УДК 004.85 **DOI: 10.34670/AR.2023.60.95.001**

Разработка алгоритма семантического анализа для определения жанра произведения

Алиев Ризван Идрисович

Ассистент кафедры «Бизнес-информатика»,
Чеченский государственный университет им. А.А. Кадырова,
364093, Российская Федерация, Грозный, ул. Асланбека Шерипова, 32;
e-mail: Chibo18@mail.ru

Максимович Андрей Владимирович

Генеральный директор,
ООО «АППС5 СТУДИО»,
214012, Российская Федерация, Смоленск, ул. 2-я Садовая, 25а;
e-mail: a.maksimov@aplatforma.ru

Строкин Никита Алексеевич

Студент,
Национальный исследовательский университет «Высшая школа экономики»,
101000, Российская Федерация, Москва, ул. Мясницкая, 20;
e-mail: n.strokin@aplatforma.ru

Аннотация

На сегодняшний день задача семантического анализа и кластеризации решается путем сравнения либо векторного представления корпуса текста целиком. Также возможен вариант с выявлением одинаковых последовательностей векторов слов: корпусы текстов преобразуются в векторы, после чего сравниваются между собой. Статья посвящена исследованию и разработке алгоритма семантического анализа литературных произведений при помощи искусственного интеллекта (далее – ИИ) и машинного обучения. В статье представлены не только теоретическая часть разработки данного алгоритма, но и фрагменты кода, которые были написаны в процессе разработки системы. Разработанный алгоритм предполагает кластеризацию текстовых произведений по жанру. Сравниваемые корпуса текстов в данном случае имеют большую длину; для подобного рода задач на сегодняшний день это вызывает проблему данных высокой размерности. Для преодоления данного недостатка разработан алгоритм обучения модели основного словаря: модель обучается на произведениях определенных жанров, что позволяет ограничить размерность данных в заданных пределах. Алгоритм семантического анализа произведений создан для определения жанра произведения с целью поиска подходящих поставщиков услуг, которые специализируются в определенных жанрах. В разработке данного алгоритма использованы технологии обработки естественного языка (Natural Language Processing, далее – NLP) для распознавания письменной человеческой речи.

Данная технология является направлением в машинном обучении и находится в соприкосновении таких дисциплин, как лингвистика и ИИ.

Для цитирования в научных исследованиях

Алиев Р.И., Максимович А.В., Строкин Н.А. Разработка алгоритма семантического анализа для определения жанра произведения // Психология. Историко-критические обзоры и современные исследования. 2023. Т. 12. № 5А-6А. С. 229-236. DOI: 10.34670/AR.2023.60.95.001

Ключевые слова

Семантический анализ, искусственный интеллект, машинное обучение, алгоритм, нейросети, векторы, классификация текстов, кластеризация, метрики вычислений.

Введение

На сегодняшний день задача семантического анализа и кластеризации решается путем сравнения либо векторного представления корпуса текста целиком. Также возможен вариант с выявлением одинаковых последовательностей векторов слов: корпусы текстов преобразуются в векторы, после чего сравниваются между собой.

Основным недостатком данного подхода является отсутствие семантических связей между словами после их преобразования, что затрудняет анализ коротких корпусов, где близкие по смыслу тексты выражены разной лексикой. Также существует проблема наличия данных высокой размерности, особенно это становится заметно, когда работа ведется с корпусами текстов большой длины: в подобных данных разница между двумя точками может быть незначительной, и многие алгоритмы, которые используют метрики на основе расстояний между объектами, теряют свою эффективность и требуют больше ресурсов для вычислений [Radlinski et al., 2009; Киселев, 2012].

Разработанный алгоритм предполагает кластеризацию текстовых произведений по жанру. Сравниваемые корпуса текстов в данном случае имеют большую длину; для подобного рода задач на сегодняшний день это вызывает проблему данных высокой размерности.

Для преодоления данного недостатка разработан алгоритм обучения модели основного словаря: модель обучается на произведениях определенных жанров, что позволяет ограничить размерность данных в заданных пределах [Omar, 2020; Серов и др., 2015].

Логика работы разработанного алгоритма

Подаваемые на вход произведения проходят предварительное преобразование, которое включает в себя следующие действия:

– первичный анализ: определяются границы заголовков, абзацев и отдельных предложений, с которыми в дальнейшем будет вестись работа алгоритма; так как для определения жанра заголовки не важны, они не учитываются при последующем анализе (планируется на втором этапе);
– с помощью библиотеки Word2vec слова в корпусе текста преобразуются в начальную форму;
– далее производится токенизация (каждому слову присваивается токен);

– с целью увеличения производительности алгоритма, из дальнейшего анализа на этапе подготовки текста исключаются т.н. «стоп-слова» (междометия, союзы, частицы, предлоги, знаки препинания и др.) [Киселев, 2012; Серов и др., 2015].

На «Рис. 1» показан начальный элемент кода, реализующего алгоритм предварительного преобразования текста.

```
textCl = gensim.utils.simple_preprocess(text)
text = ' '.join(textCl)
text = m.lemmatize(text)
with open(dirpathResult+'/'+fileName+'_lemma.txt', "w") as output_file:
    output_file.write(''.join(text))

text_sentences = []
for word in text:
    if word != ' ' and word != '\n':
        word_sentences = []
        word_sentences.append(word)
        text_sentences.append(word_sentences)

common_texts = text_sentences
bigram_transformer = Phrases(common_texts)
model = Word2Vec(bigram_transformer[common_texts], min_count=2)
model.wv.save_word2vec_format(dirpathResult+'/'+fileName+'_vectors_word2vec_format.kv', fvocab=None, binary=False)
```

Рисунок 1 - Процесс предподготовки корпуса текста

Основной этап. Выделяются главные герои или события, по каждому из них выбирается ограниченное количество семантически близких к ним слов; данные слова используются для создания обучаемой модели алгоритма, которая предназначена для классификации произведений [там же].

На «Рис. 2» показан элемент кода, реализующего алгоритм выбора близких слов.

```
if hero1 != '':
    file = open(dirpathResult+'/'+fileName+'_context.txt', "w")
    file.write('Для '+hero1+':\n')
    for i in model.wv.most_similar(positive=[hero1], topn=300):
        file.write(str(i))
        file.write('\n')
        wordsContextAllStories.append(i[0])
    file.close()

if hero2 != '':
    file = open(dirpathResult+'/'+fileName+'_context.txt', "a")
    file.write('\nДля '+hero2+':\n')

    try:
        wordsContext = model.wv.most_similar(positive=[hero2], topn=300)
        for i in wordsContext:
            file.write(str(i))
            file.write('\n')
            wordsContextAllStories.append(i[0])
    except BaseException as e:
        print("Error BaseException..."+fileName+"....................." + str(e))

    file.close()
```

Рисунок 2 - Процесс выбора близких слов

Выборка слов основного словаря позволяет ограничить размерность данных в заданных пределах, не теряя основной семантики произведения.

Обучаемая модель алгоритма – это нейросеть, создаваемая с использованием предварительно обученных слов, которая имеет следующие слои:

– слой внедрения, который инициализируется со случайными весами и изучает встраивание для всех слов в наборе обучающих данных;
– подготовительный слой, который уменьшает размерность выходных данных первого слоя, пытаясь выделить важные элементы;
– плотный слой с функцией активации relu, которая добавляет нелинейность, превращая отрицательные числа в 0;
– плотный выходной слой с тремя нейронами, соответственно количеству классов, с функцией активации softmax, благодаря которой сумма вероятностей будет равна 1.

На «Рис. 3» показан элемент кода, реализующего процесс создания модели нейронной сети.

```
filters='!"#$%&()*+,-./:;<=>?@[\\]^_`{|}~\t\n'
tokenizer = Tokenizer(num_words=5000)
tokenizer.fit_on_texts(keyWordsArr)
X_train = tokenizer.texts_to_sequences(sentences_train_arr)
vocab_size = len(tokenizer.word_index) + 1
maxlen = 3000
X_train = pad_sequences(X_train, padding='post', maxlen=maxlen)

embedding_dim = 2000
model = Sequential()
model.add(layers.Embedding(input_dim=vocab_size, output_dim=embedding_dim, input_length=maxlen))
model.add(layers.GlobalMaxPooling1D())
model.add(layers.Dense(10, activation='relu'))
model.add(layers.Dense(3, activation='softmax'))
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])
model.summary()
```

Рисунок 3 - Процесс создания модели нейронной сети

Инновационность предлагаемого решения

Инновационным является предложение использовать для решения задачи обучения нейросети жанровой классификации текстов. На первом этапе обучения производится подбор текстов одного жанра (например, детектив) с единым сквозным (присутствующих во всех произведениях подборки) героем, который определяется как именованная сущность. Нейросеть выбирает слова (леммы), относящиеся грамматически и семантически к данной именованной сущности, среди которых существуют маркированные слова, относящиеся к исследуемому жанру, и помещает их в список. Следующим шагом этот список очищается от слов-дубликатов, которые попали в него в ходе работы алгоритма. Предполагается возможность вычисления веса слов, попавших в список, на основе тегов, присвоенных при предварительной обработке текста [Radlinski et al., 2009; Omar, 2020; Николаев, 2022].

Далее этот список может сравниваться с «идеальным словарем» по исследуемому жанру с целью определения нейросетью представленного произведения как относящегося к конкретному жанру на основе найденных маркированных слов. Этап с идеальным словарем в случае необходимости может опускаться. Предполагается, что с помощью обучения на одной именованной сущности, находящейся в подборке текстов, алгоритм сможет наиболее

эффективно определять жанровую принадлежность текста благодаря тому, что многие маркированные слова находятся близко к именованным сущностям. Для лучшего определения того, что является именованной сущностью, легче учить нейросеть на одном повторяющемся примере (в каждой отдельной подборке текстов) с последующей экстраполяцией на другие произведения на следующем этапе [Мочалова, 2014].

После завершения первого этапа обучения, на втором этапе предлагается использовать подборки текстов одного жанра, но уже без единого героя. Задачей нейросети будет являться самостоятельное нахождение корректных именованных сущностей и определение грамматически и семантически относящихся к данной именованной сущности слов и занесение их в список так же, как и на предыдущем этапе, с последующим удалением слов-дубликатов.

Преимущества предлагаемого алгоритма по сравнению с аналогами

Данный алгоритм обучения нейросети позволит экономить время и ресурсы для решения проблемы жанровой классификации текста, так как в ходе выполнения программы алгоритма возможно использовать относительно небольшие подборки текстов (в среднем от 20 до 80). Предполагается, что по сравнению с другими методиками, в которых необходимо использовать большое количество текстов (от 500 и более), данная методика будет не только менее ресурсоемкой, но и более эффективной.

В предлагаемой методике не требуется работа со всем текстом, а только с именованной сущностью, которая указывается заранее и встречается в тексте достаточно часто, чтобы обучить алгоритм, но недостаточно часто, чтобы сильно нагрузить систему. Это позволит снизить нагрузку на систему, выполняющую алгоритм, и позволит реализовать данный алгоритм на менее производительных системах [Николаев, 2022; Лапшин, Лебедев, Спивак, 2019].

Заключение

Для тестирования использовались произведения авторов: А.К. Дойл, Ж. Сименон, А. Кристи, Р. Стаут, Н. Леонов, Д. Толкин, Р. Джордан, А. Сапковский, С. Лем, Р. Хайнлайн, Р. Шекли, А. и Б. Стругацкие, И. Ефремов и др.

Вся совокупность произведений была разделена по трем жанрам: «детектив», «фэнтези», «научная фантастика», а также на обучающий и тестовый набор в отношении 80% и 20% соответственно.

Общее количество произведений для обучения на первом этапе составило более 600, примерно по 200 для каждого жанра. Разброс по объему между жанрами – не более 10%; произведения имеют два типа: «рассказ» (менее 100 000 знаков с пробелами), «роман» (более 100 000 знаков с пробелами). Отдельные произведения были приведены в формат второго типа («роман») путем разделения на части, с сохранением выбранной именованной сущности в каждой части [Shajalal et al., 2022; Liu, 2022].

Первым этапом был создан основной словарь по всем произведениям, затем на базе выбранного словаря была создана модель нейронной сети с встраиванием обученных слов.

Ожидаемые требования по точности к данному набору произведений были не ниже 80%. По результатам теста точность при обучении составила порядка 82%.

На «Рис. 4» результаты тестирования представлены графически, по оси Y – точность, по оси X – количество эпох.

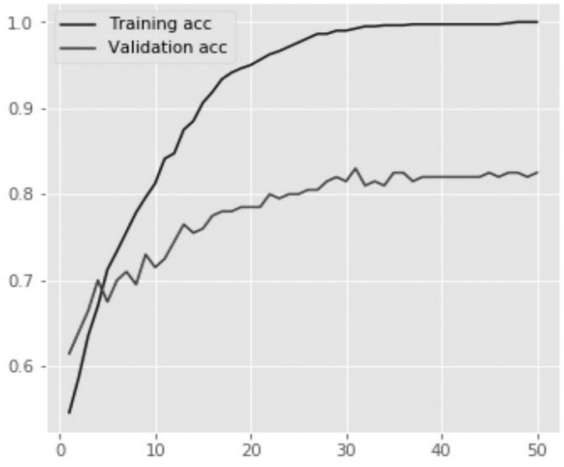

Рисунок 4 - Результат теста

Training accuracy – результат при обучении сети,

Validation accuracy – результат при проверке после обучения.

Инструменты разработки

Для разработки алгоритма семантического анализа и АСАП в целом используются среда разработки на базе Docker, редакторы кода ATOM и VSCode, база данных MySQL, для алгоритма кластеризации язык программирования Python и библиотеки Keras, Word2Wec, компьютеры iMac.

Для обучения использовался компьютер с параметрами:

– iMac 21,2

– Apple M1

– 8 ядер

– 8Гб память

Библиография

1. Батраева И.В., Нарцев А.Д., Лезгян А.С. Использование анализа семантической близости слов при решении задачи определения жанровой принадлежности текстов методами глубокого обучения // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2020. № 50. С. 14-22.
2. Киселев Ю.А. Перспективы использования жанровой классификации Веб документов в поисковых системах // Инженерный вестник Дона. 2012. № 4. URL: ivdon.ru/ru/magazine/archive/n4p2y2012/1425/
3. Лапшин С.В., Лебедев И.С., Спивак А.И. Кластеризация текстов с использованием семантико-синтаксических связей слов // Научно-технический вестник информационных технологий, механики и оптики. 2019. № 6. С. 1058-1063.
4. Мочалова А.С. Алгоритм семантического анализа текста, основанный на базовых семантических шаблонах с удалением // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 5 (93). С. 126-132.
5. Николаев П.Л. Классификация книг по жанрам на основе текстовых описаний посредством глубокого обучения // International Journal of Open Information Technologies. 2022. № 1. С. 36-40.
6. Серов С.С. и др. Сокращение времени оценки схожести текстовых документов на неоднородной многопроцессорной вычислительной системе // Инженерный вестник Дона. 2015. № 2. URL: ivdon.ru/ru/magazine/archive/n2p2y2015/3031/
7. Liu M. et al. Short Text Dynamic Clustering Approach for Semantic-Enhanced Knowledge // ICCSE 2022.

Communications in Computer and Information Science. 2022. Vol. 1811.
8. Omar A. Classifying letarary genre: a methodological synergy of computational modelling and lexical semantics // Classificação de gêneros literários: uma sinergia metodológica de modelagem computacional e semântica lexical. Texto Livre: Linguagem e Tecnologia. 2020. Vol. 13. No. 2. P. 83-101.
9. Radlinski F. et al. Redundancy, diversity and interdependent document relevance // Newsletter ACM SIGIR Forum. 2009. Vol. 43. Issue 2. P. 46-52.
10. Shajalal M. et al. Textual Entailment Recognition with Semantic Features from Empirical Text Representation // SPELLL 2022. Communications in Computer and Information Science. 2022. Vol. 1802.

Development of a semantic analysis algorithm for determining the genre of a work

Rizvan I. Aliev

Assistant of the Department of Business Informatics,
Kadyrov Chechen State University,
364049, 32, Sheripova str., Grozny, Russian Federation;
e-mail: Chibo18@mail.ru

Andrei V. Maksimovich

CEO of APPS5 STUDIO LLC,
214012, 25a, Vtoraya Sadovaya str., Smolensk, Russian Federation;
email: a.maksimov@aplatforma.ru

Nikita A. Strokin

Student,
Higher School of Economics – National Research University,
101000, 20, Myasnitskaya str., Moscow, Russian Federation;
e-mail: n.strokin@aplatforma.ru

Abstract

To date, the problem of semantic analysis and clustering is solved by comparing or vector representation of the entire text corpus. It is also possible to identify identical sequences of word vectors: text corpora are converted into vectors, after which they are compared with each other. The article is devoted to the research and development of an algorithm for the semantic analysis of literary works using artificial intelligence (hereinafter referred to as AI) and machine learning. The article presents not only the theoretical part of the development of this algorithm, but also code fragments that were written during the development of the system. The developed algorithm assumes clustering of text works by genre. The compared corpora of texts in this case are longer; for this kind of task today, this causes a problem of high-dimensional data. To overcome this shortcoming, an algorithm for learning the main dictionary model has been developed: the model is trained on works of certain genres, which makes it possible to limit the data dimension within the specified limits. The semantic analysis algorithm for works is designed to determine the genre of a work in order to find suitable service providers who specialize in certain genres. In the development of this algorithm,

natural language processing technologies (Natural Language Processing, hereinafter referred to as NLP) were used to recognize written human speech. This technology is a direction in machine learning and is in contact with disciplines such as linguistics and AI.

For citation

Aliev R.I., Maksimovich A.V., Strokin N.A. (2023) Razrabotka algoritma semanticheskogo analiza dlya opredeleniya zhanra proizvedeniya [Development of a semantic analysis algorithm for determining the genre of a work]. *Psikhologiya. Istoriko-kriticheskie obzory i sovremennye issledovaniya* [Psychology. Historical-critical Reviews and Current Researches], 12 (5A-6A), pp. 229-236. DOI: 10.34670/AR.2023.60.95.001

Keywords

Semantic analysis, artificial intelligence, machine learning, algorithms, neural networks, vectors, text classification, clustering, computing metrics.

References

1. Batraeva I.V., Nartsev A.D., Lezgyan A.S. (2020) Ispol'zovanie analiza semanticheskoi blizosti slov pri reshenii zadachi opredeleniya zhanrovoi prinadlezhnosti tekstov metodami glubokogo obucheniya [Using the analysis of the semantic similarity of words in solving the problem of determining the genre of texts by deep learning methods]. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika* [Bulletin of the Tomsk State University. Management, computer technology and informatics], 50, pp. 14-22.
2. Kiselev Yu.A. (2012) Perspektivy ispol'zovaniya zhanrovoi klassifikatsii Veb dokumentov v poiskovykh sistemakh [Prospects for the use of the genre classification of Web documents in search engines]. *Inzhenernyi vestnik Dona* [Don Engineering Bulletin], 4. Available at: ivdon.ru/ru/magazine/archive/n4p2y2012/1425/ [Accessed 05/05/2023]
3. Lapshin S.V., Lebedev I.S., Spivak A.I. (2019) Klasterizatsiya tekstov s ispol'zovaniem semantiko-sintaksicheskikh svyazei slov [Clustering of texts using semantic-syntactic connections of words]. *Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologii, mekhaniki i optiki* [Scientific and technical bulletin of information technologies, mechanics and optics], 6, pp. 1058-1063.
4. Liu M. et al. (2022) Short Text Dynamic Clustering Approach for Semantic-Enhanced Knowledge. *ICCSE 2022. Communications in Computer and Information Science*, 1811.
5. Mochalova A.S. (2014) Algoritm semanticheskogo analiza teksta, osnovannyi na bazovykh semanticheskikh shablonakh s udaleniem [Algorithm for semantic text analysis based on basic semantic templates with deletion]. *Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologii, mekhaniki i optiki* [Scientific and technical bulletin of information technologies, mechanics and optics], 5 (93), pp. 126-132.
6. Nikolaev P.L. (2022) Klassifikatsiya knig po zhanram na osnove tekstovykh opisanii posredstvom glubokogo obucheniya [Classifying Books by Genre Based on Text Descriptions Using Deep Learning]. *International Journal of Open Information Technologies*, 1, pp. 36-40.
7. Omar A. (2020) Classifying letarary genre: a methodological synergy of computational modelling and lexical semantics. *Classificação de gêneros literários: uma sinergia metodológica de modelagem computacional e semântica lexical. Texto Livre: Linguagem e Tecnologia*, 13, 2, pp. 83-101.
8. Radlinski F. et al. (2009) Redundancy, diversity and interdependent document relevance. *Newsletter ACM SIGIR Forum*, 43, 2, pp. 46-52.
9. Serov S.S. et al. (2015) Sokrashchenie vremeni otsenki skhozhesti tekstovykh dokumentov na neodnorodnoi mnogoprotsessornoi vychislitel'noi sisteme [Reducing the time for evaluating the similarity of text documents on a non-homogeneous multiprocessor computer system]. *Inzhenernyi vestnik Dona* [Don Engineering Bulletin], 2. Available at: ivdon.ru/ru/magazine/archive/n2p2y2015/3031/ [Accessed 05/05/2023]
10. Shajalal M. et al. (2022) Textual Entailment Recognition with Semantic Features from Empirical Text Representation. *SPELLL 2022. Communications in Computer and Information Science*, 1802.