Министерство науки и высшего образования Российской Федерации
ФГАОУ ВО «Севастопольский государственный университет»
Институт информационных технологий

Кафедра «Информационная безопасность»

РАСЧЁТНО-ГРАФИЧЕСКАЯ РАБОТА

по теме

«Исследование методов векторизации текста
и извлечения признаков»

по дисциплине

«Защита программ и данных»

Выполнил: студент гр. ИБ/б-21-1-о
Проскуряков К. А.
Защитил с оценкой: _____
Принял: доцент Лихолоб П. Г.

Севастополь
2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Машинное обучение предоставляет возможность быстро и эффективно решать как внешние, так и внутренние задачи, которые возникают перед бизнесом. С каждым днем чат-боты становятся всё более совершенными, и отличить поведение программы от человеческого становится всё сложнее и сложнее.

С помощью моделей машинного обучения и внедрения их в код программы осуществляется генерация уникальных ответов в чат-ботах. Для создания эффективной модели необходимо осуществить предварительную обработку текста, а именно токенизацию, лемматизацию, удаление стоп-слов (союзов, предлогов, междометий и т. д.) и векторизацию.

Одним из механизмов классификации текста является векторизация. Для обработки используются следующие алгоритмы:

– *One-hot encoding*;

– *TD-IDF*;

– *CountVectorizer*;

– *word2vec*.

Целью работы является проверка алгоритмов векторизации на практике и сравнение их эффективности путем использования полученного корпуса текстов в методе машинного обучения.

В задачи работы входит:

1) Дать определение используемым в ходе работы понятиям;
2) Применить к полученному корпусу текстов алгоритмы векторизации;
3) Использовать полученный результат в качестве моделей для алгоритма машинного обучения;
4) Сравнить полученный результат.

Объектом исследования является текст, предметом – методы векторизации текста.

Работа изложена на INSERT страницах основного текста, включающего INSERT рисунков, INSERT таблиц, список литературных источников из INSERT наименований, INSERT приложений.

1 ОЗАГЛАВИТЬ

1.1 Правовое поле использованных библиотек

1.1.1 Библиотеки получения корпуса

– *string* – стандартная библиотека *Python*. Распространяется по лицензии *PSF*;

– *re* – стандартная библиотека *Python*. Распространяется по лицензии *PSF*;

– *SpaCy* – библиотека обработки естественного языка. Распространяется по лицензии *MIT*.

1.1.2 Библиотеки векторизации

– *sklearn* – библиотека машинного обучения. Распространяется по лицензии *BSD-3*;

– *gensim* – библиотека обработки естественного языка и информационного поиска. Распространяется по лицензии *LGPL-2.1*.

1.2 Глоссарий

Текст – это некоторая последовательность предложений, имеющая логическую последовательность и сообщающая какую-либо информацию.

Корпус текстов – это подобранная и обработанная по определенным правилам совокупность текстов, используемая для исследования языка.

Токен – это текстовая единица (слово, словосочетание и т. д.).

1.3 Обозначение входных и выходных данных

pass

1.4 Математические модели методов векторизации

1.4.1 One-hot encoding

pass

1.4.2 TD-IDF

Общая формула показателя IDF выглядит следующим образом:

$$IDF(t, D) = \log \frac{|D| + 1}{DF(t, D) + 1}$$

где

– D – количество документов в корпусе,

– $DF(t, D)$ – количество документов, в которых встречается слово.

Так, если слово встречается во всех документах, то $IDF = 0$. В итоге,

$$TFIDF = IDF \cdot TF$$

1.4.3 CountVectorizer

pass

1.4.4 word2vec

pass

2 ОЗАГЛАВИТЬ

2.1 Методы предварительной обработки и фильтрации

2.1.1 Токенизация

Токенизация представляет из себя процесс разбиения больших участков текста на абзацы, предложения и слова. Данная операция не требует сторонних библиотек и может быть реализована с помощью стандартных модулей языка *Python*.

```
import string
import re

# Считывание текста
text = ''.join([''.join([line for line in open(f'./text-data/{i}.txt', 'r')]) for i in range(1, 4 + 1)])

# Удаление знаков пунктуации
text_without_punctuation = re.sub(f'[{string.punctuation}\\n—]', '', text)

# Разделение полученного текста на слова
words = text_without_punctuation.split(' ')

# Перевод всех токенов в нижний регистр
tokenized_words = [word.lower() for word in words]
```

Рисунок 1 – Токенизация стандартными библиотеками языка Python

2.1.2 Лемматизация

Лемматизация – это процесс приведения слова к его словарной (изначальной) форме. Данная операция требует подключения сторонних библиотек. На рис. 2 представлен процесс лемматизации с помощью библиотеки *SpaCy*.

```
import spacy

nlp = spacy.load('ru_core_news_sm')

sentence = 'Съешь ещё этих мягких французских булок, да выпей чаю.'

document = nlp(sentence)
```

Рисунок 2 – Лемматизация библиотекой SpaCy

2.1.3 Удаление шумовых слов

Под шумовыми словами подразумевают слова, не несущие смысловой нагрузки (междометия, союзы и т. д.). Операция может быть выполнена средствами языка программирования.

```
stop_words = ['да', 'ещё', 'этих', 'мягких']

sentence = 'Съешь ещё этих мягких французских булок, да выпей чаю.'
sentence_without_stop_words = ''

for word in sentence.split(' '):
    if word not in stop_words:
        sentence_without_stop_words += word + ' '
```

Рисунок 3 – Удаление стоп-слов

3 ОЗАГЛАВИТЬ

3.1

ЗАКЛЮЧЕНИЕ