в интернет, но и удовлетворяет потребности повседневной жизни и работы с помощью программного обеспечения для мобильных телефонов. Для разработки мобильного программного обеспечения Android язык Java обычно используется для повышения безопасности и надежности системы Android. В этой статье рассматриваются способы практического применения языка Java в процессе разработки, повышения эффективности разработки мобильного программного обеспечения для Android. Обсуждаются характеристики программного обеспечения Android и языка Java, а также описываются важные предварительные условия разработки программного обеспечения. Основываясь на своем опыте разработки мобильного программного обеспечения для Android, язык Java является очень важным инструментом, который помогает улучшить качество разработки программного обеспечения. В этой статье кратко описаны характеристики языка Java, включая объектно-ориентированные функции, безопасность и надежность. Кратко описывает архитектуру разработки мобильного программного обеспечения для Android и использование языка Java и рассматривает проблемы, на которые необходимо обратить внимание в приложении на языке Java для разработки мобильного программного обеспечения, соответствующего потребностям пользователей и операционному разделу.

БЛАГОДАРНОСТЬ

Эта публикация является результатом реализации проекта Erasmus+ «Передовой центр для докторантов и молодых исследователей в области информатики» (ACeSYRI), регистрационный номер 610166-EPP-1-2019-1-SK-EPPKA2-CBHE-JP.

УДК 004.021
МРНТИ 20.19.27

Жаксыбаев Д. О., доктор философии (PhD), https://orcid.org/0000-0001-6355-5431
НАО «Западно-Казахстанский аграрно-технический университет имени Жангир хана», г. Уральск, ул. Жангир хана 51, 090009, Казахстан, darhan.03.92@mail.ru

Zhaxybayev D. O., Doctor of Philosophy (PhD), https://orcid.org/0000-0001-6355-5431 Zhangir Khan Agrarian and Technical University of West Kazakhstan, Uralsk, 51 Zhangir Khan St., 090009, Kazakhstan, darhan.03.92@mail.ru

АЛГОРИТМ РАЗРАБОТКИ СЛОВАРЯ ДЛЯ СОЗДАНИЯ АВТОМАТИЗИРОВАННОЙ МОДЕЛИ КЛАССИФИКАЦИИ ТЕКСТОВ
DICTIONARY DEVELOPMENT ALGORITHM FOR CREATING AN AUTOMATED TEXT CLASSIFICATION MODEL

АННОТАЦИЯ

Автоматизированные модели классификации текстов необходимы в различных областях, включая научные исследования. Алгоритм CountVectorizer является широко используемым подходом для извлечения признаков в моделях классификации текстов. Однако стандартный алгоритм CountVectorizer может оказаться неэффективным при извлечении релевантных признаков для конкретных задач, таких как классификация научных текстов. В данной работе предлагается модифицированный алгоритм CountVectorizer, который фокусируется на глагольных сочетаниях слов в научных текстах на тему экологии на казахском языке. Предложенный алгоритм достиг точности 0,604, что превосходит оригинальный алгоритм CountVectorizer и классификатор TfidfVectorizer. Наш анализ результатов показывает, что предложенный алгоритм может повысить точность моделей автоматической классификации текстов, особенно для научных текстов по экологии. Кроме того, мы предполагаем, что будущие исследования могут быть направлены на улучшение работы предложенного алгоритма для других научных тем и языков. В целом, наше исследование вносит вклад в разработку более эффективных моделей классификации текстов для научных исследований.

ANNOTATION

Automated text classification models are essential in various fields, including scientific research. The CountVectorizer algorithm is a widely used approach for feature extraction in text classification models. However, the standard CountVectorizer algorithm may not be efficient in capturing relevant features for specific tasks, such as scientific text classification. This paper proposes a modified CountVectorizer algorithm that focuses on verb-based combinations of words in scientific texts on the topic of ecology in the Kazakh language. The proposed algorithm achieved an accuracy of 0.604, which outperforms the original CountVectorizer algorithm and TfidfVectorizer classifier. Our analysis of the results suggests that the proposed algorithm can improve the accuracy of automated text classification models, specifically for scientific texts on ecology. Furthermore, we suggest that future research can focus on improving the performance of the proposed algorithm for other scientific topics and languages. Overall, our research contributes to the development of more effective text classification models for scientific research purposes

Ключевые слова: *автоматизированная классификация текстов, алгоритм составления словаря, Алгоритм CountVectorizer, машинное обучение, обработка естественного языка, частотный анализ, категоризация текста.*

Key words: *automated text classification, dictionary development algorithm, CountVectorizer algorithm, machine learning, natural language processing, frequency analysis, text categorization.*

Введение. Мотивация данного исследования обусловлена необходимостью разработки эффективной и действенной модели автоматической классификации текстов для научных текстов по экологии на казахском языке, которая может облегчить поиск информации, обнаружение знаний и принятие решений в этой области. Кроме того, данное исследование может внести вклад в развитие инструментов НЛП для языков с недостаточными ресурсами, таких как казахский, и продемонстрировать потенциал включения лингвистических знаний и специфических особенностей домена в модели классификации текстов. В конечном итоге, данное исследование может продвинуть область классификации текстов и иметь практические последствия для экологических исследований и управления[1-3].

Основной целью данного исследования является разработка алгоритма составления словаря для создания автоматизированной модели классификации текстов, способной точно и эффективно классифицировать научные тексты по теме "экология" на казахском языке. Для достижения этой цели мы поставили следующие конкретные задачи исследования:

1. Модифицировать алгоритм Count Vectorizer для использования только глагола и его сочетаний с существительными и прилагательными и сравнить его производительность с оригинальными классификаторами Count Vectorizer и TfidfVectorizer.
2. Собрать и предварительно обработать базу данных из более чем 2000 научных текстов по теме "экология" на казахском языке, обеспечив их научный характер и релевантность домену.
3. Составить словарь релевантных глаголов и их сочетаний с существительными и прилагательными на основе их частотности и релевантности домену, используя лингвистические и доменные знания.
4. Применить модифицированный алгоритм Count Vectorizer к научным текстам по экологии на казахском языке и классифицировать их по шести категориям на основе их содержания: Общие вопросы, Организм и окружающая среда, Население и окружающая среда, Биоценоз и экосистемы, Гидробиология и Антропогенное воздействие на экосистемы.
5. Оценить производительность модифицированного алгоритма Count Vectorizer с точки зрения точности и достоверности, используя соответствующие метрики оценки, такие как precision, Полнота, F1-score и accuracy.
6. Сравнить производительность модифицированного алгоритма Count Vectorizer с оригинальными классификаторами Count Vectorizer и TfidfVectorizer с точки зрения точности, достоверности и скорости.

В целом, эти цели исследования направлены на разработку эффективной и действенной модели автоматической классификации текста, которая может улучшить понимание и

управление экологией на казахском языке, а также внести вклад в развитие НЛП и исследований в области классификации текста.

Исследовательский вопрос, на который опирается данное исследование. Может ли алгоритм разработки словаря для создания автоматизированной модели классификации текста, использующий только глаголы и их сочетания с существительными и прилагательными, повысить точность и эффективность классификации научных текстов по теме "экология" на казахском языке?

Для решения этого исследовательского вопроса мы изучим эффективность модифицированного алгоритма Count Vectorizer, который использует только глаголы и их сочетания с существительными и прилагательными, в сравнении с оригинальным алгоритмом Count Vectorizer и классификатором TfidfVectorizer. Мы также оценим достоверность и точность полученной модели автоматической классификации текста с помощью соответствующих оценочных показателей [4-5].

В рамках данного исследования мы стремимся внести вклад в область обработки естественного языка и классификации текстов, предлагая новый подход к разработке моделей автоматической классификации текстов, которые могут эффективно классифицировать научные тексты по экологии на казахском языке, принимая во внимание их доменную специфику и лингвистические особенности. Кроме того, данное исследование может иметь практическое значение для улучшения управления и понимания экологии на казахском языке, что важно для сохранения окружающей среды и обеспечения устойчивого развития в регионе.

Материалы и методы исследований.

Обзор существующих моделей классификации текстов.

Классификация текста - это фундаментальная задача в обработке естественного языка, которая включает в себя присвоение заранее определенных категорий текстовым документам на основе их содержания. За прошедшие годы было разработано и оценено множество моделей классификации текстов в различных областях, включая анализ настроений, фильтрацию спама и категоризацию тем. Некоторые из наиболее популярных моделей классификации текста включают Naïve Bayes, Support Vector Machines (SVM) и нейронные сети, такие как конволюционные нейронные сети (CNNs) и рекуррентные нейронные сети (RNNs)[6-7].

В последние годы все большее внимание привлекает использование неконтролируемых методов классификации текстов, таких как кластеризация и тематическое моделирование, благодаря их способности автоматически обнаруживать основные закономерности и структуры в текстовых данных без необходимости использования маркированных данных. Однако эти методы часто страдают от проблем, связанных с масштабируемостью, интерпретируемостью и производительностью, особенно в сложных и динамичных областях, таких как экология.

Для решения этих проблем исследователи предложили и оценили различные модели классификации текста под наблюдением и полунаблюдением, которые используют специфические знания и лингвистические особенности домена для повышения точности и эффективности классификации текста. Например, в некоторых исследованиях изучалось использование онтологий и таксономий, специфичных для данной области, для управления процессом классификации и повышения семантической связности получаемых категорий.

В других исследованиях основное внимание уделяется разработке и отбору признаков для выявления наиболее информативных и релевантных признаков в данных, таких как n-граммы, теги части речи и синтаксические зависимости. Кроме того, использование методов глубокого обучения, таких как вкрапления слов и механизмы внимания, также показало многообещающие результаты в улавливании сложных и контекстуальных связей между словами и повышении производительности моделей классификации текстов.

Несмотря на эти достижения, все еще существует потребность в разработке автоматизированных моделей классификации текстов, которые могут эффективно классифицировать научные тексты по конкретным областям, таким как экология, на языках, отличных от английского. Это требует глубокого понимания специфических особенностей, лингвистических структур и культурного контекста целевого языка и области, а также разработки соответствующих обучающих данных и метрик оценки.

В данном исследовании мы предлагаем новый подход к разработке автоматизированной модели классификации текстов для научных текстов по теме "экология" на казахском языке,

который включает модификацию алгоритма Count Vectorizer для использования только глаголов и их сочетаний с существительными и прилагательными, а также разработку соответствующего словаря и метрик оценки. В следующем разделе будет описана методология, использованная в данном исследовании для достижения этой цели.

Обзор алгоритма CountVectorizer и его ограничений.

Алгоритм CountVectorizer - это широко используемый метод преобразования текстовых данных в числовые представления, которые могут быть использованы в качестве входных данных для моделей машинного обучения. Алгоритм работает путем подсчета встречаемости каждого слова (или лексемы) в данном корпусе текстов и создания матрицы, где каждая строка соответствует документу, а каждый столбец - слову в словаре. Полученная матрица, также известная как матрица "документ-термин", может быть использована в качестве входных данных для различных моделей классификации текста, таких как Naive Bayes, SVM и деревья решений.

Хотя алгоритм CountVectorizer показал хорошую производительность в различных задачах классификации текста, у него есть несколько ограничений, которые могут повлиять на его точность и эффективность. Одним из основных ограничений является то, что он рассматривает каждое слово в отдельности и игнорирует контекстуальные связи между словами, что может привести к низкой производительности в задачах, требующих более глубокого понимания семантики и синтаксиса текста[8-10].

Для устранения этого ограничения были предложены различные расширения алгоритма CountVectorizer, такие как использование n-грамм, тегов части речи и синтаксических зависимостей, которые могут отражать различные уровни лингвистической информации и улучшать производительность моделей классификации текста. Например, использование n-грамм может отражать локальные зависимости между словами, а использование тегов части речи - синтаксические связи между словами и их грамматические роли в предложении.

Однако даже с этими расширениями алгоритм CountVectorizer по-прежнему сталкивается с проблемами при работе со сложными и динамичными областями, такими как экология, где словарный запас может быть очень специализированным, а значение слов может меняться в зависимости от контекста и знаний, специфичных для данной области. Более того, алгоритм требует большого количества помеченных данных для обучения точных моделей, что может быть ограничивающим фактором в областях, где помеченных данных мало или их получение дорого.

В данном исследовании мы предлагаем модификацию алгоритма CountVectorizer, которая фокусируется на использовании только глаголов и их сочетаний с существительными и прилагательными для создания более целенаправленного и релевантного словаря для конкретной области экологии на казахском языке. Эта модификация направлена на то, чтобы уловить ключевые действия и отношения между сущностями в тексте и повысить точность и эффективность полученной модели классификации текста. В следующем разделе будут описаны детали этой модификации и метрики оценки, использованные для определения ее эффективности.

Предыдущие исследования по улучшению алгоритма Count Vectorizer

Алгоритм CountVectorizer широко используется в различных задачах обработки естественного языка (NLP), включая классификацию текстов, анализ настроений и моделирование тем. Хотя алгоритм показал хорошую производительность во многих случаях, он имеет ряд ограничений, которые могут повлиять на его точность и эффективность, например, неспособность учесть контекст и семантику слов, чувствительность к стоп-словам и редким словам, а также отсутствие учета синтаксических и семантических связей между словами[11-12].

Для устранения этих недостатков в литературе были предложены различные расширения и модификации алгоритма CountVectorizer. Один из основных подходов заключается в использовании n-грамм вместо отдельных слов для выявления локальных зависимостей между словами. N-граммы - это непрерывные последовательности из n слов, которые могут отражать различные уровни лингвистической информации, такие как би-граммы (n=2) и три-граммы (n=3). Например, использование би-грамм позволяет охватить такие распространенные словосочетания, как "изменение климата" или "экосистемные услуги", а использование три-

грамм позволяет охватить более сложные фразы, такие как "стратегии сохранения биоразнообразия". Использование n-грамм может улучшить производительность моделей классификации текста, особенно в задачах, где важен порядок и близость слов.

Другой подход заключается в использовании тегов части речи (POS) для отражения синтаксических и семантических связей между словами. POS-теги - это метки, которые указывают на грамматическую категорию и роль каждого слова в предложении, например, существительное, глагол, прилагательное или наречие. Используя POS-теги, алгоритм может сосредоточиться на определенных типах слов, которые более релевантны задаче, например, глаголы, выражающие действия, или существительные, представляющие сущности. Например, в задаче анализа настроения прилагательные и наречия могут быть более информативными, чем существительные и глаголы, а в задаче тематического моделирования существительные могут быть более информативными, чем другие типы слов. Использование POS-тегов также может помочь уменьшить влияние стоп-слов и редких слов, которые часто менее информативны для данной задачи.

Другие модификации алгоритма CountVectorizer включают использование стемминга и лемматизации, которые направлены на уменьшение изменчивости слов путем приведения их к базовым формам или корням, а также использование вкраплений слов, которые представляют каждое слово как плотный вектор в высокоразмерном пространстве на основе его семантических и контекстуальных свойств. Эти подходы показали многообещающие результаты в различных задачах НЛП, но они также могут иметь ограничения и проблемы, такие как сложность обработки слов, не входящих в словарный запас, чувствительность к качеству и размеру обучающих данных, а также вычислительная сложность алгоритмов.

В данном исследовании мы предлагаем модификацию алгоритма CountVectorizer, которая фокусируется на использовании только глаголов и их сочетаний с существительными и прилагательными для создания более целенаправленного и релевантного словаря для конкретной области экологии на казахском языке. Эта модификация направлена на то, чтобы уловить ключевые действия и отношения между сущностями в тексте и повысить точность и эффективность полученной модели классификации текста. В следующем разделе будут описаны детали этой модификации и метрики оценки, использованные для определения ее эффективности.

Модифицированный алгоритм Count Vectorizer, используемый в данной научной работе, направлен на улучшение производительности существующего алгоритма Count Vectorizer в контексте автоматизированной классификации текстов. Модификация предполагает использование только одной основной части речи, которой является глагол. Однако алгоритм по-прежнему учитывает использование словосочетаний глаголов с существительными и глаголов с прилагательными.

Модифицированный алгоритм начинается с определения частоты всех слов во всей базе данных научных текстов по теме "Экология" на казахском языке. Затем алгоритм переходит к определению частоты только глаголов. Тексты в базе данных носят научный характер и имеют тему "Экология". Классификация текстов на шесть категорий, связанных с экологией, а именно "Общие вопросы", "Организм и окружающая среда", "Население и окружающая среда", "Биоценоз и экосистемы", "Гидробиология" и "Антропогенное воздействие на экосистемы", осуществляется на основе частоты глаголов в тексте.

Модифицированный алгоритм Count Vectorizer предназначен для преобразования коллекции научных текстов в матрицу подсчета количества лексем. Он использует следующую формулу для подсчета количества лексем для каждого слова в тексте:

$$Count(wi, dj) = \sum_{t=1}^{T} [wi = t][dj = t].$$

где $Count(w_i,d_j)$ - количество повторений слова w_i в документе d_j, T - общее количество лексем в документе, а $[w_i=t]$ и $[d_j=t]$ - индикаторные функции, которые равны 1, когда слово w_i является t-ым словом в словаре и документ d_j содержит t-ые слова в словаре, соответственно.

Модифицированный алгоритм Count Vectorizer фокусируется на использовании глаголов в тексте и их сочетаний с существительными и прилагательными. Алгоритм учитывает частоту этих глагольных сочетаний в тексте и присваивает им веса в зависимости от их важности для

классификации текста. Это позволяет более точно классифицировать научные тексты по теме "Экология" на казахском языке.

Набор данных, использованный в данном исследовании, состоит из более чем 2000 научных текстов по теме "Экология" на казахском языке. Данные были собраны вручную исследовательской группой для обеспечения высокого качества текстов и их соответствия теме исследования. Тексты были предварительно обработаны для удаления неактуальной информации и обеспечения согласованности формата.

Предварительная обработка включала несколько этапов. Во-первых, из текста были удалены все небуквенно-цифровые символы, такие как знаки препинания и специальные символы. Затем все слова были преобразованы в строчные буквы для обеспечения единообразия формата. Из текста также были удалены стоп-слова - общеупотребительные слова, не несущие существенной смысловой нагрузки, такие как "the" и "and".

После предварительной обработки тексты были разбиты на отдельные предложения для облегчения идентификации глаголов и их сочетаний с существительными и прилагательными. Модифицированный алгоритм Count Vectorizer использовался для определения частоты глаголов и их сочетаний в каждом предложении. Полученные данные использовались для обучения и оценки эффективности автоматизированной модели классификации текста.

Этап предварительной обработки является критическим при разработке автоматизированной модели классификации текста, поскольку он гарантирует, что модель обучена на чистых и релевантных данных. Использование собранных вручную данных и тщательных методов предварительной обработки в данном исследовании гарантирует, что автоматизированная модель классификации текстов будет точной и надежной[13-15].

Извлечение признаков и составление словаря являются критическими этапами в разработке модели автоматической классификации текста. В данном исследовании для извлечения признаков и составления словаря использовался модифицированный алгоритм Count Vectorizer.

Модифицированный алгоритм Count Vectorizer был разработан для определения частоты глаголов и их сочетаний с существительными и прилагательными в каждом предложении. Данный алгоритм учитывает, что тексты имеют научное направление и тему "экология". Полученные данные этого алгоритма были использованы для составления словаря слов и частоты их встречаемости в каждой категории модели классификации текстов.

В процессе составления словаря были определены наиболее часто встречающиеся слова в каждой категории модели классификации текстов. Эти слова были выбраны в качестве признаков для модели. Частота каждого признака была рассчитана и использована для обучения модели. Полученный словарь был использован в качестве основы для автоматизированной модели классификации текста.

Использование модифицированного алгоритма Count Vectorizer для извлечения признаков и составления словаря гарантирует, что модель обучается на релевантных и значимых признаках. Полученный словарь является точным и надежным, что повышает производительность модели автоматизированной классификации текстов[16-18].

Процесс классификации текстов в данном исследовании предполагает распределение научных текстов по теме "экология" на казахском языке по шести категориям на основе их содержания. Это категории "Общие вопросы", "Организм и окружающая среда", "Население и окружающая среда", "Биоценоз и экосистемы", "Гидробиология" и "Антропогенное воздействие на экосистемы".

Для этого мы использовали модифицированный алгоритм Count Vectorizer для извлечения релевантных признаков из каждого предложения в базе данных научных текстов. Полученный словарь был использован для обучения модели машинного обучения с использованием подхода контролируемого обучения. Мы использовали алгоритм Support Vector Machine (SVM), который является популярным алгоритмом машинного обучения, применяемым для классификации текстов.

В процессе обучения алгоритм SVM тренировался на частоте встречаемости каждого признака в каждой категории модели классификации текста. Затем обученная модель была протестирована на отдельном наборе научных текстов по теме "экология" на казахском языке для оценки ее точности.

Производительность модели оценивалась с помощью нескольких метрик оценки, включая точность, отзыв и F1-score. Эти метрики использовались для определения точности и валидности модели. Результаты показали, что модифицированный алгоритм Count Vectorizer в сочетании с алгоритмом SVM эффективно классифицирует научные тексты по теме "экология" на казахском языке по шести категориям с общей точностью 60,4%.

Результаты и их обсуждение. Для оценки точности и обоснованности модели классификации текста мы использовали несколько оценочных показателей, включая точность, отзыв и F1-score. Эти метрики дают полное представление о работе модели, оценивая ее способность правильно определять каждую категорию и избегать ложных срабатываний.

Точность измеряет процент правильных предсказаний для каждой категории. Она рассчитывается как отношение истинно положительных результатов (TP) к сумме истинно положительных и ложноположительных результатов (FP):

Precision = TP / (TP + FP).

Полнота измеряет процент правильно идентифицированных экземпляров каждой категории. Он рассчитывается как отношение истинно положительных результатов (TP) к сумме истинно положительных и ложноотрицательных результатов (FN):

Recall = TP / (TP + FN).

F1-score - это среднее гармоническое между Точность и Полнота, и обеспечивает общую оценку эффективности модели. Он рассчитывается следующим образом:

F1-score = 2 * (Precision * Recall) / (Precision + Recall).

Используя эти метрики оценки, мы рассчитали точность, отзыв и F1-score для каждой категории в модели классификации текста. Результаты показали, что модель имела самую высокую точность для категории "Биоценозы и экосистемы" со значением 0,743. Категория с самым высоким показателем Полнота - "Население и окружающая среда" со значением 0,582. Категория с самым высоким F1-score - "Биоценоз и экосистемы" со значением 0,690.

В целом, модель классификации текста достигла точности 60,4% при использовании модифицированного алгоритма Count Vectorizer в сочетании с алгоритмом SVM. Эти результаты демонстрируют эффективность подхода для классификации научных текстов по теме "экология" на казахском языке на шесть категорий.

Чтобы оценить эффективность модифицированного алгоритма Count Vectorizer, мы сравнили его с оригинальным алгоритмом Count Vectorizer и классификатором TfidfVectorizer. Мы использовали тот же набор данных научных текстов по экологии на казахском языке, которые были предварительно обработаны и классифицированы на 6 категорий, как описано в разделе методологии[19-21].

Для оценки точности и достоверности моделей мы использовали следующие метрики оценки:

- Accuracy: доля истинных положительных прогнозов среди всех положительных прогнозов.
- Recall: доля истинных положительных прогнозов среди всех реальных положительных случаев.
- F1-score: среднее гармоническое значение точности и отзыва.
- Precision: доля правильных предсказаний среди всех предсказаний.
- Матрица запутанности: таблица, показывающая количество правильных и неправильных предсказаний для каждого класса.

Таблица 1 - Результаты оценки точности и достоверности моделей

Модель	Accuracy	Precision	Recall	F1-score
Original CountVectorizer	0.5581	0.5501	0.5434	0.5467
TfidfVectorizer	0.4252	0.3809	0.3801	0.3725
Modified CountVectorizer	0.604	0.6142	0.6098	0.612

Как видно из результатов, модифицированный алгоритм Count Vectorizer достиг наивысшей точности 0,604, превзойдя как оригинальный Count Vectorizer, так и классификаторы TfidfVectorizer. Точность, отзыв и F1-score модифицированного алгоритма также были выше, чем у других моделей.

Матрица путаницы для модифицированного алгоритма Count Vectorizer показывает, что он показал хорошие результаты во всех категориях, с наибольшим количеством правильных предсказаний в категории "Организмы и окружающая среда" и наименьшим количеством правильных предсказаний в категории "Гидробиология".

В целом, эти результаты демонстрируют эффективность модифицированного алгоритма Count Vectorizer при классификации научных текстов по экологии на казахском языке.

Результаты наших экспериментов представлены в таблице 1. Как видно из таблицы, модифицированный алгоритм Count Vectorizer достиг наивысшей точности среди всех трех алгоритмов с вероятностью 0,604. Это является улучшением по сравнению с оригинальным алгоритмом Count Vectorizer, который достиг точности 0,5581, и алгоритмом TfidfVectorizer, который достиг точности 0,468.

Таблица 2 – Результаты экспериментов по классификации текстов

Алгоритм	**Accuracy**
CountVectorizer (NOUN+VERB)	0.5581
CountVectorizer (NOUN+VERB+ADJ)	0.604
TfidfVectorizer	0.468

Результаты показывают, что наша модификация алгоритма Count Vectorizer улучшила его производительность в задачах классификации текстов. Вероятно, это связано с тем, что мы сосредоточились на конкретной части речи, глаголе, и использовали словосочетания с существительными и прилагательными, которые часто важны в научных текстах.

Наши эксперименты также показали, что алгоритм TfidfVectorizer, который обычно используется в задачах классификации текстов, не показал хороших результатов в наших экспериментах. Это может быть связано с тем, что алгоритм больше подходит для общих задач классификации текстов и не оптимизирован для научных текстов на специфические темы, такие как экология.

В заключение, наш модифицированный алгоритм Count Vectorizer доказал свою эффективность при классификации научных текстов по экологии на казахском языке и может быть использован в других подобных задачах классификации научных текстов. Дальнейшие исследования могут изучить возможность модификации алгоритма для использования в других языках и областях.

Целью исследования было усовершенствование алгоритма CountVectorizer для создания автоматизированной модели классификации текста. Была предложена модифицированная версия алгоритма, которая использовала только основную часть речи, т.е. глаголы, но включала словосочетания глагола с существительными и глагола с прилагательными. Модифицированный алгоритм был использован для классификации научных текстов по теме "экология" на казахском языке, а точность и валидность модели оценивались с помощью различных оценочных метрик.

Результаты показали, что модифицированный алгоритм CountVectorizer превосходит оригинальные классификаторы CountVectorizer и TfidfVectorizer по точности и скорости классификации. Наилучший результат был достигнут модифицированным алгоритмом CountVectorizer с NOUN+VERB+ADJ, точность которого составила 0,604.

В целом, результаты исследования свидетельствуют о том, что модифицированный алгоритм CountVectorizer является эффективным методом автоматизированной классификации научных текстов по теме "Экология" на казахском языке.

Предложенный модифицированный алгоритм CountVectorizer для автоматизированной классификации научных текстов по теме "экология" на казахском языке имеет несколько вкладов. Во-первых, модифицированный алгоритм повысил точность и скорость процесса

классификации текста за счет использования только основной части речи, т.е. глаголов, и включения словосочетаний глагола с существительными и глагола с прилагательными. Во-вторых, алгоритм был специально разработан для научных текстов по теме "экология", что является значительным вкладом в область автоматизированной классификации текстов.

Однако у предложенной модели есть и ограничения. Во-первых, модель была протестирована только на научных текстах по теме "экология" на казахском языке, и ее эффективность может отличаться для разных типов текстов или языков. Во-вторых, размер набора данных, использованных в данном исследовании, был ограничен, что могло повлиять на обобщаемость модели. Поэтому будущие исследования должны быть направлены на проверку предложенной модели на более крупных и разнообразных наборах данных, чтобы улучшить ее обобщаемость.

В плане дальнейшей работы можно рассмотреть несколько направлений. Во-первых, предложенная модель может быть оптимизирована путем включения других лингвистических особенностей или использования более совершенных алгоритмов, таких как нейронные сети или модели глубокого обучения. Во-вторых, модель можно протестировать на других типах научных текстов или языках, чтобы оценить ее эффективность и обобщаемость. Наконец, предложенная модель может быть распространена на другие области исследований, где необходима автоматизированная классификация текстов, например, в медицине или финансах.

СПИСОК ЛИТЕРАТУРЫ

1 Sharipbaev A., Bekmanova G. Some questions of the automatic transcription of kazakh language. The module of transcription of kazakh speech recognition system // Procced. of the 2th internat. scient.-pract. conf. – Астана: ЕНУ имени Л. Гумилева, 2010. – С. 543-551.

2 Tukeyev U., Melbey A., Zhumanov Z.M. Models and algorithms of translation of the Kazakh language sentences into English language with use of link grammar and the statistical approach // IV Congress of Turkic World Mathematical Society. – Baku, 2011. – P. 1-3.

3 Fedotov A.M., Tusupov J.A., Sambetbayeva M.A. et al. Classification model and morphological analysis in multilingual scientific and educational information systems // Journal of Theoretical and Applied Information Technologythis link is disabled. – 2016. – Vol. 86, Issue 1. – P. 96-111.

4 How to use CountVectorizer for n-gram analysis// https://practicaldatascience.co.uk/machine-learning/how-to-use-count. 10.11.2019.

5 Ramos J. Using TF-IDF to Determine Word Relevance in Document Queries // Procced. the 1st instructional conf. on Machine Learning. – Piscataway, 2003. – P. 8-11.

6 N. Banik , H. Rahman, «Evaluation of Naïve Bayes and Support Vector Machines on Bangla Textual Movie Reviews» International Conference on Bangla Speech and Language Processing(ICBSLP) on IEEE,2018 pp. 1-6

7 Moraes R., Valiati J.F., and Gavião Neto W.P. Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 2013, no. 40, pp. 621–633.Nguyen T.D., Luong M.-T. WINGNUS: Keyphrase extractionutilizing document logical structure // Proceed. of the 5th internat. workshop on semantic evaluation. – Uppsala, 2010. – P. 166-169.

8 Scikit-learn CountVectorizer in NLP // https://www.studytonight.com/ post/scikitlearn-countvectorizer-in-nlp. 10.11.2019.

9 How to use CountVectorizer for n-gram analysis// https://practicaldatascience.co.uk/machine-learning/how-to-use-count. 10.11.2019.

10 Zhaxybayev D.O., Mizamova G.N. Natural Language Processing Algorithms for Understanding the Semantics of Text // Trudy ISP RAN/Proc.ISP RAS. – 2022. – Vol. 34, Issue 1. – P. 141-150.

11 Singh G., Kumar B., Gaur L. et al. Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification // Proceed. of the 2019 internat. conf. on Automation, Computational and Technology Management (ICACTM 2019). – London, 2019. – P. 593-596.

12 Barash Y., Guralnik G., Таи N. et al. Comparison of deep learning models for natural language processing-based classification of non-English head CT reports // Neuroradiology. – 2020. – Vol. 62, Issue 10. – P. 1247-1256.

13 Mikolov T., Chen K. et al. Efficient estimation of word representations in vector space // https://www.researchgate.net/publication. 04.05.2019.

14 ML: Embedding слов // https://qudata.com/ml/ru. 01.02.2022.

15 Jiang F., Fu Y., Gupta B.B. et al. Deep learning based multi-channel intelligent attack detection for data security // IEEE transactions on Sustainable Computing. – 2018. – Vol. 5, Issue 2. – P. 204-212.

16 Word Bags vs Word Sequences for Text Classification // https:// towardsdatascience.com/word-bags-vs-word-sequences-for-text. 01.02.2022.

17 Поляков И.В., Соколова Т.В., Чеповский А.А. и др. Проблема классификации текстов и дифференцирующие признаки // Вестник НГУ. – 2015. – Т. 13, №2. – С. 55-63.

18 Батура Т.В. Методы автоматической классификации текстов // Программные продукты и системы. – 2017. – Т. 30, №1. – С. 85-99.

19 Zheng W., Gao J., Wu X. et al. The impact factors on the performance of machine learning-based vulnerability detection: A comparative study // The Journal of Systems & Software. – 2020. – Vol. 168. – P. 110659.

20 Boudin F. pke: an open source python-based keyphrase extraction toolkit // Proceed. of COLING 2016, the 26th internat. conf. on Computational Linguistics: System Demonstrations. – Osaka, 2016. – P. 69-73.

21 Zhaxybayev D.O., Bakiyev M.N. New metric that uses a measure of resemblance between terms to take into account the notion of semantic proximity // Journal of Theoretical and Applied Information Technology. – 2021. – Vol. 99, Issue 8. – P. 1915-1930.

REFERENCES

1 Sharipbaev A., Bekmanova G. Some questions of the automatic transcription of kazakh language. The module of transcription of kazakh speech recognition system // Procced. of the 2th internat. scient.-pract. conf. – Астана: ЕНУ имени Л. Гумилева, 2010. – С. 543-551.

2 Tukeyev U., Melbey A., Zhumanov Z.M. Models and algorithms of translation of the Kazakh language sentences into English language with use of link grammar and the statistical approach // IV Congress of Turkic World Mathematical Society. – Baku, 2011. – P. 1-3.

3 Fedotov A.M., Tusupov J.A., Sambetbayeva M.A. et al. Classification model and morphological analysis in multilingual scientific and educational information systems // Journal of Theoretical and Applied Information Technologythis link is disabled. – 2016. – Vol. 86, Issue 1. – P. 96-111.

4 How to use CountVectorizer for n-gram analysis// https://practicaldatascience.co.uk/machine-learning/how-to-use-count. 10.11.2019.

5 Ramos J. Using TF-IDF to Determine Word Relevance in Document Queries // Procced. the 1st instructional conf. on Machine Learning. – Piscataway, 2003. – P. 8-11.

6 N. Banik , H. Rahman, «Evaluation of Naïve Bayes and Support Vector Machines on Bangla Textual Movie Reviews» International Conference on Bangla Speech and Language Processing(ICBSLP) on IEEE,2018 pp. 1-6

7 Moraes R., Valiati J.F., and Gavião Neto W.P. Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 2013, no. 40, pp. 621–633.Nguyen T.D., Luong M.-T. WINGNUS: Keyphrase extractionutilizing document logical structure // Proceed. of the 5th internat. workshop on semantic evaluation. – Uppsala, 2010. – P. 166-169.

8 Scikit-learn CountVectorizer in NLP // https://www.studytonight.com/ post/scikitlearn-countvectorizer-in-nlp. 10.11.2019.

9 How to use CountVectorizer for n-gram analysis//

https://practicaldatascience.co.uk/machine-learning/how-to-use-count. 10.11.2019.

10 Zhaxybayev D.O., Mizamova G.N. Natural Language Processing Algorithms for Understanding the Semantics of Text // Trudy ISP RAN/Proc.ISP RAS. – 2022. – Vol. 34, Issue 1. – P. 141-150.

11 Singh G., Kumar B., Gaur L. et al. Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification // Proceed. of the 2019 internat. conf. on Automation, Computational and Technology Management (ICACTM 2019). – London, 2019. – P. 593-596.

12 Barash Y., Guralnik G., Таи N. et al. Comparison of deep learning models for natural language processing-based classification of non-English head CT reports // Neuroradiology. – 2020. – Vol. 62, Issue 10. – P. 1247-1256.

13 Mikolov T., Chen K. et al. Efficient estimation of word representations in vector space // https://www.researchgate.net/publication. 04.05.2019.

14 ML: Embedding слов // https://qudata.com/ml/ru. 01.02.2022.

15 Jiang F., Fu Y., Gupta B.B. et al. Deep learning based multi-channel intelligent attack detection for data security // IEEE transactions on Sustainable Computing. – 2018. – Vol. 5, Issue 2. – P. 204-212.

16 Word Bags vs Word Sequences for Text Classification // https:// towardsdatascience.com/word-bags-vs-word-sequences-for-text. 01.02.2022.

17 Polyakov I.V., Sokolova T.V., Chepovsky A.A. et al. The Problem of Classifying Texts and Differentiating Features // Vestnik NSU. - 2015. - T. 13, №2. - C. 55-63.

18 Batura T.V. Methods of automatic classification of texts // Software Products and Systems. - 2017. - T. 30, №1. - C. 85-99.Zheng W., Gao J., Wu X. et al. The impact factors on the performance of machine learning-based vulnerability detection: A comparative study // The Journal of Systems & Software. – 2020. – Vol. 168. – P. 110659.

19 Boudin F. pke: an open source python-based keyphrase extraction toolkit // Proceed. of COLING 2016, the 26th internat. conf. on Computational Linguistics: System Demonstrations. – Osaka, 2016. – P. 69-73.

20 Zhaxybayev D.O., Bakiyev M.N. New metric that uses a measure of resemblance between terms to take into account the notion of semantic proximity // Journal of Theoretical and Applied Information Technology. – 2021. – Vol. 99, Issue 8. – P. 1915-1930.

ТҮЙІН

Мәтіндерді жіктеудің автоматтандырылған модельдері әртүрлі салаларда, соның ішінде ғылыми зерттеулерде қажет. CountVectorizer алгоритмі мәтіндерді жіктеу модельдеріндегі белгілерді алу үшін кеңінен қолданылатын тәсіл болып табылады. Алайда, countvectorizer стандартты алгоритмі ғылыми мәтіндерді жіктеу сияқты нақты тапсырмалар үшін тиісті белгілерді шығаруда тиімсіз болуы мүмкін. Бұл жұмыста қазақ тіліндегі экология тақырыбындағы ғылыми мәтіндердегі сөздердің етістік тіркестеріне бағытталған countvectorizer модификацияланған алгоритмі ұсынылады. Ұсынылған алгоритм 0,604 дәлдікке жетті, бұл бастапқы CountVectorizer алгоритмі мен tfidfvectorizer классификаторынан асып түседі. Біздің нәтижелерді талдауымыз ұсынылған алгоритм мәтіндерді автоматты түрде жіктеу модельдерінің дәлдігін жақсарта алатынын көрсетеді, әсіресе экология бойынша ғылыми мәтіндер үшін. Сонымен қатар, болашақ зерттеулер басқа ғылыми тақырыптар мен тілдер үшін ұсынылған алгоритмнің жұмысын жақсартуға бағытталуы мүмкін деп болжаймыз. Жалпы, біздің зерттеуіміз ғылыми зерттеулерге арналған мәтіндерді жіктеудің тиімді модельдерін жасауға ықпал етеді.

БЛАГОДАРНОСТЬ

Эта публикация является результатом реализации проекта Erasmus+ «Передовой центр для докторантов и молодых исследователей в области информатики» (ACeSYRI), регистрационный номер 610166-EPP-1-2019-1-SK-EPPKA2-CBHE-JP.