# **УВОД:**

В днешно време информацията в интернет е леснодостъпна, но не винаги достоверна. Много медии използват сензационни заглавия и подвеждаща информация, за да привлекат повече внимание, което води до по-ниска информираност на обществото. В резултат на това се формира обществено недоверие към медиите и към представените от тях новини. Заради това има необходимост от автоматизирана система, която да анализира и оценява сензационността на публикуваните статии. Проектът има за цел да се направи приложение, което да анализира и оценява сензационността и емоционалната натовареност на новините, публикувани в българските онлайн медии. Това се постига чрез сравняване на публикуваните новини с проверен и надежден източник - Българската телеграфна агенция. Важно е да има приложение, което да може автоматично да класифицира новините като сензационни и/или с емоционална натовареност. Към настоящият момент няма такова приложение, както и AI модел, които да показват сензационността и емоционалната натовареност на публикуваните новини в България.

# **ЦЕЛИ:**

Основните и специфични цели на проекта са:

• Създаване на подход и алгоритъм за оценяване на сензационността и емоционалната натовареност на публикуваните новини от медийните сайтове.

• Разработване на уеб приложение и fine-tuning на AI модел, които прилагат подхода и алгоритъма.

• Валидиране на подхода в рамките на приложението и AI модел с примерни данни.

• Създаване на техническа документация и презентация за разпространение на резултатите.

# **ОСНОВНИ ЕТАПИ В РЕАЛИЗИРАНЕТО НА ПРОЕКТА:**

1. Разработване на план на проекта.

2. Избор на медия с проверен източник на информация, където представянето ѝ не е сензационно- БТА.

3. Избор на други 5 медии за сравнение – Медиапул, Пик, БНР, Дунавмост, 24часа.

4. Създаване на шест автоматизирани алгоритъма, които извличат новините от предходния ден от шестте медии и ги запазват в подходящ за обработка и сравнение формат.

5. Създаване на списък със сензационни думи, чрез разглеждане на медии, които използват сензационен език.

6. Сравняване на подходящи методи за подобност на текст, чрез подобие в заглавието и съдържанието. Чрез избрания подход могат да се сравнят новините от петте новинарски източника с БТА и запазването на резултатите в подходящ за обработка формат.

7. Избиране на метода за сравнение, подходящ за целите на проекта и имплементирането му в програмен код.

8. Разработване на алгоритъм за автоматично качване на новините и качване на резултата от алгоритъма за подобност в базата данни.

9. Разработване на алгоритъм за разбиране на сензационен език в сървърната част, за да могат ефективно да се създават статистики.

10. Създаване на уеб сайт за визуализация на новините от БТА и сходните на тях от изследваните новинарски сайтове.

11. Създаване на метод за подчертаване на сензационния език в новините на изследваните медии.

12. Изчисляване на процент на сензационност на всяка новина и медия.

13. Представяне на стиловите конструкции / сензационни думи/, които най-често са използвани в обследваните статии.

14. Избиране на подходящ AI модел за трениране, който отговаря на изискванията и на конкретната задача

15. Извличане на данни за fine-tuning на модела и подготвянето им в подходящ формат

16. Трениране на модела

17. Извършване на нужните корекции, за да бъде модела най-точен

18. Качване на модела в HuggingFace - <https://huggingface.co/NikolaML/NewsAnalizer/tree/main>

19. Прилагане на модела в уеб приложението

20. Създаване и прилагане на алгоритъм, който открива най-използваните сензационни думи от всяка от изследвани медии.

21. Разработване на техническа документация и презентация.

# 

# **ПРОЦЕСИ И ПРАВИЛА ПРИ ОЦЕНЯВАНЕ НА НОВИНИ И ИНТЕРНЕТ МЕДИИ:**

• Изграждане на система за автоматично извличане на новини от различни новинарски сайтове.

• Създаване на алгоритъм за сравняване на новините с тези на Българската телеграфна агенция, чрез подобие в заглавието и съдържанието.

• Оценка на всяка новина по критерии за сензационност на използвания публицистичен стил посредством употребата на списък със скандални думи и изрази, които да покажат дали новината е представена сензационно или не.

• Изчисляване на рейтинг за обективен публицистичен стил както на новините, така и на медиите, които са ги публикували.

• Fine-tuning на XML-RoBERTa и внедряването му в сайта. Този AI показва каква емоция се предава на потребителя при изчитане на новината. Той връща процент на неутралност, позитивност или негативност при представяне на новината.

# **ОПИСАНИЕ НА ПОДХОДА И АЛГОРИТЪМА:**

**Извличане на новини:** Процесът на извличане на новини използва автоматизиран механизъм, който ежедневно събира съдържание от избрани медийни сайтове. За всеки сайт е разработен специализиран скрейпър, адаптиран към неговата уникална структура. Скрейпърите се стартират на всеки 24 часа. Те отварят зададената медия в браузър, след което обработват всяка новина поотделно, като извличат заглавието и съдържанието ѝ, игнорирайки останалите елементи на страницата. Само за БТА скрейпърът извлича и снимки. Тази технология предлага значително по-добър и ефективен метод за достъп до актуална информация от множество източници, елиминирайки нуждата от ръчна намеса.

**Сравняване на новини:** Новините публикувани от Българската телеграфна агенция са написани на обективен публицистичен стил. Те служат за еталон при сравнение на сходна новина публикувана в изследваните медийни сайтове. Това сравнение се извършва чрез алгоритъм за подобност между заглавието и съдържанието на новините в шестте медии. Създадения алгоритъм идентифицира коя новина от дадена медия съответства на новина от БТА, като изчислява в проценти сходството между тяхното съдържание. Използвал съм TF-IDF -извлича значимостта на думите в текст, като отчита тяхната честота в текста и в целия набор от текстове и ги обръща във вектори, а косинусовата подобност измерва ъгловото сходство между векторите на сравняваните текстове, за да определи тяхната степен на сходство.

**Сравнение на алгоритми за подобност:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Метод | Описание | Предимства | Недостатъци | Защо e избран |
| **Жакартова подобност** | **Измерва сходството чрез съотношението на общите и уникални елементи в двата текста.** | **Лесен за имплементиране, бърз за сравнение.** | **Не отчита контекста на думите, неподходящ за сложни текстове.** | **Не е подходящ за сложни текстови анализи, поради липса на контекст.** |
| **Разстояние на Левенщайн** | **Измерва минималния брой редакции (вмъкване, изтриване, замяна) за преобразуване на текстове.** | **Подходящ за сравнение на текстове с малки разлики.** | **Не е ефективен за дълги текстове и не отчита семантиката на думите.** | **Не се справя добре с дълги текстове и сложни структури.** |
| **TF-IDF с Косинува подобност** | **Комбинира TF-IDF за извличане на важни термини и косинува подобност за измерване на сходство.** | **Ефективен при сравнение на текстове с различна дължина, акцентира върху значимите термини.** | **Не отчита контекста на думите, не е подходящ за семантични анализи.** | **Най-подходящ за дълги текстове и най – ефективно и бързо се справя със задачата** |

**Установяване на ниво на сензационност на използвания публицистичен стил:** Алгоритмът следи за наличие на скандални думи и изрази, които са предварително зададени и са често използвани в сензационни статии, като „шок“, „уау“, „екшън“ и други. За всяка новина се изчислява рейтинг на сензационност въз основа на този критерий. Този отделен рейтинг на всяка статия се натрупва към общия рейтинг на медията и така може да се оцени не само конкретната статия, но и конкретната медия, както и да се проследи движението и от използваща обективен стил на изразяване към сензационен и обратно.

**Установяване на най-често използваните сензационни думи от всяка от изследваните медии:** Методът минава през всяка новина и записва за всяка медия вида и броя на използваните сензационни думи и изрази. Те биват показани на потребителя в страницата „Новини“.

**Трениране на AI модел, за показване на начина на представяне на новините:** Използва се моделът XML-RoBERTa, който го fine-tune-ах да показва с по-голяма прецизност каква емоция се предава на потребителя при изчитане на новината. Извлечена бе информация от две медии – БТА (емоционално неутрални новини) и Пик (емоционално натоварени новини). Данните бяха свалени от скрейпърите в продължение на седмица. Те бяха разделени на две части. Първата – новини с различен контекст, а втората – със сходен контекст и с различно емоционално натоварване. Така моделът се обучи по-добре да преценява начина на представяне новината. Създаден бе и метод, който е четим за потребителя изход от модела.

# **ЛОГИЧЕСКО И ФУНКЦИОНАЛНО ОПИСАНИЕ НА ПРИЛОЖЕНИЕТО:**

**Скрейпване на новини:** Това е методът за събиране на данни. Той се състои от множество scraper-и, всеки от които е специфичен за даден медиен сайт. Scraper-ите се активират ежедневно чрез cron job и извличат новини, публикувани през изминалия ден в наблюдаваните сайтове, както и в сайта на БТА.

**Сравнение на новини:** Сравнява се съдържанието на новините публикувани в наблюдаваните сайтове с тези на БТА. Има текстов анализатор, който изчислява процента на съвпадение между думите на статиите. Ако съвпадението надхвърля определен процент, новините се считат за идентични.

**Достоверност:** В текста на проверяваната статия се търсят думи и изрази, които обикновено се използват в сензационните медии. Показва се вида и броя на използваните сензационни думи и изрази на всяка от изследваните медии.

**Първична обработка на входните данни за трениране на модела: Процесът за събиране на данни започна с извличането им в период на седмица от две медии – БТА(с неутрален емоционален характер), както и Пик(с негативен, позитивен и неутрален характер) през създадените скрейпъри.** Тези данни се подават на друг модел, за да се оцени тяхната емоционална натовареност, като така се определя дали те са неутрални, негативни или позитивни, както и се дава допълнителна оценка за интензитета им. Информацията **се разделя на две части. Първата минава през метода за сходство, като така се намират сходните новини от двете медии, което ще покаже на модела при трениране разликата между неутрална и друг вид емоционално натоварена новина. В другата част са новините, за които не е намерено подобие.** Данните се съхраняват в JSON файл и се зареждат в програмата. След това те се разделят на още две части: тренировъчени и тестови, като 80% от данните се използват за обучение, а 20% – за оценка на модела.

**Вторична** **обработка на текстовите данни:** За обработката на текстовите данни се използва tokenizer, който преобразува текстовете в подходящ формат за подаване в модела. Всяко текстово изречение се разделя на токени, прилага се padding и се задава максимална дължина, за да може моделът да работи ефективно. Емоционалата натовареност се преобразува в числови стойности (0 за негативен, 1 за неутрален и 2 за позитивен).

**Използван модел:** За класификацията се използва предварително обучения модел XLM-RoBERTa, който е специализиран за анализ на чувства и емоции в текст. Този модел е fine-tuned-ат със зададените тренировъчни данни с цел адаптация към конкретната задача.

**Конфигуриране на тренировъчния процес:** Моделът се обучава чрез библиотеката Transformers, като се задават специфични параметри за обучение. Основните параметри включват:

• Стратегия за оценка на всеки epoch.

• Запазване на най-добрата версия на модела въз основа на загубата при оценка.

• Максимален брой epoch за обучение - 3.

**Процес на обучение:** След конфигурирането, моделът преминава през тренировъчен процес, където научава как да разпознава емоционалната натовареност на новините въз основа на предоставените данни. Оценката на качеството на обучението се извършва чрез критерии за точност, което помага за измерване на ефективността на модела.

**Запазване на обучен модел:** След приключване на тренировката, обученият модел и tokenizer-ът се запазват във файлове, за да могат да бъдат използвани в последствие, като моделът е качен в HuggingFace, от където се създаде url, който осигурява достъп до него. Интегрирах този модел в уеб приложението, като създадох endpoint, който изпълнява Python програма, приемаща данни от фронтенда. Тези данни преминават през модела и резултатите се визуализират на потребителския интерфейс. Моделът е публично достъпен и всеки, който иска може да го изполва.

**Резултати при употреба на обучения AI:** При подаване на текст – включително и на различни езици от българския, се извеждат два показателя – първият – вида на емоционалната натовареност на новината: неутрална, позитивна или негативна, а вторият - процент на сигурност за този вид – 1 % до 100%, това показва колко моделът е сигурен във взетото си решение. Те се връщат във формат JSON, за лесна последваща обработка.

**Уеб интерфейс:** Представят се нагледно новините от БТА и сходните на тях от изследваните медии. Потребителят може да отвори всяка новина от БТА и да види нагледно разликата в стила на представянето и в изследваните медии. Има възможност за подчертаване на сензационните думи и изрази използвани в представянето на новините. Новините могат и да се филтрират според датата на качването им. На страницата се вижда статистка за това в колко на брой статии на дадена изследвана медия са използвани сензационни думи и изрази, като се показва и процента на сензационност на медията. Представяне на стиловите конструкции / сензационни думи/, които най-често са използвани в обследваните статии от различните медии. На всяка статия се показва и нейната емоционална натовареност, като неутрална, позитивна или негативна, както и процент на сигурност на емоционалната оценка.

# **ТЕХНИЧЕСКО ОПИСАНИЕ НА ПРИЛОЖЕНИЕТО:**

Приложението, което реализира scraper-ите е направено чрез библиотеката puppeteer. Има алгоритъм, който всеки ден се активира чрез cron job и извиква scraper-ите, които вземат информация и я съхраняват в отделни текстови файлове. За всяка една новина от БТА се проверява и намира сходната на нея от наблюдаваните медии. Стартира се анализатор, който чрез набор от скандални думи за конкретната новина, установява до колко тя е сензационна. Добавя се изчисления рейтинг към общия на медията и се визуализира на потребителя. Направен е и уеб интерфейс за по-добър и достъпен UX. Това е създадено чрез програмните езици ReactJS, JavaScrip, ЕxpressJs, MongoDB, Python и Scikit Learn, PyTorch, Puppeteer.

Използвани са програмни езици, които са подходящи за приложените алгоритми и функционалности. Подбраните технологии са използвани ефективно по предназначение и са съобразени с изискванията на проекта.

Използването на ReactJS с Javascript позволява динамично генериране на уеб страници. Приложението е качено в GitHub хранилището, като то може много лесно да бъде инсталирана локално и използвана през стандартен уеб-браузър.

Дизайнът е авторски. Предвид целта на проекта, при разработката стремежът беше за ненатрапчивост, леснота на употреба и бързина на действията на потребителя. Основната навигация се извършва през стандартно мобилно меню, а сайтът е създаден и тестван за използване върху мобилни устройства. Гамата на цветовете е съобразена с естеството на приложението и са използвани публично достъпни изображения.

Всички променливи са наименовани по стандартите на camelCase – най-използваната конвенция при JavaScript, както и snake case, който е най-използваната конвенция в Python . Програмният код е написан в различни файлове, за да може да се използва многократно и да бъдат сведени до минимум повторенията, като използва DRY(don’t repeat yourself) принципа. Всяка една преизползваема функция има заложен коментар, който обяснява какво представлява тя. Създаденият програмен код е валидиран с примерни данни и е дебъгнат, за да се премахнат всички открити неточности. Данните на потребителя не се запазват никъде дългосрочно, като така те са защитени и никой няма достъп до тях.

Всяко едно парче код е в отделен файл да се избегнат повторения и да може лесно да се преизползва, като така се избягват ненужни забавяния. При неочаквана грешка, програмата изписва в конзолата на браузъра кратко описание на грешката, за да може потребителят лесно да докладва на автора и тя да бъде отстранена. Всички функционалности са тествани за бързодействие чрез стандартния инструментариум на Chrome Dev Tools. Цялата система е разположена на публичен хостинг, което позволява възползване от предоставяните услуги свързани с производителност, защита на мрежата, архивиране, като така е повишено нейното ниво на сигурност и стабилност.

# **ВАЛИДИРАНЕ:**

Приложението определя чрез процентно съвпадение на думите дали две новини са идентични. За оценка на точността бе извършено тестване на новини от различни медии с публикуваните в БТА, за да се провери дали алгоритмът правилно ги разпознава като идентични по съдържание. Дали дадена новина е сензационна се преценява въз основа на списък със скандални думи, който ще бъде актуализиран периодично въз основа на наблюдение на публикации в интернет медиите. Приложението беше валидирано с примерни данни, събрани от шестте скрейпъра, които извлякоха новини от шестте медии на 07.01.2025 и на 21.03.2025. След прилагане на алгоритъма за подобие, се установи, че новините са сходни. След това лично проверих дали новините в резултатите от алгоритъма са идентични, като така потвърдих, че методът за сравнение функционира правилно.

По време на обучението на AI моделът бяха заделени 20% от тренировъчните данни за тестване и редакция на конфигурациите на модела, за да може да бъде обучен ефективно. След окончателното трениране, вече обученият модел беше валидиран с нова информация, като това се осъществи чрез метод, чрез който му се подават новините във формат на JSON, те автоматично се подават на модела и се записва резултатът в текстов файл, който е лесно разбираем за четене. Аз ръчно проверих дали резултатите са коректни, като установих така, че fine-tune-ът е успешен.

# **ЗАКЛЮЧЕНИЕ:**

Приложението представя възможност за оценка на използвания публицистичен език на българските онлайн медии. То може да се използва от всички хора, които искат да се информират, както и от журналисти, анализатори и показва каква е медийната среда в България. Програмата има набелязани възможности за развитие, като: добавяне на нови медии, които да се анализират, добавяне на AI за по-точна оценка на подобността.

Към настоящия момент се разработва втори метод за сравняване на подобност на новините чрез изкуствен интелект, който се тренира специфично за тази задача.

**GitHub:** <https://github.com/Nikola-Velikov/Pure-Media>

**Обученият AI модел:** [https://huggingface.co/NikolaML/NewsAnalizer/tree/main](https://huggingface.co/NikolaML/NewsAnalizer/tree/main%20)

**Уеб приложение:** <https://pure-media-front-end-czr7.vercel.app/>