СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Проблема некорректных заимствований в научной и образовательной среде имеет многоуровневый характер и включает не только явный плагиат, но и более скрытые формы нарушения академической этики: ошибочные ссылки, неправильно приписанные источники, искаженные цитаты, отсутствие достоверной библиографии. Масштаб проблемы растет с увеличением объема научных работ и доступности технологий, позволяющих легко манипулировать текстами.

Проблема обнаружения нечетких дубликатов является одной из наиболее важных и трудных задач анализа веб-данных и поиска информации. Основным препятствием для успешного решения данной задачи является гигантский объем данных, что делает практически невозможным попарное сравнение текстов документов в разумное время [1].

Современные подходы к выявлению некорректных заимствований выходят далеко за рамки простого поиска текстовых совпадений. Требуется применение комплекса методов обработки естественного языка (NLP), машинного обучения, информационного поиска (IR) и автоматизированных систем верификации источников. Каждый класс алгоритмов решает специфическую задачу: от синтаксического обнаружения копируемых фрагментов до семантического анализа парафраза и проверки корректности ссылок на литературу [2].

Цель данной работы — классифицировать и описать основные методы и алгоритмы, применяемые для выявления некорректных заимствований.

1 Классификация и основные подходы к обнаружению заимствований

1.1 Классификация алгоритмов и методов детекции

Все многообразие подходов к обнаружению некорректных заимствований можно систематизировать в пять основных классов:

1) методы синтаксического сравнения и «отпечатков» текста;
2) лексико-статистические методы и метрики схожести;
3) методы семантического/смыслового сопоставления;
4) модели авторского стиля и стилометрия;
5) алгоритмы анализа и верификации цитирования.

Такая классификация позволяет проанализировать каждый подход с точки зрения его особенностей, преимуществ и ограничений [1].

2 Синтаксические и лексико-статистические методы

2.1 Методы синтаксического сравнения и отпечатков текста

2.1.1 Общие принципы

Для решения задачи обнаружения нечетких дубликатов текстов применяются методы синтаксического сравнения. Идея этих методов заключается в том, чтобы получить компактное представление текста, сохраняющее его уникальные черты, и сравнивать эти представления вместо полных текстов. Такое представление называется «отпечатком» или сигнатурой документа. Это позволяет значительно ускорить сравнение больших объемов текстовых данных. Основной принцип состоит в следующем: текст разбивается на перекрывающиеся или неперекрывающиеся фрагменты фиксированной длины, каждому фрагменту вычисляется хеш-значение, и затем сравниваются хеши [1].

2.1.2 Шинглы и метод Бродера

Одним из первых исследований в области нахождения нечетких дубликатов является работа А. Бродера, в которой был предложен синтаксический метод оценки сходства между документами, основанный на представлении документа в виде множества всевозможных последовательностей фиксированной длины k, состоящих из соседних слов. Такие последовательности были названы шинглами (k-грамм слов). Два документа считались похожими, если их множества шинглов существенно пересекались.

Поскольку число шинглов примерно равно длине документа в словах, были предложены два метода сэмплирования для получения репрезентативных подмножеств. Первый метод оставлял только те шинглы, чьи дактилограммы (численные отпечатки, вычисляемые по алгоритму Рабина-Карпа) делились без остатка на некоторое число m. Второй метод отбирал фиксированное число s шинглов с наименьшими значениями дактилограмм [1].

2.1.3 Алгоритм Winnowing

Winnowing — это модификация простого k-граммного анализа, предложенная для повышения эффективности. Алгоритм работает следующим обра-

зом: текст разбивается на k-граммы (подстроки из k символов); для каждой k-граммы вычисляется хеш-значение; из всех хешей выбираются только минимальные значения в пределах скользящего окна размером w. Эти избранные хеши образуют отпечаток документа, после чего отпечатки различных документов сравниваются для определения схожести. Преимущество алгоритма Winnowing состоит в устойчивости к перестановкам фрагментов и синтаксическим изменениям, при этом сохраняя низкую вычислительную сложность. Алгоритм используется в системах MOSS и JPlag для обнаружения плагиата в исходном коде и текстах [4].

2.1.4 Дактилограммы и алгоритм Рабина-Карпа

Одними из первых исследований в области нахождения нечетких дубликатов являются работы U. Manber и N. Heintze. Дактилограмма (также называемая отпечатком или хеш-сигнатурой) файла или документа включает все текстовые подстроки фиксированной длины. Численное значение дактилограмм вычисляется с помощью алгоритма случайных полиномов Рабина-Карпа. Дактилограмма отличается от простого отпечатка тем, что представляет собой набор множественных хеш-значений для разных подстрок, а не единственное значение. В качестве меры сходства двух документов используется отношение числа общих подстрок к размеру файла или документа. Алгоритм Рабина-Карпа основан на быстром вычислении хешей для перекрывающихся подстрок с использованием скользящего окна. Полиномиальное хеширование позволяет за $O(1)$ пересчитать хеш для следующей подстроки на основе хеша предыдущей. Метод особенно эффективен при поиске точных совпадений в больших массивах текста [1].

2.1.5 Мегашинглы и супершинглы

Дальнейшим развитием концепций Бродера являются исследования D. Fetterly. Для каждого документа вычисляются 84 дактилограммы по алгоритму Рабина-Карпа с помощью взаимно-однозначных и независимых функций. В результате каждый документ представлялся 84 шинглами, минимизирующими значение соответствующей функции. Затем 84 шингла разбиваются на 6 групп по 14 шинглов в каждой. Эти группы называются супершинглами. Документ

представляется всевозможными попарными сочетаниями из 6 супершинглов, которые называются мегашинглами. Число таких мегашинглов равно 15. Два документа сходны по содержанию, если у них совпадает хотя бы один мегашингл. Ключевое преимущество данного алгоритма состоит в том, что любой документ (в том числе и очень маленький) всегда представляется вектором фиксированной длины, и сходство определяется простым сравнением координат вектора [1].

2.1.6 N-граммный анализ и SimHash

N-граммный анализ — одна из самых базовых, но действенных методик. Документ представляется как набор n-грамм (последовательности из n символов или слов). Сравнение документов производится по пересечению их n-грамм. Алгоритм не учитывает порядок n-грамм в документе и работает с ними как с множеством. SimHash расширяет эту идею: для каждого документа строится битовый отпечаток фиксированной длины путем комбинирования хешей его n-грамм. Два документа считаются схожими, если расстояние Хемминга между их отпечатками мало [5].

2.1.7 Locality Sensitive Hashing (LSH)

LSH — это техника быстрого поиска похожих текстов в больших коллекциях. Алгоритм хеширует входные элементы таким образом, что похожие элементы с высокой вероятностью получают один и тот же хеш или хеши в одних и тех же «корзинах». Для поиска похожих текстов на основе LSH создается несколько таблиц хешей с разными хеш-функциями. Входной документ хешируется со всеми этими функциями, и его представление проверяется в каждой таблице. Документы, попадающие в одну и ту же корзину, являются кандидатами на сходство. Это позволяет за логарифмическое или даже сублогарифмическое время находить кандидатов на сходство без полного сравнения со всеми документами в базе. Метод особенно ценен для масштабируемых систем с миллионами или миллиардами документов [5].

2.2 Лексико-статистические методы и метрики схожести

2.2.1 Численный признак TF · IDF и векторное пространство

Для решения задачи сравнения документов на основе терминов используются лексико-статистические методы. Одним из наиболее важных методов является представление документа в виде вектора численных признаков. TF · IDF (Term Frequency-Inverse Document Frequency) — это численный признак, характеризующий важность термина в документе относительно всей коллекции. Величина TF · IDF рассчитывается как произведение двух компонент: частоты термина в документе (TF) и его редкости в коллекции (IDF).

Построение вектора TF · IDF для документа происходит следующим образом: строится частотный словарь документа; для каждого слова вычисляется произведение TF · IDF; вектор упорядочивается по убыванию этого произведения; выбираются топ-N слов с наибольшими весами и сцепляются в алфавитном порядке; в качестве сигнатуры документа вычисляется контрольная сумма (например, CRC32 или MD5) полученной строки.

Два текста сравниваются как векторы в многомерном пространстве, где каждое слово представляет одну координату. Схожесть между двумя такими векторами оценивается с помощью косинусной меры близости (косинус угла между векторами). Косинусная мера близости принимает значения от -1 до $+1$, где значение 1 означает полное совпадение, 0 означает ортогональность (отсутствие схожести), а -1 означает полную противоположность. Документы с высокой косинусной мерой близости (обычно выше порога 0.8) считаются потенциально плагиатными [6].

2.2.2 Метод I-Match

Для решения задачи быстрого выявления дубликатов применяется сигнатурный подход, основанный на лексических принципах, предложенный A. Chowdhury. Основная идея метода I-Match состоит в вычислении дактилограммы для представления содержания документов на основе отобранного подмножества слов. Сначала для исходной коллекции документов строится словарь L, который включает слова со средними значениями IDF (исключаются очень частые служебные слова и очень редкие слова). Для каждого документа форми-

руется множество U различных слов, входящих в него, и определяется пересечение U и словаря L. Список слов, входящих в пересечение, упорядочивается, и для него вычисляется I-Match сигнатура (хеш-функция SHA1). Два документа считаются похожими, если у них совпадают I-Match сигнатуры [1].

2.2.3 Метод опорных слов

Метод опорных слов, предложенный С. Ильинским, применяется для выявления дубликатов на основе двоичного представления документа. Сначала из индекса по определенному правилу выбирается множество из N слов, называемых опорными. Затем каждый документ представляется N-мерным двоичным вектором, где i-я координата равна 1, если i-е опорное слово имеет в документе относительную частоту выше определенного порога, и равна 0 в противном случае. Этот двоичный вектор называется сигнатурой документа. Два документа похожи, если у них совпадают сигнатуры или совпадает большинство бит. Для каждого слова строится распределение документов по внутридокументной частоте. Проводится несколько итераций оптимизации, в которых максимизируется покрытие документов при фиксированной точности, а затем максимизируется точность при фиксированном покрытии [1].

2.2.4 Расстояние Левенштейна

Для решения задачи поиска близких вариантов фраз и обнаружения парафраза применяется редакционное расстояние. Расстояние Левенштейна — это минимальное количество однозначных операций редактирования (вставка, удаление, замена символа), необходимых для преобразования одной строки в другую. Метрика полезна для поиска близких вариаций текста, включая опечатки и синтаксические ошибки. Однако расстояние Левенштейна не учитывает порядок элементов — два текста с переставленными предложениями будут считаться различными. На больших текстах требует квадратичного времени вычисления $O(n \cdot m)$, что ограничивает его применимость в масштабных системах [7].

2.2.5 Методы на основе предложений

Исследование Зеленкова и Сегаловича включает алгоритмы, основанные на выборе характерных предложений документа. Эти методы решают задачу

быстрого выявления потенциально похожих документов за счет анализа наиболее информативных фрагментов текста.

Long Sent: выбираются 2 самых длинных предложения документа, сцепляются в алфавитном порядке, и вычисляется контрольная сумма CRC32. Алгоритм предполагает, что длинные предложения содержат больше информации и менее вероятно копируются с модификациями. Этот алгоритм показал наивысшую F-меру (0.82) среди всех исследованных методов.

Heavy Sent: вычисляется вес каждого предложения как сумма произведений TF $\cdot$ IDF для всех слов предложения. Выбираются 2 самых тяжелых (информативных) предложения, и для них вычисляется сигнатура. Метод не учитывает порядок предложений в документе [1].

3 Семантические методы и стилометрический анализ

3.1 Методы семантического сопоставления

3.1.1 Встраивания слов: Word2Vec, fastText, GloVe

Для решения задачи выявления парафраза и семантически эквивалентных текстов применяются методы семантического сопоставления. Встраивания слов (word embeddings) преобразуют слова в плотные векторы в пространстве низкой размерности, где семантически похожие слова имеют близкие представления. Word2Vec использует модель Skip-gram или CBOW для обучения на больших текстовых корпусах. fastText расширяет Word2Vec, учитывая информацию о подсловах (символьные n-граммы), что помогает справляться с редкими словами и опечатками. GloVe комбинирует матричную факторизацию с локальным контекстным окном. При сравнении двух документов можно усреднить встраивания всех слов в документе и получить векторное представление всего документа. Косинусная мера близости между такими усредненными векторами характеризует семантическую близость документов [8].

3.1.2 BERT и трансформерные модели

BERT (Bidirectional Encoder Representations from Transformers) — модель глубокого обучения, предварительно обученная на большом количестве текста, которая генерирует контекстные представления слов и предложений. В отличие от статических встраиваний, BERT учитывает контекст слова в предложении, что позволяет более точно захватывать смысл. Sentence-BERT (SBERT) расширяет BERT для создания встраиваний всех предложений, которые прямо оптимизированы для семантической схожести. Русскоязычные варианты BERT (например, RuBERT) обеспечивают качественное представление текстов на русском языке [9].

3.1.3 Siamese и Triplet Loss архитектуры

Siamese network состоит из двух или более копий одной и той же нейронной сети, которые обрабатывают два входа и генерируют представления, сравниваемые для определения схожести. Triplet loss минимизирует расстояние между

якорным примером и похожим примером, одновременно максимизируя расстояние между якорем и непохожим примером. Эти архитектуры эффективны для обучения моделей, которые захватывают метрику сходства, необходимую для детекции плагиата [10].

3.1.4 LSTM и RNN с механизмом внимания

Recurrent Neural Networks (RNN), особенно в форме LSTM (Long Short-Term Memory) или GRU (Gated Recurrent Unit), могут обрабатывать последовательности и захватывать долгосрочные зависимости в тексте. Добавление attention механизма позволяет модели сосредоточиться на наиболее релевантных частях входа при сравнении двух текстов. BiLSTM (bidirectional LSTM) обрабатывает текст в обоих направлениях, улучшая представление. Такие архитектуры учитывают порядок слов в документе, что критично для выявления сложных парафраз [11].

3.2 Модели авторского стиля и стилометрия

3.2.1 Основы стилометрического анализа

Стилометрия — это область, изучающая характеристики письменного стиля, которые отличают одного автора от другого. Предполагается, что у каждого автора есть уникальный стиль, который сохраняется даже при осознанной попытке его изменить. Признаки стилометрии включают:

— среднюю длину предложения, распределение длин предложений;

— часто используемые функциональные слова (предлоги, союзы, артикли);

— частотность части речи (POS tags);

— лексическое разнообразие (type-token ratio);

— использование пунктуации и заглавных букв;

— лексическую плотность;

— читаемость текста.

Стилометрический анализ применяется как для авторского сопоставления (определение авторства текста), так и для интринсивной детекции плагиата [12].

3.2.2 Интринсивная детекция плагиата

Интринсивная (intrinsic) детекция плагиата ищет признаки копирования внутри документа, без привлечения внешних источников. Основной метод решает задачу поиска стилистических разрывов: фрагменты заимствованного текста обычно имеют стиль отличающийся от основного стиля документа. Путем анализа последовательных блоков текста можно выявить участки, где стиль резко меняется. Применяются статистические тесты для определения значимости изменения стилистических параметров. Мешок слов (bag of words) — это представление текста, при котором порядок слов игнорируется, и документ представляется только набором слов с их частотами. Пример мешка слов: для текста «Собака лежит в доме» мешок слов будет {собака: 1, лежит: 1, в: 1, доме: 1}. При интринсивной детекции сравниваются мешки слов последовательных фрагментов одного документа для выявления разрывов [12].

3.3 Анализ и верификация цитирования

3.3.1 Проблемы и типы ошибок в цитировании

Ошибки в цитировании принимают различные формы и являются типичными видами некорректных заимствований:

— *Некорректный источник* — ссылка приведена неправильно, источник не существует, или указывает на совершенно иное произведение;

— *Отсутствующая информация в источнике* — утверждение, приписываемое источнику, в нем не содержится, или содержится в другом контексте;

— *Неверная страница* — указан неправильный диапазон страниц, что затрудняет верификацию цитаты;

— *Отсутствует источник* — информация используется без ссылки на источник;

— *Ghost citations* (призрачные цитаты) — источники цитируются по вторичным источникам без прямого обращения к первичному источнику.

Для решения задачи выявления этих ошибок используется автоматизированная верификация источников [3].

3.3.2 Автоматическая парсинг и извлечение метаданных

Для проверки корректности источников необходимо автоматизированно извлекать метаданные из PDF и других документов. Системы парсинга, такие как CERMINE, GROBID и PDFDataExtractor, используют компьютерное зрение и обработку естественного языка для распознавания структуры документа и извлечения текста, авторов, названия, года публикации, DOI (Digital Object Identifier), диапазонов страниц, и списка литературы. Извлеченные метаданные затем сопоставляются с библиографическими базами данных [13].

3.3.3 Проверка контекста цитирования и соответствия источнику

Анализ контекста цитирования включает проверку того, действительно ли утверждение в цитирующем тексте соответствует содержимому исходного документа. Для решения этой задачи применяются методы NLP:

— выделение синтаксических единиц вокруг ссылки (несколько предложений до и после цитирования);

— извлечение соответствующих фрагментов из исходного документа, соответствующих ключевым термам из цитирующего предложения;

— сравнение семантической схожести между контекстом ссылки и релевантными фрагментами источника с использованием косинусной меры близости;

— оценка соответствия на основе порога схожести для определения, является ли цитирование корректным [14].

4 Экспериментальное исследование и оценка эффективности

4.1 Метрики оценки качества

В качестве основных показателей качества работы алгоритмов используются полнота, точность и F-мера. Для оценки эффективности систем обнаружения используются следующие метрики:

— *Точность* — доля верно обнаруженных случаев плагиата из всех случаев, помеченных как плагиат. Высокая точность означает низкий процент ложных срабатываний.

— *Полнота* — доля верно обнаруженных случаев из всех действительно существующих случаев плагиата. Высокая полнота означает, что система пропускает мало реальных случаев плагиата.

— *F-мера* — гармоническое среднее точности и полноты, дающее единую метрику качества работы алгоритма.

4.2 Результаты сравнительного исследования

Сравнение проводилось на коллекции русскоязычных веб-документов РОМИП (Russian Information Retrieval Evaluation Seminar) путем поиска заимствований в текстах научных статей. Тексты включали как полные дубликаты, так и парафразы и частичные переиспользования. Система проверяла каждый алгоритм на единообразных наборах тестовых пар документов.

Алгоритм	**Полнота**	**Точность**	**F-мера**
Long Sent	0.84	0.80	0.82
TF	0.60	0.94	0.73
Opt Freq	0.59	0.94	0.73
TF*IDF	0.59	0.95	0.73
Heavy Sent	0.62	0.86	0.72
TF-IDF	0.54	0.96	0.69
Lex Rand	0.50	0.97	0.66
Descr Words	0.44	0.77	0.56
Log Shingles	0.39	0.97	0.56
Megashingles	0.36	0.91	0.51
MD5	0.23	1.00	0.38

Таблица 4.1 — Сравнение метрик качества алгоритмов обнаружения заимствований в порядке убывания рейтинга

Результаты показывают, что алгоритм выбора длинных предложений (Long Sent) показал наилучшие результаты по F-мере, сочетая высокую полноту (0.84) и точность (0.80). Лексические методы (TF, TF*IDF, Opt Freq) показывают высокую точность (0.94-0.97), но умеренную полноту (0.54-0.60). Метод мегашинглов уступает более простым алгоритмам, что объясняется строгими требованиями к совпадению супершинглов. Методы, основанные на шинглах и мегашинглах, хорошо подходят для масштабных систем, так как требуют минимальной памяти и времени, но их точность и полнота ниже, чем у методов на основе предложений [1].

Дополнительное сравнение может быть проведено по следующим критериям: время выполнения алгоритма на документах разного размера; потребление оперативной памяти; устойчивость к переводному плагиату; обработка текстов на разных языках; качество выявления парафраза; обнаружение частичных заимствований; способность к обработке кода и структурированных данных.

ЗАКЛЮЧЕНИЕ

Проблема обнаружения некорректных заимствований требует применения интегрированного подхода, сочетающего методы из различных классов:

— высокоскоростные методы обнаружения (отпечатки, LSH, шинглы) для быстрого отсеивания явно неподозрительных документов;

— статистические и лексико-семантические методы (TF · IDF, опорные слова, расстояния, N-граммы) для детального сравнения;

— современные нейросетевые методы (BERT, Siamese networks, LSTM) для выявления сложных парафраз и переводов;

— стилометрический анализ для выявления внутридокументных разрывов стиля;

— автоматизированные системы верификации источников и метаданных.

Экспериментальные исследования показывают, что ни один единственный метод не может быть универсален. Комбинация различных подходов, правильно настроенных и взвешенных в зависимости от контекста, обеспечивает высокую точность и полноту при обнаружении различных типов нарушений академической этики.

Наилучшие результаты по F-мере (0.82) показывает алгоритм Long Sent, основанный на выборе длинных предложений. Лексические методы обеспечивают высокую точность (до 0.97), но умеренную полноту. Методы на основе шинглов и мегашинглов эффективны для масштабных систем, но требуют тщательной настройки параметров. Дальнейшее развитие в этой области идет в направлении использования более мощных языковых моделей, лучшей интеграции методов верификации ссылок, разработки более эффективных алгоритмов кросс-языкового поиска плагиата, и создания специализированных систем для различных доменов и типов документов [1].

ЗАКЛЮЧЕНИЕ

Проблема обнаружения некорректных заимствований требует применения интегрированного подхода, сочетающего методы из различных классов:

— высокоскоростные методы обнаружения (отпечатки, LSH, шинглы) для быстрого отсеивания явно неподозрительных документов;

— статистические и лексико-семантические методы (TF · IDF, опорные слова, расстояния, N-граммы) для детального сравнения;

— современные нейросетевые методы (BERT, Siamese networks, LSTM) для выявления сложных парафраз и переводов;

— стилометрический анализ для выявления внутридокументных разрывов стиля;

— автоматизированные системы верификации источников и метаданных.

Экспериментальные исследования показывают, что ни один единственный метод не может быть универсален. Комбинация различных подходов, правильно настроенных и взвешенных в зависимости от контекста, обеспечивает высокую точность и полноту при обнаружении различных типов нарушений академической этики.

Наилучшие результаты по F-мере (0.82) показывает алгоритм Long Sent, основанный на выборе длинных предложений. Лексические методы обеспечивают высокую точность (до 0.97), но умеренную полноту. Методы на основе шинглов и мегашинглов эффективны для масштабных систем, но требуют тщательной настройки параметров. Дальнейшее развитие в этой области идет в направлении использования более мощных языковых моделей, лучшей интеграции методов верификации ссылок, разработки более эффективных алгоритмов кросс-языкового поиска плагиата, и создания специализированных систем для различных доменов и типов документов [1].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Зеленков Ю.Г., Сегалович И.В. Сравнительное исследование методов определения нечетких дубликатов для Web-документов // Труды 9-й Всероссийской научной конференции RCDL2007. — Переславль-Залесский, 2007.

2. Методы анализа и поиска заимствований в тексте // URL: https://cyberleninka.ru/article/n/metody-analiza-i-poiska-zaimstvovaniy-v-tekste

3. Некорректные заимствования в диссертациях: способы их обнаружения // URL: https://cyberleninka.ru/article/n/nekorrektnye-zaimstvovaniya-v-dissertatsiyah-sposoby-ih-obnaruzheniya

4. Implementation of Winnowing Algorithm Based K-Gram to Identify Plagiarism // MATEC Web of Conferences. — 2018. — Vol. 154.

5. A Robust Document Identification Framework through f-BP Fingerprint // Applied Sciences. — 2021. — Vol. 7, No. 8.

6. Research on Text Similarity Measurement Hybrid Algorithm with TF-IDF Method // Applied Mathematics. — 2022.

7. Levenshtein Distance, Sequence Comparison and Biological Applications // MIT Open Courseware. — 2020.

8. On the Sentence Embeddings from BERT for Semantic Textual Similarity // Proceedings of EMNLP. — 2020.

9. Phrase-BERT: Improved Phrase Embeddings from BERT with an Application to Corpus Exploration // Proceedings of EMNLP. — 2021.

10. Hybrid approach of BERT extraction with deep Siamese Bi-LSTM for semantic text similarity // Scientific Reports. — 2022.

11. An LSTM-based Plagiarism Detection via Attention Mechanism and a Population-Based Approach for Pre-training Parameters with Imbalanced Dataset // arXiv preprint. — 2021.

12. Intrinsic Plagiarism Detection // Proceedings of the ACL Workshop. — 2015.

13. AutoIE: Automated Information Extraction from Scientific Literature // arXiv preprint. — 2024.

14. CiteCheck: Accurate Citation Faithfulness Detection via Semantic Graph Representation Learning // arXiv preprint. — 2025.