Ati: Система за разпознаване на авторство

Симеон Христов

6MI3400191

SOFIA UNIVERSITY
St. KLIMENT OHRIDSKI

Курсов проект по
Извличане на информация

Факултет по математика и информатика
Софийски университет

Лектор: проф. Иван Койчев

Февруари 2023

Съдържание

1 Увод

Целта на проекта е създаването на система за категоризация, която при даден корпус от документи - D, всеки от които е написан от един автор y, идентифицира автора на анонимен текст x.

Разработената система може да бъде основата за разработване на приложение, което да:

1. **Проверява на авторство**: Дали даден текст наистина е написан от определен автор?
2. **Открива плагиатство**: Намиране на прилики между два или повече текста;
3. **Създава профил или характеризира на даден автора**: Извличане на информация за възрастта, образованието, пола и т.н. на автора на даден текст;
4. **Открива стилистични несъответствия** (както може да се случи при съвместно писане): Дали авторът настина е само един?
5. **Отговоря на въпроси**: В матурата по български език и литература има въпроси, които са фокусирани върху разпознаване на автора на даден отказ или разпознаване на автора, който пише за определен герой.

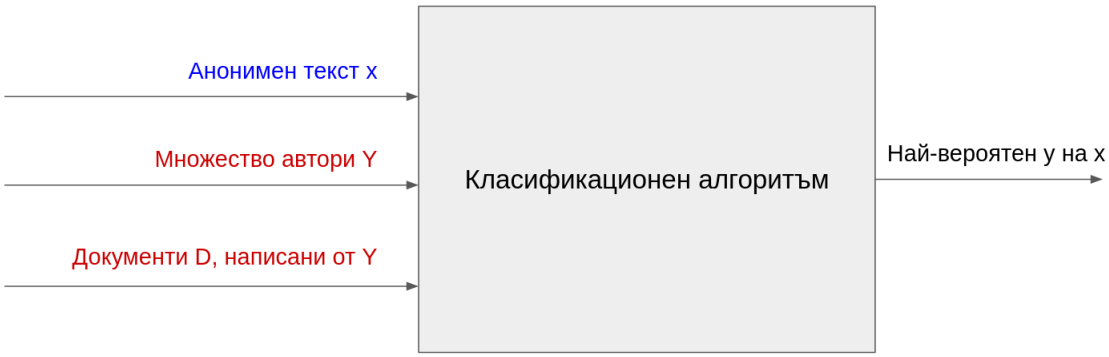
2 Преглед на областта за разпознаване на авторство

В [1] Holmes дискутира няколко начина, по които може да се създаде средство за разграничаване на автори. Най-простият подход е чрез броене на уникални думи. Това може да се използва за пресмятане на отношението между броят уникални думи в текст спрямо неговата дължина. Друг фокус може да бъдат думите, които се срещат много рядко - например, такива могат да бъдат думи от технически характер или жаргонни форми. Също толкова важно е обаче и изследването на думи, които в други задачи за извличане на информация биха били игнорирани. Това

са така наречените "стоп-думи те са част от почти всяко изречение и често играят роля на "набухвател"в текста. При задача за класификация на настроението и/или тоналността тези думи не биха представлявали характеристики, притежаващи силна дискриминативна сила. Примери са различни предлози ("на "от "с"), съюзи ("и "а "но"), числителни имена ("един "два") и местоимения ("той "тя"). В задачата за определяне на автора на даден текст обаче средното количество на тези думи в текст, писан от един автор, може да се отличава (при това силно) от средното количество на същото множество от думи в текст, писан от друг автор. Разглеждането на подобни лексикографски характеристики би произвело за даден текст матрица от неотрицателни числа. Върху нея Holmes предлага да се използва анализ на главните компоненти (principal component analysis - PCA) и чрез използване на статистически техники като факторен анализ, дискриминантен анализ и клъстерен анализ да се постигне финална класификация.

В [2] Stamatatos предлага друг подход. Вместо да се анализира текстът на ниво лексика той предлага да се анализира на ниво граматика с фокус върху морфологията. Той извършва експерименти върху 10 автори от съвременно гръцко списание, в които брои колко пъти се срещат различни морфологични форми - съществителни фрази, глаголни фрази, предложни фрази и т.н. Финалният модел е линейна регресия върху всеки автор, която постига оценка $R^2 = 0.63$, което е много добър резултат и показва, че разликата в стила на множество автори може да се представи като линейна функция, която дава най-голяма стойност за най-вероятния автор на текста. Това е нетипично използване на модела - той бива приложен върху 22 количествени характеристики, извлечени от текстовете на всеки автор и финалната класификация се определя от най-голямата стойност, която се връща. Също така се констатира, че не са необходими твърде много и твърде текстове за решаването на тяхната задача - те използват 10 текста за даден автор, всеки от които е с дължина около 1000 думи.

В [3] Grieve сравнява 39 различни стилови маркера и ги подрежда въз основа на тяхната предсказваща сила. Фокусът отново е насочен не към използване на текста за характеристика, а към извличане и формиране на характеристики от него. Той отбелязва, че съставянето на корпуса (т.е. множеството от текстове) е от изключително значение. Със сигур-



Фигура 1: Системата, представена "от птичи поглед".

ност всеки един от текстовете трябва да е подбран така, че да може да покаже стила на автора, но уточнението на Grieve е, че когато всички текстове се комбинират, те също трябва да продължат да представят различни стилове. Изследваните маркери са подробно описани и обяснени. Те могат да се използват и в други задачи за обработка на естествен език. Заключението, което Grieve прави е, че характеристиката, наречена "Профил на думи и препинателни знаци"("Word and punctuation mark profile") има най-голяма дискриминативна сила.

В [4] Stamatatos представя плюсовете и минусите на използването на много различни стилови маркери и подходи за обработка на естествен език. За разлика от предните статии тук се прави анализ и на различни по-сложни класификатори, основани на методи на машинното самообучение. Представеният финален модел е машина на опорните вектори (support vector machine - SVM). Най-голямото предимство, което е и широко дискутирано в научната литература, е, че тя е в състояние да избегне проблема с пренагаждането към тренировични данни дори когато се те са с размерност няколко хиляди няколко хиляди.

3 Проектиране

От гледна точка на машинното самообучение общата архитектура на системата е представена схематично на Фигура 1.

Обособени са пет модула, които формират процесите от свалянето на

текстовете от Интернет пространството до изписване на информация за класификацията на даден текст в потребителския интерфейс. Те са:

1. Модул scraper.py. Съдържа логиката за извличане на информация за текстовете от Интернет пространството и тяхното трансформиране в подходяща структура от данни, чрез която те да могат да бъдат обработени по лесен начин. Резултатът от използването на този модул е двумерна таблица, съставена от три колони: име на автор, име на произведение, път (т.е. URL) към Интернет ресурса, от които той може да бъде свален на локална машина.

2. Модул downloader.py. Съдържа логиката за свалянето на текстовите ресурси от Интернет. Резултатът от използването на този модул е множество от текстови файлове, разположени в локални директория.

3. Модул preprocessing.py. Съдържа логиката за обработка на текста преди подаването му към модел (т.е. класификатор). Резултатът от използването на този модул е списък от символни последователности (т.нар. "токени"). Реализацията предлага три нива на обработка, всяко от което добавя повече операции към (с други думи, надгражда) предишното. Те са уточнени в секция 4.1.

4. Набор от Jupyter тетрадки. Съдържат логиката, която свързва гореизброените модули. Те предоставят възможност за изграждане на линейна и йерархична структура на кода. Използвани са за провеждането на експерименти и визуално проследяване на различните промени, които се правят върху двумерната таблица, използвана за създаване на тренировъчно, валидацинно, и тестово множество за класификационния модел.

5. Модул app.py. Съдържа логиката за изграждането и обновяването на потребителския интерфейс, с който крайният потребител работи. Крайният резултат е интерактивен канал за комуникация с потребителя.

Изискванията за разработване на системата са:

1. Създаване на паяк, копаещ документи (текстовете на съответните автори) от уеб страница.

2. Създаване на модел, който е трениран върху тренировъчно множество (80% от данните), валидиран върху валидационно множество (10% от оставащи данни) и оценен върху тестово множество (последните 10% оставащи данни).

3. Сравняване на поне 3 стилистични метрики за всички автори.

4. Сравняване на различни представяния на текст: *tf-tdf* и *transformer sentence embeddings*.

5. Сравняване на класификатори: един и ансамбъл.

6. Създаване на потребителски интерфейс.

Реализиран е и потребителски интерфейс, който е представен на .

4 Реализация, тестване/експерименти

4.1 Използвани технологии, платформи и библиотеки

[ ] TODO: Подходящи средства за реализация за проекта (технологии, платформи и библиотеки). Избор на средствата и начин за използването им;

4.2 Реализация/Провеждане на експерименти

[ ] TODO: Реализация (на модулите); За система/приложение: На кратко: планиране на тестването - тестови сценарии,...; Анализ на резултатите.

5 Заключение

[ ] TODO: Обобщение на направеното/резултатите. Идеи за по-нататъшно развитие, усъвършенстване или други експерименти.

6 Използвани технологии

- Python: програмен език, използван за създаването на Ati;
- Numpy: външна библиотека за бързи матрични и числови операции;
- Pandas: външна библиотека за манипулация на едномерни и двумерни данни;
- pickle: външна библиотека за запазване на структури във файловата система чрез създаване на бинарен .pkl файл;
- PyCaret: външна библиотека за автоматично създаване, трениране, валидиране и фино настройване на модели, основани на изкуствен интелект;
- Classla: външна библиотека за автоматично маркиране на частите на речта;
- selectolax: външна библиотека за извличане на информацията, съхранявана в HTML и CSS елементи, изграждащи и формиращи изгледа на уеб сайтове;
- bulstem: външна библиотека за връщане на думи към тяхната коренна форма;
- lemmagen3: външна библиотека за трансформиране на думи към техните т.нар "леми общи форми за голямо множество от думи;
- nltk: външна библиотека за обработка на естествен език;
- scikit-learn: външна библиотека за (ръчно) създаване, трениране, валидиране и фино настройване на модели, основани на изкуствен интелект;
- streamlit: външна библиотека за изграждане на уеб интерфейс.

7 Използвана литература

[1] Holmes, David I. “Authorship Attribution - Language Resources and Evaluation.” SpringerLink, Kluwer Academic Publishers, https://link.springer.com/article/10.1007/
[2] Stamatatos, Efstathios. “Automatic Authorship Attribution.” ACM Digital Library, Association for Computational Linguistics, http://portal.acm.org/citation.cfm?doid=97703
[3] Grieve, Jack. “Quantitative Authorship Attribution: An Evaluation of Techniques.” Digital Object Identifier System, Literary and Linguistic Computing, https://doi.org/10.1093/llc/fqm020.
[4] Stamatatos, Efstathios. “A Survey of Modern Authorship Attribution Methods.” Wiley Online Library, Journal of the American Society for Information Science and Technology, https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.21001.

Списък на фигурите