Министерство образования Республики Беларусь

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №3 по курсу «ЕЯзИИС» на тему:

«Разработка системы автоматического реферирования документов»

Выполнили студенты группы 921703: Кравцов Михаил Сергеевич
Новицкий Сергей Владимирович
Семякин Артем Вадимович
Губаревич Иван Андреевич

Проверил: Крапивин Юрий Борисович

Минск 2022

Содержание

1. Цель работы и вариант

Цель работы - освоить на практике основные принципы автоматического реферирования документов.

Вариант 4:

4	Русский, Немецкий	Sentence extraction+ ML	Научные статьи по computer science, Сочинения по литературе

2. Информация о текстовой коллекции документов

Были использованы тексты из газет и классических произведений на соответствующем языке. В проекте можно найти следующие файлы:

- computerScience.txt — содержит тренировочный текст на немецком языке;
- essayLiterature.txt — содержит тренировочный текст на русском языке;

3. Описание системы, данных и алгоритмов

3.1. Описание структуры системы

Система представляет собой веб-приложение, написанное с использованием Flask, присутствуют классы, ответственные за интерфейс и логику за ним. Разбор текста происходит с применением метода Sentence extraction и библиотеки nltk и встроенным ML. Система реализована на языке python.

3.2. Описание типов данных

В данной системе мы использовали такие типы данных как строка, массив, ассоциативный массив.

3.3. Описание алгоритмов

1. Вычислить веса слов документа.

При этом слова из латинских букв, числа, стоп-слова - не учитываются. Базовый вес слова вычисляется по формуле TF*IDF.

2. Вычислить веса предложений согласно формулам, приведенным ниже.
3. Осуществить генерацию реферата.

Этап генерации представляет собой выбор из исходного текста определенного количества предложений с наибольшим весом в той последовательности, в которой они идут в тексте. Рекомендуемый размер реферата 10 предложений.

Вес каждого предложения S_i вычисляется произведением значений функций приведенных ниже.

Функции, характеризующие положение предложения в документе $\mathrm{Posd}(S_i)$ и положение в абзаце $\mathrm{Posp}(S_i)$:

$$\mathrm{Posd}(S_i) = 1 - \frac{BD(S_i)}{|D|}$$

$$\mathrm{Posp}(S_i) = 1 - \frac{BP(S_i)}{|P|}.$$

где

|D|- число символов в документе D, содержащем предложение S_i;
BD(S_i) – количество символов до S_i в D(S_i);
|P| - количество символов в абзаце P, содержащем предложение S_i;
BP(S_i) – количество символов до S_i в абзаце.

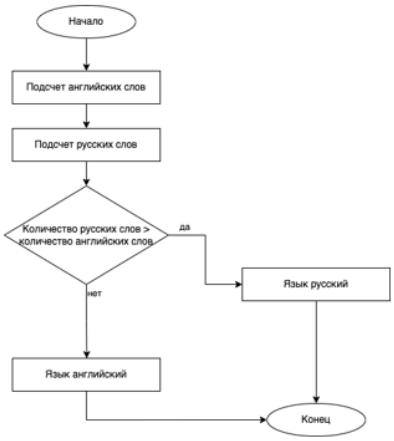
Модифицированная TFIDF функция:

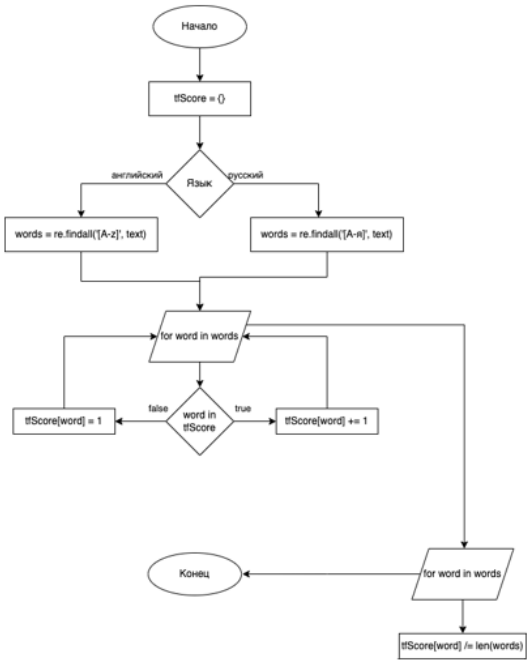
$$\mathrm{Score}(S_i) = \sum_{t \in S_i} tf(t, S_i) \cdot w(t, D).$$

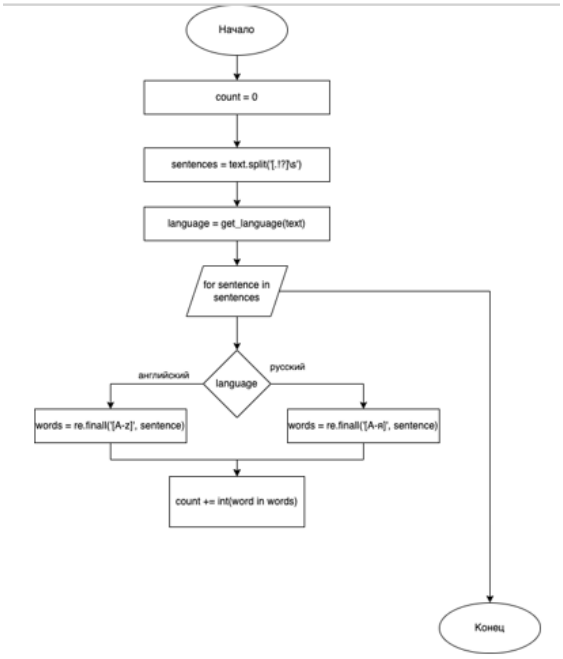
$tf(t, S_i)$- частота термина t в предложении S_i;

$$w(t, D) = 0.5 \left(1 + \frac{tf(t, D)}{tf_{max}(D)}\right) \cdot \log\left(\frac{|DB|}{df(t)}\right).$$

$tf(t, D)$- частота термина t в документе D;
$df(t)$- количество документов, с термином t;
$tf_{max}(D)$ - максимальная частота термина в документе D;
|DB| - количество документов.

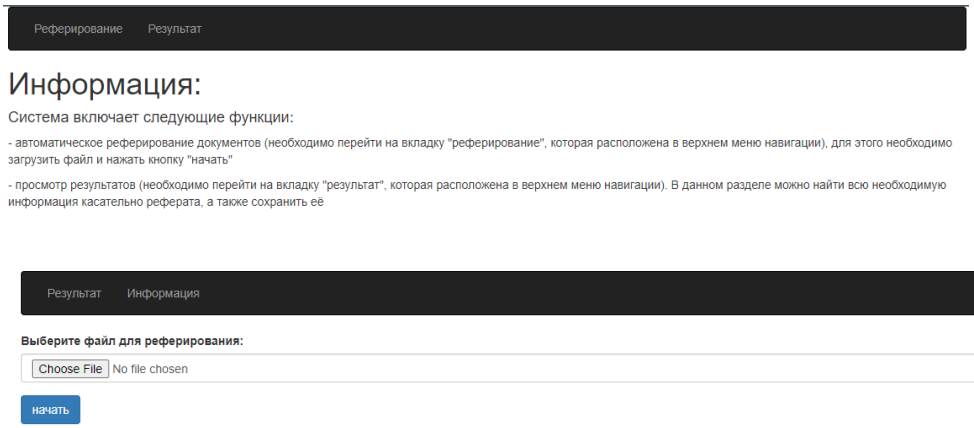

Начало
Подсчет английских слов
Подсчет русских слов
Количество русских слов > количество английских слов
да
нет
Язык русский
Язык английский
Конец

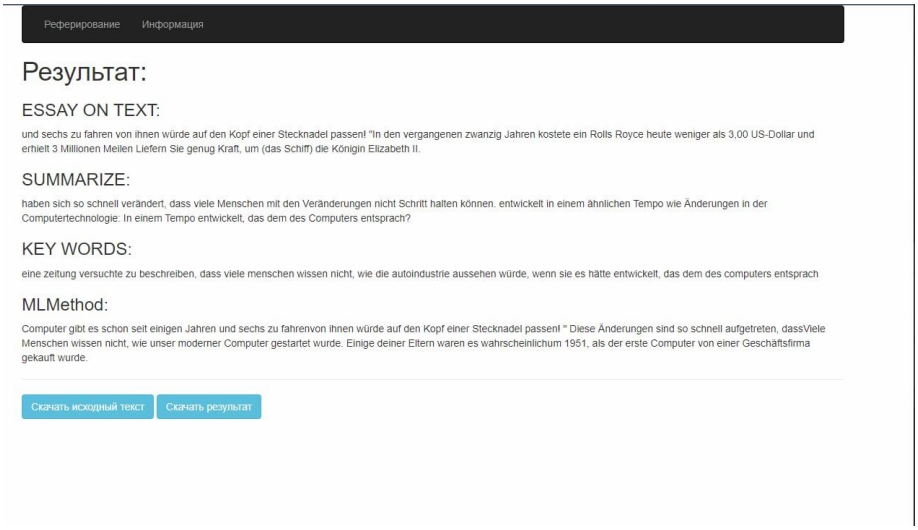

Начало
tfScore = {}
Язык
английский
русский
words = re.findall('[A-z]', text)
words = re.findall('[А-я]', text)
for word in words
word in tfScore
false
true
tfScore[word] = 1
tfScore[word] += 1
Конец
for word in words
tfScore[word] /= len(words)

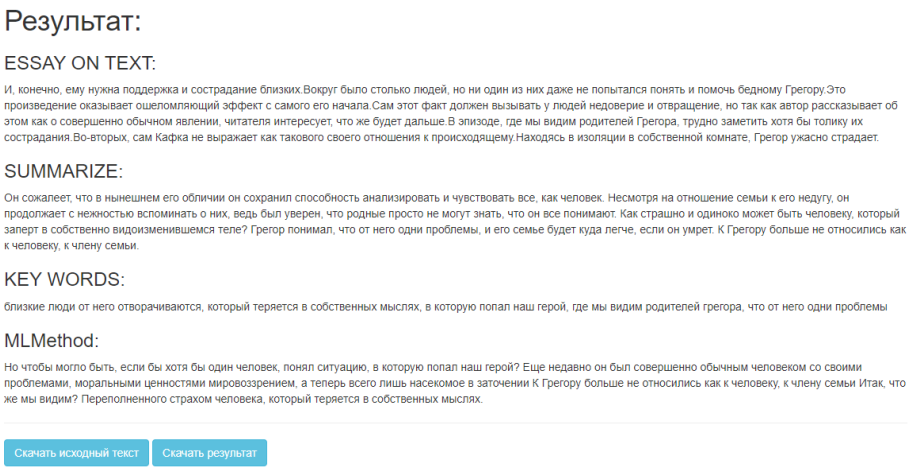

Начало
count = 0
sentences = text.split('[.!?]\s')
language = get_language(text)
for sentence in sentences
language
английский
русский
words = re.finall('[A-z]', sentence)
words = re.finall('[А-я]', sentence)
count += int(word in words)
Конец

3.4. Результат тестирования системы

Пример взаимодействия с программой выглядит следующим образом







4. Использование библиотек

Для реализации системы использовалось библиотека nltk со встроенным Machine Learning, которая позволяет работать с естественным языков (в данном случае с русским и немецким).

Для выполнения лабораторной работы использовались следующие библиотеки:

1. nltk
2. nltk.corpus
3. math
4. gensim.summarization
5. rake_nltk
6. string
7. nltk.corpus
8. nltk.cluster.util
9. numpy
10. networkx
11. sys

5. Вывод

В данной лабораторной работе была реализована система для автоматического реферирования текстов в виде веб приложения с удобным пользовательским интерфейсом, поддерживаются два языка: немецкий и русский.