова может служить стемминг. В рамках данной работы такое исследование тоже было проведено.

После применения команды стеммирования русский корпус приобрел следующий вид:

```
>>> from nltk.stem.snowball import SnowballStemmer as sn
>>> stemmer=sn('russian')
>>> rustem=[stemmer.stem(w) for w in gr2]
>>> rustem[:50]
>>> ['минтранс', 'отрица', 'подорожан', 'авиабилет', 'минтранс', 'росс', 'не', 'подтверд', 'информац', 'о', 'рост', 'цен', 'на', 'авиабилет', 'о', 'котор', 'ран', 'сообщ', 'извест', 'с', 'ссылк', 'на', 'дат', 'агрегатор', 'для', 'путешественник', 'пресс', 'служб', 'министерств', 'уточн', 'что', 'по', 'некотор', 'направлен', 'цен', 'на', 'билет', 'сниз', 'информац', 'представ', 'ряд', 'агрегатор', 'о', 'якоб', 'рост', 'цен', 'на', 'авиабилет', 'по', 'сравнен']
```

Система максимально «попыталась» оставить основу слов, хотя не всегда удачно. Такая обработка подходит для данного исследования, потому что она практически не содержит аффиксов.

Применим вышеуказанную задачу с выявлением семантического поля слова к стеммированному корпусу. Для начала было выявлено, как стеммировалось слово «суэцкий»

```
>>> rustemt.concordance('суэцк')
Displaying 25 of 357 matches:
даж посл окончан конституцион реформ суэцк
ер от блокировк движен глав управлен суэцк
кана усам раб оцен финансов потер от
en group март сест на мел в южн част суэцк
кана котор не имет дублер он следова
рс из за сильн ветер март навигац на суэцк
кана возобнов пентагон ответ на сооб
```

И применена команда выявления семантики.

```
>>> rustemt.similar('суэцк')
```

No matches

```
>>> rustemt.similar('суэцк кана')
```

No matches

Видно, что вывод получился отрицательный. Но если преобразование текста в строчный вид выполнить отдельной командой, задача облегчается:

```
>>> rustemlow=[w.lower() for w in rustem]
>>> rustext=nltk.Text(rustemlow)
```

```
>>> rustext.concordance('суэцк')
```

Displaying 9 of 9 matches:

```
орт: порт - сауд из средиземн мор и суэцк из
красн мор. суэцк кана
в район озер тимс , зат был продл к суэцк при
фараон нех ii окол 600 до
танкер из новороссийск проход через суэцк ,
например "газпр нефт " ,
северн вход в кана ) и в 04 : 00 из суэцк ( южн
вход ). корабл должн
4 год кана был заблокирова три дня . суэцк
откр : судн evergiven сдвинул
и положен поперек кана , но оператор суэцк
gas заявля , что ег подогна к
ньск порт , конечн пункт указа город суэцк .
контейнеровоз evergiven
сауд ( северн вход ) , в 4:00 – из суэцк ( южн
вход ). корабл должн
з сам узк мест – ближ к портов город суэцк .
имен в эт точк судн сбил с
```

Результаты вывода показывают, что после стеммирования «Суэцкий» получило форму «суэцк». Применим к нему команду выявления семантического поля.

```
>>> rustext.similar('суэц')
в призва вышл хот великобритан вариант
япон состоя издел осуд вод
нефтепродукт concentraц пакиста
```

Получился довольно неожиданный вывод, появились новые слова, которых не было в оригинальном и лемматизированном корпусе.

Из этого можно сделать вывод, что для выявления семантического поля имеет смысл комбинировать все три подхода к решению этой задачи.

Полученные результаты также позволяют говорить о том, что для выявления семантического поля метод стеммирования для высоко флективных языков не подходит, несмотря на то, что он оставляет только основы слова.

Та же команда, примененная к стеммированному корпусу аналитического английского языка, выдает более качественный результат:

```
>>> from nltk.stem.snowball import SnowballStemmer as sn
>>> stemmer=sn('english')
>>> enstem=[stemmer.stem(w) for w in entext]
>>> enstem[1000:1050]
['increas', 'covid', 'vaccin', '.', 'he', 'said', '65', '%', 'of', 'passeng', 'who', 'flew', 'delta', 'in', '2019', 'expect', 'to', 'have', 'at', 'least', 'one', 'dose', 'of', 'the', 'vaccin', 'by', 'may', 'I', 'throughout', 'the', 'pandem', 'vaccin', 'rate', 'and', 'consum', 'behavior', 'have', 'been', 'delta', 's', 'benchmark', 'on', 'whether', 'to', 'keep', 'block', 'seat']
>>> sn_entext=nltk.Text(enstem)
>>> sn_entext.similar('suez')
canal ship world vessel govern waterway time us
countri case crisi eu way past rule author
```

Результаты исследования показывают, что обработка естественного языка для машинного перевода на данном этапе не может быть унифицирована. Также нет оснований полагать, что решения, представленные в рамках данной статьи для русского языка, будут эффективны для других высокофлексивных языков. Частично это зависит от моделей и алгоритмов лемматизатора и стеммера для конкретного языка. Есть основание полагать, что «на данном этапе развития нейронных сетей в переводе более эффективным представляется их комбинирование со статистическими системами» [5].

Д. А. Усталов в работе «Модели, методы и алгоритмы построения семантической сети слов для задач обработки естественного языка» [6] отмечает, что общей проблемой метода построения связей на основе лексико-синтаксических шаблонов является разреженность данных. Например, если пара слов (кот, животное) присутствует в коллекции документов, а пара слов (кошка, животное) нет, то не существует возможности восстановить связь между словами, близкими по значению, без использования внешних семантических ресурсов.

Среди других вариантов решения этой проблемы может быть использование специальных тегов, они позволяют извлечь достоверные связи в предложении и словосочетании, но разработка таких шаблонов трудоемка и экономически не эффективна.

Такого рода проблемы приводят к созданию новых технических решений, которые особенно актуальны для таких систем как «недоученные нейронные сети» [4]. Одним из них является представление слова в нейронных сетях. Это тот же метод, т.е. метод оценки соседствующих с основным словом n-грам позволяет представить семантическое поле слова в цифровом виде. Такие методы применяются сегодня в нейронных машинных переводчиках. Алгоритм работает по принципу: «Скажи мне, кто твой друг, и я скажу, кто ты». Слова, с которыми лексема встречается в предложениях корпуса, позволяют «вычислить» семантику слова, классифицировать его по жанру, автоматически тегировать часть речи и многое другое.

Если представить, что есть система координат с отрезком от -1 до 1 и, к примеру, описать размеры некоторых животных, что позволяет сравнивать «с помощью матриц репрезентативных расстояний» [8].

Рис. 1.

Но звери отличаются не только размером. Добавим степень агрессивности.

Рис. 2.

Таким образом, в нейронной системе на основе корпусных данных формируются многомерные вектора – 1 000-мерные и более, которые представляют собой значение слова. В представленном выше рис. 1 показана двумерная модель, и даже с такими минимальными данными можно сравнить данные, например, медведя и лису, через косинусы углов между векторами.

Найдем скалярное произведение векторов:

$$\begin{aligned} \vec{a} \cdot \vec{b} &= a_x \cdot b_x + a_y \cdot b_y + a_z \cdot b_z = \\ &= 0.5 \cdot 0.8 + 0.4 \cdot 0.7 + 0 \cdot 0 = 0.4 + 0.28 + 0 = \\ &= 0.68 \end{aligned}$$

Вычислим длины векторов:

$$\begin{aligned} |\vec{a}| &= \sqrt{a_x^2 + a_y^2 + a_z^2} = \sqrt{(0.5)^2 + (0.4)^2 + 0^2} = \\ &= \sqrt{0.25 + 0.16 + 0} = \sqrt{0.41} = \sqrt{41}/10 \end{aligned}$$

$$\begin{aligned} |\vec{b}| &= \sqrt{b_x^2 + b_y^2 + b_z^2} = \sqrt{(0.8)^2 + (0.7)^2 + 0^2} = \\ &= \sqrt{0.64 + 0.49 + 0} = \sqrt{1.13} = \sqrt{113}/10 \end{aligned}$$

Найдем угол между векторами:

$$\cos \alpha = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|}$$

$$\cos \alpha = \frac{0.68}{\sqrt{41}/10 \cdot \sqrt{113}/10}$$

$$= 0.999028234937562$$

Таким образом, центральной идеей в нейронных сетях является то, что можно представить семантику слов в виде векторов в многомерной системе координат, посчитать и сравнить, насколько они синонимичны или антонимичны, являются ли омонимами, гиперонимами или меронимами и др.

Так, согласно источнику <https://jalammar.github.io/illustrated-word2vec/>, координаты векторов для слова “king” равны:

[0.50451, 0.68607, -0.59517, -0.022801, 0.60046, -0.13498, -0.08813, 0.47377, -0.61798, -0.31012, -0.076666, 1.493, -0.034189, -0.98173, 0.68229, 0.81722, -0.51874, -0.31503, -0.55809, 0.66421, 0.1961, -0.13495, -0.11476, -0.30344, 0.41177, -2.223, -1.0756, -1.0783, -0.34354, 0.33505, 1.9927, -0.04234, -0.64319, 0.71125, 0.49159, 0.16754, 0.34344, -0.25663, -0.8523, 0.1661, 0.40102, 1.1685, -1.0137, -0.21585, -0.15155, 0.78321, -0.91241, -1.6106, -0.64426, -0.51042].

Если пересчитать косинусы векторов для слов “king”, “man”, “woman” и выполнить следующее

вычисление “king” - “man” + “woman”, получается удивительный результат (из вышеупомянутого источника):

```
model.most_similar(positive=["king", "woman"], negative=["man"])
[('queen', 0.8523603677749634),
 ('throne', 0.7664333581924438),
 ('prince', 0.7592144012451172),
 ('daughter', 0.7473883032798767),
 ('elizabeth', 0.7460219860076904),
 ('princess', 0.7424570322036743),
 ('kingdom', 0.7337411642074585),
 ('monarch', 0.721449077129364),
 ('eldest', 0.7184862494468689),
 ('widow', 0.7099430561065674)]
```

С помощью той же команды “similar” система выводит “queen”.

Таким образом, простая нейронная модель качественно воспроизводит множество разнообразных закономерностей, лежащих в основе семантического поля, обеспечивая при этом аналитическое понимание того, как статистическая структура среды может взаимодействовать с нелинейной динамикой глубокого обучения, порождая эти закономерности. В рамках данной статьи рассматривались не только задачи определения семантического поля слова, но и технологии и алгоритмы обработки естественного языка, способные решать и такие задачи, как выделение синонимов, ключевых слов, выявление морфологических и синтаксических характеристик, выявление модальности и многих других лингвистических свойств.

Отметим при этом, что качество полученного результата существенно зависит от объема корпусных данных.

ЛИТЕРАТУРА

1. Богомолов Ю. А. Обзор моделей нейронных сетей для обработки естественного языка / Научно-образовательный журнал для студентов и преподавателей «StudNet». Т. 3, №4. 2020. С. 203-217.
2. Бочаров В. В., Митренина О. В. Компьютерная морфология // Прикладная и компьютерная. Изд. 2-е. М.: ЛЕ-НАНД, 2017. 37 с.
3. Мифтахова Р. Г. Выявление оттенков смысла в текстах СМИ// Языки в диалоге культур: Проблемы многоязычия в полиэтническом пространстве. Уфа: РИЦ БашГУ, 2020. С. 300-303.
4. Моисеев И. В., Жуков Л. А. Решение лингвистических задач с помощью нейронных сетей с учителем на примере задачи о распознавании слов прилагательных английского языка. URL: <https://cyberleninka.ru/article/n/reshenie-lingvisticheskikh-zadach-s-pomoschyu-neyronnyh-setey-s-uchitelem-na-primere-zadachi-o-raspoznavanii-slov-prilagatelnyh/viewer>
5. Морозкина Е. А., Мифтахова Р. Г. Машинный перевод. Нейроперевод // Вестник Башкирского университета. 2019. Т. 24. №2. С. 497-502.
6. Усталов Д. А. Модели, методы и алгоритмы построения семантической сети слов для задач обработки естественного языка: автореф. дис. ... канд. физ.-мат. наук. Екатеринбург, 2017 г.
7. Черезов Д. С., Тюкачев Н. А. Применение нейронных сетей в процессе индексирования и ранжирования текста // Вестник ВГУ, серия: Системный анализ и информационные технологии, 2007, №2. С. 65-68.
8. Kriegeskorte N., Rogier A. Kievit Representational geometry: Integrating cognition, computation, and the brain. Trends in Cognitive Sciences. August 2013, Vol. 17, No.8. Pp. 401-412.

Поступила в редакцию 21.11.2021 г.

DOI: 10.33184/bulletin-bsu-2021.4.48

NEURAL REPRESENTATION OF A SEMANTIC FIELD

© R. G. Miftakhova*, E. A. Morozkina

*Bashkir State University
19 Kommunisticheskaya Street, 450076 Ufa, Republic of Bashkortostan, Russia.**Phone: +7 (927) 332 66 51.***Email: miftahovar@yandex.ru*

An important problem in natural language processing is ambiguity, which requires the system to understand the context and the subject area in which the word is used. Such information can be presented in the so-called semantic network used in neural machine translation. These are special databases that provide information about vocabulary in the form of concepts and relationships between them. This information forms a semantic hierarchy. In the field of natural language processing, the most popular semantic network for the English language today is WordNet. It is widely used in machine translation systems. Creating such tagged databases for other languages is a long and time-consuming process; therefore, scientists pay great attention to the automatic construction of semantic networks. The authors of the paper present algorithms of text processing in highly inflectional Russian for identification of the semantic field of a word. A comparative analysis of the applied algorithms for analytical English is also carried out.

Keywords: natural language processing, semantic field, models and algorithms, inflectional and analytical languages.

Published in Russian. Do not hesitate to contact us at bulletin_bsu@mail.ru if you need translation of the article.

REFERENCES

1. Bogomolov Yu. A. Obzor modelei neironnykh setei dlya obrabotki estestvennogo yazyka / Nauchno-obrazovatel'nyi zhurnal dlya studentov i predavatelei «StudNet». Vol. 3, No. 4. 2020. Pp. 203-217.
2. Bocharov V. V., Mitrenina O. V. Prikadnaya i komp'yuternaya. Izd. 2-e. Moscow: LENAND, 2017.
3. Miftakhova R. G. Yazyki v dialoge kul'tur: Problemy mnogoyazychiya v polietnicheskom prostranstve. Ufa: RITs BashGU, 2020. Pp. 300–303.
4. Moiseev I. V., Zhukov L. A. Reshenie lingvisticheskikh zadach s pomoshch'yu neironnykh setei s uchitelem na primere zadachi o raspoznavanii slov prilagatel'nykh angliiskogo yazyka. URL: <https://cyberleninka.ru/article/n/reshenie-lingvisticheskikh-zadach-s-pomoschyu-neironnykh-setey-s-uchitelem-na-primere-zadachi-o-raspoznavanii-slov-prilagatelnyh/viewer>
5. Morozkina E. A., Miftakhova R. G. Vestnik Bashkirskogo universiteta. 2019. Vol. 24. No. 2. Pp. 497–502.
6. Ustalov D. A. Modeli, metody i algoritmy postroeniya semanticheskoi seti slov dlya zadach obrabotki estestvennogo yazyka: avtoref. dis. ... kand. fiz.-mat. nauk. Ekaterinburg, 2017 g.
7. Cherezov D. S., Tyukachev N. A. Vestnik VGU, seriya: Sistemnyi analiz i informatsionnye tekhnologii, 2007, No. 2. Pp. 65–68.
8. Kriegeskorte N., Rogier A. Kievit Representational geometry: Integrating cognition, computation, and the brain. Trends in Cognitive Sciences. August 2013, Vol. 17, No.8. Pp. 401–412.

Received 21.11.2021.