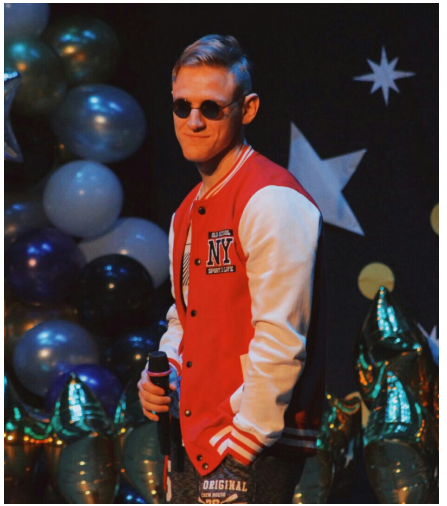
Talent Case Contest 2023

LeTeam	
Семыкин Владислав	**Смирнов Илья**
	

Опыта в ML, DL, DS и прочем схожем нет

Содержание

Подход к решению задачи

1. Посмотреть на решение проблемы с человеческой точки зрения.

2. Интерпретация в понятный для программы вид.

3. Исследование разработанного метода.

Принцип основан на сравнении символов, который реализован несколькими операциями.

Обоснование выбора метода получения данных

При поиске оптимального относительно скорости получения данных были оценены 2 метода получения данных:
1) Bash
2) Python

Была взята выборка из 10’000 измерений для Python-скрипта и 1’000 измерений для Bash-скрипта.

Результаты исследований

Python Exec Time (seconds)	Bash Exec Time (seconds)
0.0014512538909912 1	0.048987
0.0036885738372802 7	0.034631
0.0011804103851318 4	0.034303
0.0011086463928222 7	0.035065
0.0012328624725341 8	0.034356
0.0013146400451660 2	0.043244
0.0013713836669921 9	0.037732
0.001143693923950 2	0.034863
0.001189947128295 9	0.03474

На основе результатов измерения скорости получения данных и их записи в .csv файл, представленных в таблице слева, было принято решение использовать наиболее быстрый способ получения при помощи Python-скрипта, который наиболее быстро справлялся с поставленной задачей.

Более подробно ознакомиться с результатами можно в директории *Performance tests on extracting/*

Установка зависимостей

Перед началом реализации метода было принято решение проверить наличие некоторых используемых зависимостей и их установки при надобности, см. исходный код файла `inst_dependencies.py`

```
<loveit@fedora Talent Case Contest 2023>$ py main.py
re is already installed
json is already installed
time is already installed
psutil is already installed
Enter name of the file >>
```

Реализация алгоритма

- Получение каждого предложения с его идентификатором из датасета в формате JSON.
- Преобразование каждого символа строки в нижний регистр.
- Удаление пробелов, знаков препинания, специальных символов и всего того, что не является русскими буквами.
- Удаление стоп-слов.
- Сравнение полученных последовательностей символов между собой для нахождения одинаковых.
- Перечисление ID и соответствующих им предложений, имеющих общий смысл.

Результаты

```
<loveit@fedora Talent Case Contest 2023>$ py main.py
re is already installed
json is already installed
time is already installed
psutil is already installed
Enter name of the file >> sample.json
The same by meaning sentences:
[2, 15, 150]:
2 - Почему она так со мной поступает?
15 - Почему она так с ней поступает?
150 - Почему он так со мной поступает?
[3, 266]:
3 - Никто туда больше не ходит.
266 - Никто больше туда не ходит.
[4, 13]:
4 - У него с собой не было тогда денег.
13 - У него тогда не было с собой денег.
```

```
[328, 379]:
328 - Я знаю, что это для тебя важно.
379 - Я знаю, что для тебя это важно.
[338, 347]:
338 - Том уже выпил три чашки кофе.
347 - Том выпил уже три чашки кофе.
[350, 374]:
350 - Я бы хотел куриного супа.
374 - Я хотел бы куриного супа.
[388, 405]:
388 - Кен был вчера дома?
405 - Кен вчера был дома?
Memory usage: 14.61 Mb
Execution time: 31.98 ms
```

Преимущества и недостатки

Безотказная работа на предложениях, сформированных методом перестановки слов.

```
[78, 123]:
78 - Мы их только что нашли.
123 - Мы только что их нашли.
[85, 259]:
85 - Том не может сейчас подойти к телефону.
259 - Том сейчас не может подойти к телефону.
[89, 299]:
89 - Какое это имеет отношение к школе?
299 - Камое это имеет отношение к школе?
[90, 404]:
90 - Я не хотел ввести никого в заблуждение.
404 - Я никого не хотел ввести в заблуждение.
[92, 187]:
92 - Том нам ничего не дал.
187 - Том ничего нам не дал.
```

Некорректная работа алгоритма в случае подачи на вход перемешанных букв в предложениях.

```
<loveit@fedora Talent Case Contest 2023>$ py main.py
re is already installed
json is already installed
time is already installed
psutil is already installed
Enter name of the file >> test.json
The same by meaning sentences:
[1, 2]:
1 - Привет, мир!
2 - ирП! ,теривм
Memory usage: 13.75 Mb
Execution time: 0.61 ms
```