**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И

ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

Направление: 09.03.03 – «Прикладная информатика»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**Разработка алгоритма обработки естественного языка при помощи машинного обучения с использованием нейросетей BERT**

**Работа завершена:**

Студент 4 курса

группы 09-852

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. \_\_\_\_\_\_\_\_\_\_\_\_ Саидмуродов С.С.

**Работа допущена к защите:**

Научный руководитель

старший преподаватель

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. \_\_\_\_\_\_\_\_\_\_\_\_ Прокопьев Н.А.

Заведующий кафедрой

канд. экон. наук, доцент

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. \_\_\_\_\_\_\_\_\_\_\_\_ Вахитов Г.З.

Казань – 2022

**Оглавление**

[Введение 5](#_Toc98361617)

[1. Формирование требований 8](#_Toc98361618)

[1.1. Изучение предметной области машинного обучения и нейронных сетей 8](#_Toc98361619)

[1.2. Анализ подходов к обработке естественного языка 9](#_Toc98361620)

[1.2.1. BoW 9](#_Toc98361621)

[1.2.2. TF-IDF 10](#_Toc98361622)

[1.2.3. Word2Vec 11](#_Toc98361623)

[1.2.4. GloVe 11](#_Toc98361624)

[1.3. Техническое задание 11](#_Toc98361625)

[2. Проектирование платформы для обучения нейросетевой модели 12](#_Toc98361626)

[2.1. Используемые модели нейронных сетей 12](#_Toc98361627)

[2.2. Формализация алгоритмов основного анализа текста 12](#_Toc98361628)

[2.3. Структура платформы для обучения нейросетевой модели 12](#_Toc98361629)

[3. Экспериментальная реализация алгоритма анализа текста 13](#_Toc98361630)

[3.1. Подготовка данных для обучения 13](#_Toc98361631)

[3.2. Реализация платформы для обучения нейросетевой модели 13](#_Toc98361632)

[3.3. Результаты машинного обучения 13](#_Toc98361633)

[4. Тестирование по результатам обучения модели для основного анализа текста 14](#_Toc98361634)

[4.1. Экспериментальная оценка результатов обучения 14](#_Toc98361635)

[4.2. Целевой оценочный тест реализации алгоритма 14](#_Toc98361636)

[Заключение 15](#_Toc98361637)

[Список использованных источников 16](#_Toc98361638)

[Введение 5](#_Toc98361572)

[1. Формирование требований 8](#_Toc98361573)

[1.1. (Păiş, et al., 2021) (Păiş, et al., 2021)Изучение предметной области машинного обучения и нейронных сетей 8](#_Toc98361574)

[1.2. Анализ подходов к обработке естественного языка 9](#_Toc98361575)

[1.2.1. BoW 9](#_Toc98361576)

[1.2.2. TF-IDF 10](#_Toc98361577)

[1.2.3. Word2Vec 11](#_Toc98361578)

[1.2.4. GloVe 11](#_Toc98361579)

[1.3. Техническое задание 11](#_Toc98361580)

[2. Проектирование платформы для обучения нейросетевой модели 12](#_Toc98361581)

[2.1. Используемые модели нейронных сетей 12](#_Toc98361582)

[2.2. Формализация алгоритмов основного анализа текста 12](#_Toc98361583)

[2.3. Структура платформы для обучения нейросетевой модели 12](#_Toc98361584)

[3. Экспериментальная реализация алгоритма анализа текста 13](#_Toc98361585)

[3.1. Подготовка данных для обучения 13](#_Toc98361586)

[3.2. Реализация платформы для обучения нейросетевой модели 13](#_Toc98361587)

[3.3. Результаты машинного обучения 13](#_Toc98361588)

[4. Тестирование по результатам обучения модели для основного анализа текста 14](#_Toc98361589)

[4.1. Экспериментальная оценка результатов обучения 14](#_Toc98361590)

[4.2. Целевой оценочный тест реализации алгоритма 14](#_Toc98361591)

[Заключение 15](#_Toc98361592)

[Список использованных источников 16](#_Toc98361593)

Введение

Для понимания текста людьми и программами искусственного интеллекта (ИИ) часто требуется правильное использование заглавных букв и знаков препинания. Простые предложения, состоящие из нескольких слов, и люди, и ИИ могут читать и обрабатывать, даже если использование заглавных букв или знаков препинания отсутствует или неправильно. Например, в случае голосовых команд обработка обычно выполняется только для распознанных слов нижнего регистра. Однако, если объем текста, подлежащего анализу, увеличивается, например, до целых абзацев или страниц, то даже для человека быстрое понимание его смысла становится сложной задачей. Это было изучено Джонсом и др. (2003), которые проанализировали влияние заглавных букв и пунктуации на читаемость расшифровки речи в текст.

Ранние работы рассматривали пунктуацию только как подсказки с точки зрения читателя к возможным просодическим характеристикам и паузам текста (Markwardt, 1942). Нанберг (1990) утверждает, что пунктуация играет гораздо большую роль. Кроме того, знаки препинания классифицируются как разграничивающие, разделяющие и устраняющие неоднозначность. Некоторые метки, такие как запятая, могут принадлежать к нескольким категориям, поскольку они могут выполнять несколько ролей. Джонс (1994) доказывает, что «для более длинных предложений реального языка грамматика, использующая знаки препинания, значительно превосходит аналогичную грамматику, которая их игнорирует». Основываясь на этом и других подобных выводах, современные языковые модели считают пунктуацию частью своего словарного запаса. Сюда входят новейшие модели, такие как BERT, ELMo, OpenAI GPT-2 и GPT-3.

Алгоритмы обработки естественного языка (NLP – Natural language processing), такие как распознавание именованных сущностей (NER – Named entitiy recognition), идентификация части речи, анализ зависимостей, машинный перевод (MT – Machine translation), используют заглавные буквы в качестве признаков обрабатываемого в данный момент слова, в то время как пунктуация используется. как признаки для соседних слов. Например, Stanford Named Entity Recognizer рассматривает признаки на основе формы слова. Это означает построение представления слова на основе типа символов, встречающихся в слове. Было предложено несколько алгоритмов представления формы слова, но общая идея состоит в том, чтобы закодировать прописную букву определенным символом, скажем, «X», строчную букву «x» и цифру «d». В этом случае слово типа «McDonald» станет «ХхХхххх». Работа любых таких алгоритмов возможна только в том случае, если слова правильно представлены в виде прописных и строчных букв.

Особое внимание следует уделить системам автоматического распознавания речи (ASR – Automatiс speech recognition). Первичный вывод таких систем обычно состоит из необработанного текста с использованием одного и того же регистра (нижнего, либо верхнего регистра) и без знаков препинания. В таких ситуациях перед применением дальнейших алгоритмов NLP требуется дополнительная предварительная обработка, чтобы восстановить правильный регистр букв и пунктуацию. Их иногда называют «богатыми транскрипциями». Одна из первых инициатив, касающихся автоматической расширенной транскрипции разговорной речи, началась в 2002 году в контексте программы DARPA «Эффективное, доступное повторное использование речи в текст» (EARS – Effective, Affordable, Reusable Speech-to-text), целью которой было улучшение уровня развития алгоритмов обработки языка. С этой целью NIST (National institute of Standards and Technology) выпустил серию обширных наборов данных для оценки транскрипции, чтобы помочь в оценке таких систем.

Несмотря на то, что большой объем данных, требующих восстановления заглавных букв и пунктуации, поступает из систем ASR, необходимо также учитывать и другие источники. Миллер и др. (2000) идентифицируют другие источники шума в виде текста, полученного с помощью оптического распознавания символов (OCR), или в некоторых газетных статьях. В этих случаях отсутствие надлежащей буквы или пунктуации затрагивает не весь текст, а его части. В случае OCR некоторые знаки препинания могут быть не распознаны, в то время как в случае некоторых статей первое предложение или абзац могут быть написаны только заглавными буквами. Кроме того, в случае коротких текстовых сообщений (SMS), чатов, твитов или других действий в микроблогах люди также могут игнорировать правильный регистр и пунктуацию.

Одна из трудностей, при создании человеко-компьютерных интерфейсов с использованием естественного языка, с которыми приходится сталкиваться, связана с непоследовательным использованием пользователем пунктуации и использования заглавных букв. В этом контексте многие подходы пытаются скрыть проблему, удаляя все знаки препинания и заглавные буквы как из данных обучения, так и из входных данных, полученных во время работы. Кроме того, Coniam (2014) также проанализировал вывод чат-ботов с точки зрения человека, использующего эти программы для изучения английского как второго языка. Он смог определить проблемы с заглавными буквами и пунктуацией даже в произведенном тексте. Тем не менее, он утверждает, что для коротких предложений, создаваемых чат-ботами, «переход на английский язык за счет все более широкого распространения текстовых сообщений делает спорным вопрос о том, можно ли считать эти проблемы важными в наши дни».

# Формирование требований

## Изучение предметной области машинного обучения и нейронных сетей

Итак, разберемся что же такое машинное обучение. В Википедии можно найти следующее определение: «Машинное обучение — класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение за счёт применения решений множества сходных задач». Так же можно найти более современное определение машинного обучения, данное Томом Митчеллом: «A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E». Различают два основных типа обучения:

* Дедуктивное обучение, или обучение с учителем, предполагает формализацию знаний экспертов и их перенос в компьютер в виде базы знаний
* Обучение по прецедентам, или индуктивное обучение (обучение без учителя), основано на выявлении эмпирических закономерностей в собранных данных.

В первом случае важна правильно составленная и размеченная обучающая выборка данных. Для этого необходимо выделить важные признаки данных, и заранее определить правильный ожидаемый результат для каждого набора данных. Дедуктивное обучение принято относить к области экспертных систем, поэтому иногда под машинным обучением понимают дедуктивное обучение.

Второму случаю присуще обучение на основе неразмеченных данных, и целью алгоритмов является определение признаков данных и их приоритетов. Многие методы, применяемые в обучении без учителя тесно связаны с извлечением информации и анализом данных.

Так как машинное обучение является достаточно крупной областью науки, далее более детально будут рассмотрены только подходы к обработке естественного языка, подробнее с машинным обучением можно ознакомится в приведенных источниках.

## Анализ подходов к обработке естественного языка

В данной работе рассмотрим подходы, основанные на машинном обучении. Далее будут коротко описаны методы решения задач обработки естественного языка с применением таких технологий как BoW, TF-IDF, Word2Vec, GloVe, fastText, GRU, LSTM, BERT и ELMO.

### BoW

*BoW* (от англ. Bag of words – мешок слов) является наиболее простым и часто встречающимся векторным представлением. BoW работает следующим образом: на вход подается анализируемый текст, для каждого предложения строится вектор с длиной n-элементов, где n – это количество уникальных токенов в тексте. Каждый элемент вектора хранит в себе частоту вхождения того или иного токена в предложении. Токеном может являться как слово, так и n-грамма.

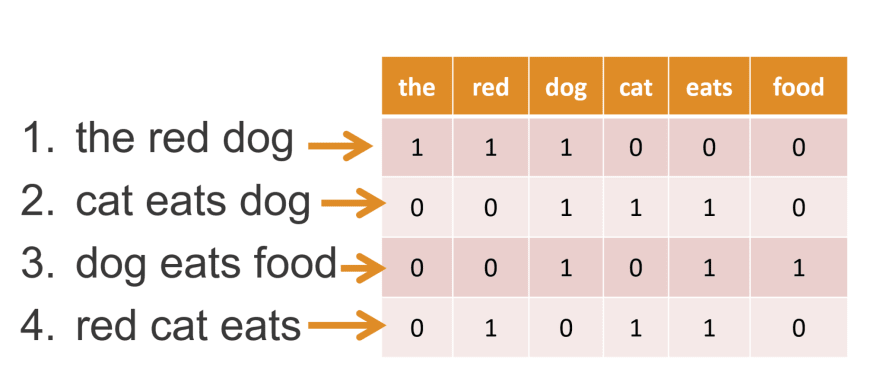


Рисунок 1. Пример векторного представления текста по методике BoW, при этом исполуется 3-граммная модель

Представления BoW часто используются в методах классификации документов, где частота каждого слова, двухсловного или трехсловного слова является полезной функцией для обучения классификаторов. Одна из проблем с представлениями мешка слов заключается в том, что они не кодируют никакой информации относительно значения данного слова. В BoW вхождения слов имеют одинаковый вес независимо от того, как часто и в каком контексте они встречаются. Однако в большинстве задач NLP одни слова более важны, чем другие.

### TF-IDF

*TF-IDF* (от англ. TF — term frequency, IDF — inverse document frequency) — статистическая мера, которая используется для оценки значимости слова в контексте документа, являющегося частью коллекции документов или корпуса. Вес некоторого слова пропорционален частоте употребления этого слова в документе и обратно пропорционален частоте употребления слова во всех документах коллекции. Мера TF-IDF находится путем перемножения частоты слова в документе и обратной частоты слова в корпусе документов:

Для нахождения частоты слова используют следующую формулу:

где – число вхождений слова в документ, а знаменатель – общее число слов в документе. Обратная частота документа – инверсия частоты, с которой некоторое слово встречается в документах коллекции находится следующей формулой:

где – число документов, – число документов из коллекции , в которых встречается слово (когда )

### Word2Vec

*Word2Vec* – совокупность моделей на основе нейронных сетей, используемых для получения векторных представлений слов. Используется для изучения семантики естественных языков. Word2Vec разработан исследовательской группой Google под руководством чешского ученого Томаса Миколова в 2013 году. Word2Vec включается в себя две архитектуры моделей: Continuous bag-of-words (CBOW) для предсказания слова наиболее подходящего для данного контекста (при этом порядок слов не важен как и в BoW) и Continuous skip-gram используется для получения набора схожих слов со входящим словом

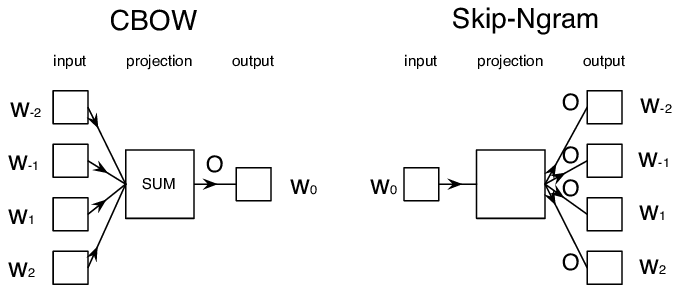


Рисунок 2. Наглядное сравнение архитектур Word2Vec

### GloVe

GloVe (от англ. Global Vectors – глобальные векторы) – еще одна языковая модель, основанная на индуктивном обучении. В этой модели слова отображаются в векторном пространстве, где расстояние между словами определяет их семантическое сходство. GloVe создавался как проект с открытым исходным кодом в Стэнфорде в 2014 году. Эту модель можно использовать для поиска взаимосвязей между такими словами, такими как синонимы, отношения между компанией и продуктом, почтовые индексы и города и т. д. Хотя векторы GloVe быстрее обучаются, ни GloVe, ни Word2Vec, как показала практика, не дают однозначно лучших результатов, при тестировании на одних и тех же наборах данных.

### fastText

## Техническое задание

Asdada

# Проектирование платформы для обучения нейросетевой модели

Asdada

## Используемые модели нейронных сетей

Asdada

## Формализация алгоритмов основного анализа текста

фвфвф

## Структура платформы для обучения нейросетевой модели

Текст рыба

# Экспериментальная реализация алгоритма анализа текста

Текст рыба

## Подготовка данных для обучения

Текст рыба

## Реализация платформы для обучения нейросетевой модели

Текст рыба

## Результаты машинного обучения

рыыыба

# Тестирование по результатам обучения модели для основного анализа текста

Рыба

## Экспериментальная оценка результатов обучения

Рыба

## Целевой оценочный тест реализации алгоритма

Заключение

Рыба

# Список использованных источников

**Mitchell Tom M.** Machine Learning [Книга]. - [б.м.] : McGraw Hill, 1997. - стр. 2.

**Păiş Vasile и Tufis Dan** Capitalization and Punctuation Restoration: a Survey [В Интернете] // ResearchGate. - ResearchGate, 21 ноябрь 2021 г.. - 15 январь 2022 г.. - https://www.researchgate.net/publication/356456267\_Capitalization\_and\_Punctuation\_Restoration\_a\_Survey.

Машинное обучение [В Интернете] // Википедия. Свободная энциклопедия.. - 23 январь 2022 г.. - 13 февраль 2022 г.. - https://ru.wikipedia.org/wiki/Машинное\_обучение.