|  |  |  |
| --- | --- | --- |
|  | **Софийски университет „Св. Кл. Охридски”**  Факултет по математика и информатика |  |

**Курсов Проект**

на тема: „Търсене в текстове на песни с емоция”

Студент: **Мартин Бисеров Гергиев** Ф.Н. **25907**

Курс: „Извличане на информацията“, Учебна година: 2018/19

Преподаватели: **проф. Иван Койчев**

=================================

Декларация за липса плагиатство:

* Плагиатство е да използваш, идеи, мнение или работа на друг, като претендираш, че са твои. Това е форма на преписване.
* Тази курсова работа е моя, като всички изречения, илюстрации и програми от други хора са изрично цитирани.
* Тази курсова работа или нейна версия не са представени в друг университет или друга учебна институция.
* Разбирам, че ако се установи плагиатство в работата ми ще получа оценка “Слаб”.

28.1.19 г. Подпис на студента:

**Съдържание**

[**1**](#_gjdgxs) **УВОД 2**

[**2**](#_30j0zll) **РАЗПОЗНАВАНЕ НА ЕМОЦИЯ В ТЕКСТ 2**

[**3**](#_1fob9te) **ПРОЕКТИРАНЕ 2**

[**4**](#_3znysh7) **РЕАЛИЗАЦИЯ, ТЕСТВАНЕ/ЕКСПЕРИМЕНТИ 2**

[4.1](#_2et92p0) Използвани технологии, платформи и библиотеки 2

[4.2](#_tyjcwt) Реализация 2

[**5**](#_3dy6vkm) **ЗАКЛЮЧЕНИЕ 2**

[**6**](#_1t3h5sf) **ИЗПОЛЗВАНА ЛИТЕРАТУРА 2**

# **Увод**

Проекта представлява търсене по конкретни думи от текстове на песни. Но освен това те могат да се сортират по дадена емоция. Такъв вид търсачка би била полезна на различни видове хора да могат да намират песни спрямо настроението им.

Трябва да се реализира потребителски интерфейс, в който да може да се въвежда текст и да се избира емоция от (“щастие”, “тъга”, “спокойствие” и “ярост”). Да се показват резултати на песни близки до даденият низ за търсене и да се сортират по избрана емоция.

# Разпознаване на емоция в текст

За момента разпознаването на емоция не е напълно решен проблем, но има добри разработки, които се справят добре с него.

Най - стандартните и начални подходи са били чрез използването на речник с емоционален характер за определяне към коя категория принадлежи дадена дума. От тук нататък се използват статистически методи като Наивен Бейсов Класификатор или “Tf-idf”.

Друг интересен подход е чрез трениране на прост модел върху отбелязани думи за конкретни песни в сайтове. Чрез този метод е създадената една от малкото бази от данни с отбелязани емоции на текстове от песни. Нарича се “MoodyLyrics4Q”.

Друг подход е чрез изследване на възбудата и умереността на даден текст на песен, чрез използването на речници предназначени за това. После се тренира прост модел и се отсяват тези които са възможно най - определени. Чрез този метод е създадена базата от данни “MoodyLyrics”.

# **Проектиране**

Системата е разделена на няколко модула:

* Създаване на модел за класификация на текст на песен към дадена емоция
* Извличане сравнително голям набор от текстове на песни от интернет
* Класифициране на текстовете чрез модела
* Индексиране на текстовете за възможност за търсене по тях
* Потребителски интерфейс

# **Реализация, тестване/експерименти**

## ***Използвани технологии, платформи и библиотеки***

Използвани технологии са:

* “Python jupyter notebook” - за проектиране
* “Bert” модел за кодиране на информация от текст
* “google colab” и “keras” с основа “tensorflow” - за създаване на модел за класификация
* “scrapy” - за извличане на информация от сайтове
* “Woosh” - инструмент за индексиране на текстове
* “Flask” - за сървър
* “Angular 6” - за потребителски уеб интерфейс

**Забележка**: За трениране на модела за класификация, който представлява невронна мрежа съм използвал горепосочените бази от данни “MoodyLyrics4Q” и “MoodyLyrics”.

## ***Реализация/Провеждане на експерименти***

* + 1. **Създавне на модел за класификация:**

Комбиниране на двете бази от данни като се махат повтарящите. Използват се за обучаващо множество. Разделяне на текстовете на по изречения по 25 думи. Всяко изречение се кодира чрез “Bert” езика. Трениране на невронна мрежа от 2 слоя на линейни слоя на “keras” върху облак на “google colab”. Резултат е във формата на четворка с процентната вероятност дадения текст да изразява съответстващата на индекса категория. Финално оценяване на текста. Средно аритметично на всички четворки резултати за всяко изречени от един текст. Постигане на 80% разпознаване на тестово множество.

* + 1. **Ивличане на текстове на песни:**

Извличане на текстовете чрез “scrapy” от сайта: “http://www.azlyricdb.com”. Сравнително лесно има линкове за всички букви от английската азбука. Те водят до страници с линкове за всички изпълнители от дадената буква. Те водят до линкове с всички песни на даден изпълнител. И всяка песен е линк към текста й.

* + 1. **Краен вид на данните:**

Конкретна част от извлечените текстове се класифицират с модела, който е получен и след това се индексират с “Woosh” по “n”-грами, за ефективно търсене.

* + 1. **Потребителски интерфейс:**

Създаване на уеб приложение с проста функционалност за търсене, която изпраща заявки до сървър, който се обръща до индексираните текстове и извлича резултата и ги сортира по избрана емоция спрямо резултатът и от модела.

# **Заключение**

Резултатите след тренирането на модела са сравнително добри. Това е вследствие на добрата ефективност и синтезирането на най-важната информация от езика за моделиране на текст “Bert”.

Като бъдеща работа може да се направи проучване как да се подобри модела за по - добра генерализация и като се извлекат повече текстове на данни за по богат избор на резултати.

# **Използвана литература**

<https://www.researchgate.net/publication/317031495_MoodyLyrics_A_Sentiment_Annotated_Lyrics_Dataset>

<https://keras.io/>

<https://arxiv.org/pdf/1810.04805.pdf> - език за моделиране Bert

<https://www.allthelyrics.com/index>

<https://scrapy.org/>

<https://whoosh.readthedocs.io/en/latest/intro.html>

<http://flask.pocoo.org/>

<https://angular.io/>