Софийски университет “Св. Климент Охридски”
Факултет по математика и информатика
Катедра “Софтуерни технологии”

Дисертационен труд

на тема

Автоматично резюмиране на правни текстове

Докторант: Валентин Венков Змийчаров
професионално направление 4.6 “Информатика и компютърни науки”;
Докторантска програма “Софтуерни Технологии” – Откриване на знания

Научен ръководител:
проф. д-р Иван Койчев
катедра “Софтуерни технологии”, ФМИ, СУ “Св. Климент Охридски”

София, Май 2025 г.

Съдържание

Благодарности **6**

1 Въведение **7**

1.1 Мотивация и актуалност на темата 7

1.2 Цел и задачи на дисертационния труд 8

1.3 Структура на дисертацията 9

2 Обзор на предметната област **11**

2.1 Специализирани термини и понятия 11

2.2 Обработка на естествен език 14

2.3 Методи за обработка на естествен език чрез дълбоки невронни мрежи 16

2.3.1 Текстови вектори (вграждания на думи) 17

2.3.2 Рекурентни невронни мрежи 18

2.4 Архитектура на Трансформатор 22

2.4.1 Механизъм на самовнимание 22

2.4.2 Механизъм на внимание с много глави 25

2.4.3 Позиционно кодиране 26

2.4.4 Архитектура на Трансформатор 27

2.5 Предварително обучени езикови модели 30

2.5.1 BERT 30
2.5.2 Езиков модел T5 33
2.5.3 Големи езикови модели, GPT 35
2.6 Автоматично резюмиране на текстове 37
2.6.1 Видове резюмиране 38
2.6.2 Подходи за резюмиране 40
2.6.3 Набори от данни за резюмиране 40
2.6.4 Оценяване при автоматичното резюмиране 43
2.7 Свързани разработки и съществуващи решения 45
2.7.1 Автоматизирано тълкуване на правни документи чрез големи езикови модели 45
2.7.2 Стандартизиране и оценка на модели за обобщаване на правни текстове 46
2.7.3 Голям набор от съдебни решения на Върховния съд на САЩ за автоматично резюмиране 47
2.7.4 Многоезично резюмиране на съдебни решения от швейцарската съдебна система 48

3 Нов корпус от данни за правни текстове и техните резюмета 49

3.1 Въведение 49
3.2 Събиране и изчистване на данните 50
3.3 Данни на български език 52
3.4 Данни на други езици 57

3.5 Публикуване на набора от данни 58

4 Разработени и адаптирани подходи за резюмиране на текстове 61

4.1 Базов подход (избиране на първите изречения в текста) 61

4.2 Екстрактивен подход основан на TF-IDF 62

4.3 Екстрактивен подход TextRank 63

4.4 Екстрактивен подход основан на BERT и K-means 65

4.5 Екстрактивен подход основан на LegalBERT и K-means 67

4.6 Модел за резюмиране PreSumm 68

4.7 Езиков модел T5 70

4.8 Голям езиков модел GPT 71

4.9 Предизвикателства на различните езици 73

4.10 StructExtSum - Нов екстрактивен подход основан на структура на документа 74

4.11 Техническа имплементация 77

4.11.1 Архитектура на процеса 78

4.11.2 Използвани технологии и библиотеки 79

4.11.3 Инфраструктура и съхранение на данни 79

4.11.4 Отворен код и възпроизводимост 80

5 Резултати от проведените експерименти 81

5.1 Дизайн на експериментите 81

5.2 Резултати за резюмиране на правни текстове на английски . . . 81

5.3 Резултати за резюмиране на научни статии на английски 85

5.4 Резултати за резюмиране на правни текстове на английски и български език чрез анализ на структурата 86

5.5 Резултати за резюмиране на правни текстове на 24 езика с използване на големи езикови модели 88

5.6 Човешка оценка на качеството на резюметата 90

5.6.1 Екстрактивни подходи 90

5.6.2 Абстрактивни подходи 93

6 Заключение и бъдеща работа **96**

6.1 Приноси на дисертацията 96

6.1.1 Научни приноси 96

6.1.2 Приложни приноси 97

6.1.3 Публикации по темата на дисертацията 98

6.2 Посоки за бъдещи изследвания 99

Литература **102**

Списък на фигурите **110**

Списък на таблиците **112**

Декларация за оригиналност на резултатите **114**

Благодарности

Първо и най-вече, бих искал да изразя своята искрена благодарност към моя научен ръководител, проф. д-р Иван Койчев, за безценната подкрепа и насоки през целия период на моето докторантско обучение. Неговото вдъхновение, научна свобода и методична помощ бяха основополагащи за успеха на тази дисертация.

Изказвам благодарност и към моите съавтори на публикуваните статии – Милен Чечев, Гергана Лазарова и Тодор Цонков, за съвместната работа и ценния принос към научните изследвания, осъществени в хода на докторантурата. Работата с вас бе изключително вдъхновяваща и полезна.

Накрая, искам да благодаря на всички, които са ми помагали по този път, независимо дали със съвети, критики или с други форми на подкрепа.

1 Въведение

1.1 Мотивация и актуалност на темата

Генерирането на текст има за цел да създаде правдоподобен и четим текст на човешки език по зададени входни данни. През последните години се наблюдава значителен напредък в областта на дълбокото обучение, особено по отношение на обработката на естествен език (NLP). То се разглежда като обещаващ подход както в академичните среди, така и в индустрията. Особен принос за това имат невронните мрежи, изградени върху предварително обучени езикови модели. Моделите, използвани за генериране на текст, се съсредоточават върху решението на задачи като машинен превод, резюмиране на текст, задаване и отговаряне на въпроси, генериране на изцяло нов текст по зададена тема и други.

Дисертационният труд поставя фокус върху задачата за резюмиране на текст. Резюмирането на текст извлича важна информация от съдържанието, като същевременно гарантира, че смисълът е запазен. Това съкращава времето, необходимо за разбиране на дълги материали, без да се пропуска критична информация. Този процес е от особено значение в области, където бързият достъп до релевантна информация е ключов за вземането на решения.

Такъв пример е правната сфера, в която текстовете често са изключително дълги, технически и формулирани с висока степен на прецизност. Общото при правните документи – закони, наредби, съдебни решения и други – е, че те не са с фиксирана дължина и притежават сложна логическа и езикова структура. При тях се наблюдават големи вариации в стила, формата и обема на текста, както и в структурата и съдържанието на съпътстващите резюмета. Това прави автоматичното резюмиране още по-предизвикателно, особено ако се цели висока прецизност и правна коректност.

В допълнение, правната индустрия е изправена пред експоненциален ръст на обема информация, който трябва да се обработва ежедневно. Правни ана-

лизатори, съдии, адвокати и регулатори често са принудени да работят с хиляди страници документация в кратки срокове. В този контекст, автоматичното резюмиране се явява не просто удобство, а необходимост. То предлага възможност за ускорено запознаване със съдържание, навременно откриване на релевантна информация и подобряване на продуктивността при анализа на нормативна уредба и съдебна практика.

Същевременно, развитието на големи езикови модели създава нови възможности за прецизно и адаптивно резюмиране дори на дълги и многослойни текстове. Въпреки потенциала им, приложението им в правната сфера остава ограничено и слабо проучено, особено в многоезиков контекст. Това поражда нуждата от създаване на нови езикови ресурси и методологии, адаптирани към спецификите на правните документи и различията между езиците и правните системи.

Актуалността на темата се подсилва и от глобалните усилия за подобряване на достъпа до правосъдие чрез технологични решения. Възможността за автоматизирано резюмиране на правни документи допринася за по-прозрачно, достъпно и ефективно правно обслужване – както за професионалисти, така и за граждани.

1.2 Цел и задачи на дисертационния труд

Цел на дисертационния труд:

Разработване на методология за автоматично резюмиране на дълги правни текстове на различни езици чрез използване на съвременни езикови модели и адаптирани подходи.

Във връзка с това са определени следните задачи на изследването:

1. Създаване на ново множество от данни за правни текстове на 24 езика, които да допринесат за изследванията в областта на автоматичното резюмиране.

2. Анализ и оценка на съществуващите методи за автоматично резюмиране на дълги текстове.

3. Адаптиране на съществуващи и разработване на нов подход за генериране на резюмета.

4. Експериментиране с големи езикови модели като GPT за автоматично резюмиране на правни документи.

5. Оценка и анализ на качеството на получените резюмета.

1.3 Структура на дисертацията

Настоящата дисертация е организирана в няколко основни раздела, които целенасочено разглеждат темата на изследването, представят получените резултати и формулират направените изводи. Всеки раздел изпълнява специфична функция, като обхваща отделни аспекти на изследователския процес. Тази структура осигурява последователност и яснота, позволявайки на читателя да проследи логиката на изложението – от мотивацията и теоретичната рамка до използваните експериментални подходи, анализа на данните и заключителните разсъждения.

Първата глава, "Въведение", има за цел да въведе читателя в същността на изследваната тема и да очертае основната мотивация за провеждането на настоящото изследване. В тази част се разглежда актуалността на задачата за автоматично резюмиране на текстове, особено в контекста на правните документи и многоезиковите предизвикателства. Представени са основните цели и задачи, които стоят пред дисертационния труд, като се акцентира върху научния и приложния принос на изследването.

Втората глава, "Обзор на предметната област", обхваща теоретичния контекст на изследването. В нея се разглеждат основните концепции и алгоритми в областта на обработката на естествен език и автоматичното резюмиране. Специално внимание е отделено на методите за обработка на правни документи и

предизвикателствата, свързани с резюмирането на текстове на 24 езика. Анализирани са съществуващите подходи – екстрактивни и абстрактивни, както и ролята на предварително обучените езикови модели, като BERT, T5 и GPT. Тази глава включва и анализ на наличните набори от данни за правни текстове, като се акцентира върху тяхната пригодност за обучение на съвременни модели за резюмиране.

Третата глава, "Нов корпус от данни за правни текстове и техните резюмета", се фокусира върху събирането, изчистването и подготовката на данните, използвани в експериментите. Представен е нов набор от данни за правни текстове, обхващащ 24 езика, като подробно са описани характеристиките му, процесът на събиране и предизвикателствата при многоезиковата обработка. Включени са и сравнения между данните на различни езици, което предоставя основа за оценка на ефективността на подходите за резюмиране.

Четвъртата глава, "Разработени и адаптирани подходи за резюмиране на текстове", е централна за дисертацията и описва подробно методите и техниките, използвани за резюмиране на текстове. Включени са както базови подходи, така и нови иновативни техники, като StructExtSum, които използват структурата на документа за подобряване на резюметата.

Петата глава, "Резултати от проведените експерименти", включва разширен анализ на ефективността на използваните методи. Разглеждат се резултатите, постигнати върху различните езици, като се акцентира на приложимостта на подходите за специфични езици и типове документи. Добавен е раздел за оценка на резултатите, постигнати с помощта на големи езикови модели (GPT), и сравнението им с други подходи.

Шестата глава, "Заключение и бъдеща работа", обобщава основните резултати от проведеното изследване и посочва научните и приложни приноси на дисертацията. Тази част включва и насоки за бъдещи изследвания, като разширяване на набора от данни, подобрения в използваните модели и тяхното приложение в различни контексти. Разгледани са и етичните аспекти на автоматичното резюмиране на правни документи.

2 Обзор на предметната област

2.1 Специализирани термини и понятия

В тази секция са представени обяснения на основните термини и понятия, използвани в дисертацията. Те са подредени по тематична свързаност – от общи към по-специализирани.

- **Автоматично резюмиране** - Процесът на създаване на кратко резюме на текст, като се запазва основната информация и смисъл, но се премахват излишни детайли.
- **Екстрактивно резюмиране** - Подход за резюмиране, при който се извличат и комбинират части от оригиналния текст, като изречения или абзаци, за да се създаде резюме.
- **Абстрактивно резюмиране** - Подход за резюмиране, при който моделът генерира ново резюме, което може да съдържа фрази и идеи, които не са непременно в оригиналния текст.
- **Невронни мрежи** - Изчислителни модели, вдъхновени от начина, по който работи човешкият мозък, използвани за решаване на сложни задачи, като разпознаване на образи, обработка на език и други.
- **Дълбоки невронни мрежи (Deep Neural Networks, DNN)** - Вид невронни мрежи, които се състоят от множество слоеве, между които има неврони, които обработват и трансформират входните данни, като подобряват точността на моделите при решаването на сложни задачи. DNN са основа за много съвременни технологии, включително в областта на обработката на естествен език и разпознаване на изображения.
- **BERT (Bidirectional Encoder Representations from Transformers)** - Модел за обработка на естествен език, който използва трансформаторна архитектура и е предварително обучен върху големи текстови корпуси, за да генерира контекстуални представяния на думи.

- **GPT (Generative Pre-trained Transformer)** - Модел, използващ трансформаторна архитектура, който се обучава върху големи текстови данни и може да генерира текст, който имитира човешки език.

- **PreSumm** - Модел за резюмиране, който комбинира екстрактивни и абстрактивни подходи за генериране на текстови резюмета, използвайки дълбоки невронни мрежи.

- **Рекурентни невронни мрежи (Recurrent Neural Networks, RNN)** - Вид невронни мрежи, които обработват последователности от данни, като текст или реч, като използват вътрешни състояния за запомняне на информация от предходни стъпки.

- **Механизъм на самовнимание** - Механизъм в трансформаторни модели, който позволява на всеки елемент в последователността да обръща внимание на други елементи, без да зависи от реда на обработка на данни.

- **TF-IDF (Term Frequency-Inverse Document Frequency)** - Метрика за оценка на значимостта на думи в текстови документи, като се отчита честотата на появата на термини в даден документ и тяхната рядкост в целия корпус.

- **Токенизация** - Процесът на разделяне на текст в отделни части (токени), като думи, изречения или символи, които могат да бъдат използвани в анализа на текста.

- **Част на речта (Part of Speech - POS)** - Групирането на думите в текст в категории, като съществителни, глаголи, прилагателни и др., въз основа на тяхната синтактична роля в изречението.

- **Вграждания на думиs (Word Embedding)** - Преобразуването на думи в числови вектори с цел да се улесни тяхното обработване в алгоритми за машинно обучение и обработка на естествен език.

- **Конволюционни невронни мрежи (Convolutional Neural Networks, CNN)** - Вид невронни мрежи, използвани основно в об-

работката на изображения, които използват филтри за извличане на важни характеристики от данните.

- **Векторизация** - Процесът на преобразуване на текстови данни в числово представяне, което може да бъде обработвано от алгоритми за машинно обучение.

- **Трансформаторна архитектура** - Модел за обработка на естествен език, който използва механизъм на внимание и е доказал своята ефективност при задачи като машинен превод, резюмиране и генериране на текст.

- **Предварително обучени модели** - Модели, които са обучавани върху големи набори от данни и могат да бъдат използвани за решаване на различни задачи, като се финализират с обучение върху специфични данни.

- **Извличане на информация (Information Extraction)** - Процесът на извличане на структурирана информация от неструктурирани текстови данни, като например имена на места, дати или събития.

- **Метрики за оценка на резюмирането** - Критерии, използвани за измерване качеството на автоматично генерираните резюмета, като ROUGE и BLEU.

- **Модел за трансферно обучение** - Модел, който използва знанията, натрупани при решаване на една задача, за да се прилага към друга задача, като значително намалява нуждата от ново обучение.

- **Увеличаване на данни (Data Augmentation)** - Техника, използвана за разширяване на обема на данни чрез прилагане на различни трансформации, като завъртане, мащабиране или изкривяване на изображения, за да се подобри обучението на моделите.

- **Обработване на естествен език (Natural Language Processing, NLP)** - Област на изкуствения интелект, която се занимава с обработката и анализа на текстови и говорни данни на човешки език.

- **Long Short-Term Memory (LSTM)** - Специализиран вид рекурентна невронна мрежа, която може да запомня информация за дълги периоди, като решава проблема с "изчезващите градиенти"в традиционните RNN мрежи.
- **Gated Recurrent Unit (GRU)** - Подобен на LSTM вид рекурентна невронна мрежа, но с по-малка сложност и по-бърза обработка, която използва механизъм за забравяне и актуализиране на паметта.
- **Механизъм на внимание (Attention Mechanism)** - Техника в невронните мрежи, която позволява на моделите да се фокусират върху важни части от входния текст, като игнорират незначителни детайли.
- **Множество глави на внимание (Multi-head Attention)** - Разширение на механизма на внимание, което позволява на моделите да учат различни представяния на входа, използвайки различни глави на внимание.
- **Финално обучение (Fine-tuning)** - Процес на дообучаване на предварително обучен модел върху специфична задача или набор от данни, за да се постигне по-добра производителност за съответната задача.

2.2 Обработка на естествен език

Обработката на естествен език (NLP) е област, която съчетава лингвистиката, компютърните науки и машинното обучение. Тя се фокусира върху разработването на изчислителни техники за автоматичен анализ, разбиране и генериране на съдържание на човешки език (Cambria and White 2014). Приложенията на сферата включват превеждане на текст от един език на друг, категоризация на текст, филтриране на спам, извличане на структурирана информация, резюмиране, диалогови системи за автоматични разговор с потребители и преобразуване на реч в текст (Khurana *et al.* 2017).

Визията на сферата е да изгради машини, които могат интелигентно да четат, разбират и интегрират големи количества човешко знание и да ни по-

магат с ежедневни задачи, свързани с тях (Hirschberg and Manning 2015), като например предоставяйки ни персонализирано резюме на документ. Обработката на текст става особено важна в ерата на големи данни, където натрупването на текстова информация, налична в мрежата, нараства неимоверно в последните години, не само за общи познания като Wikipedia или новините, но и в научната литература и много други сфери.

Ранните изследвания в областта се фокусират върху задачата за дефиниране на ръчни правила за трансформация на езикови единици. Тази задача се оказва предизвикателна, поради голямата променливост и двусмисленост на човешкия език. Това измества фокуса върху статистическите подходи, които напоследък са по-често срещани (Jurafsky and Martin 2008). Много от тези методи разчитат на разнообразни лексикални характеристики, като n-грами (поредица от n думи), последователности от части на речта и други. Тези характеристики могат да бъдат използвани да захранват стандартни алгоритми като линейна и логистична регресия. Те обаче не отвеждат до търсения краен резултат.

Успехът в сферата се дължи на наличието на големи количества данни за обучение.

През последните години дълбоките невронни мрежи правят революция в много области на машинното обучение, включително в обработката на естествен език. Те постигат това като трансформират входните данни (текст) в междинно представяне (вектори), които са оптимизирани спрямо функция на загуба. Параметрите на модел за задълбочено обучение се научават да се самоорганизират от край до край, за да минимизират тази загуба. Този подход намалява необходимостта от изготвяне на характеристики, специфични за задачата - за задълбочено прилагане са необходими малко или никакви предварителни познания в лингвистичната област.

Успехът на дълбоките модели доведе до прилагането им в редица задачи в обработката на текст, като анализ на настроението в текст (sentiment) (Socher *et al.* 2013), генериране на заглавие за изображение (K. Xu *et al.* 2015)

и невронен машинен превод (Bahdanau, Cho, and Bengio 2014). Освен това има много потенциални приложения на дълбоки невронни мрежи в биоинформатиката и обработката на медицински език (Mamoshina *et al.* 2016), например за автоматизирано откриване на лекарства или за навигиране в огромните неструктурирани данни, налични в научната литература.

Големите езикови модели от серията GPT (Generative Pre-trained Transformer) (OpenAI *et al.* 2024) са сред водещите технологии в обработката на естествен език, които значително надграждат съществуващите методи за автоматично резюмиране, превод и генериране на текст. Тези модели се базират на трансформаторна архитектура, която позволява ефективна обработка на дълги текстови последователности и запазване на контекстуални зависимости. Обучени върху разнообразни и мащабни текстови корпуси, GPT моделите са способни да извършват сложни задачи, които изискват разбиране на контекста и генериране на смислени и съгласувани изрази.

Особеност на GPT моделите е тяхната универсалност и адаптивност, което ги прави приложими в многоезикови и домейн-специфични контексти. Благодарение на предварителното си обучение върху голямо количество данни на различни езици, тези модели показват отлични резултати дори при обработката на езици с ограничени ресурси, като български. Възможността им да запазват точността и смисловата пълнота на текста е особено ценна в задачите за резюмиране на юридически документи, където изискванията за прецизност са изключително високи. Това ги утвърждава като мощен инструмент за съвременните изследвания в областта на автоматичното резюмиране.

2.3 Методи за обработка на естествен език чрез дълбоки невронни мрежи

Този раздел разглежда важни методи и концепции в NLP, които са от значение за този труд.

2.3.1 Текстови вектори (вграждания на думи)

Една от основните причини за скорошния успех на дълбоките невронни мрежи в NLP е въвеждането на векторно представяне на текста т.нар. вграждания (embeddings). Вгражданията преобразуват единици текст към вектор в многомерно пространство с размерност d (например с размер 100-500). Методите за вграждане на думи (Mikolov *et al.* 2013a; Pennington, Socher, and Manning 2014) се фокусират върху научаването на вграждания на отделни думи от големи количества неструктуриран текст, като обикновено разчитат на функция за самоконтролирано обучение, като например предсказване на дума, при даден съпътстващ текст. Вграждания на изречения (Le and Mikolov 2014; Pagliardini, Gupta, and Jaggi 2018) се фокусира върху моделирането на цели изречения. Текстовите вграждания са полезни за редица задачи като отговаряне на въпроси (Rajpurkar *et al.* 2016), извод на естествен език (Bowman *et al.* 2015), извличане на най-близкия съсед (Kusner *et al.* 2015) и корекция на текст (Nikolov and Hahnloser 2019).

Популярен подход за генериране на вектори от думи е Word2Vec (Mikolov *et al.* 2013b). Word2vec е двуслойна невронна мрежа, която обработва текст чрез "векторизиране" на думи. Неговият вход е текстов корпус, а изходът му е набор от вектори: характерни вектори, които представляват думи в този корпус. Word2vec не е дълбока невронна мрежа, но той превръща текста в числова форма, която дълбоките невронни мрежи могат да разберат.

Приложенията на Word2vec са разнообразни. Може да се приложи към гени, код, харесвания, плейлисти, графики на социалните медии и други вербални или символни серии, в които могат да се разпознаят модели.

Причината за това е, че думите са просто отделни състояния като другите данни, споменати по-горе, и ние просто търсим преходните вероятности между тези състояния: вероятността те да се появят едновременно.

Целта и полезността на Word2vec е да групира векторите на подобни думи заедно във векторно пространство. Тоест открива сходствата математически.

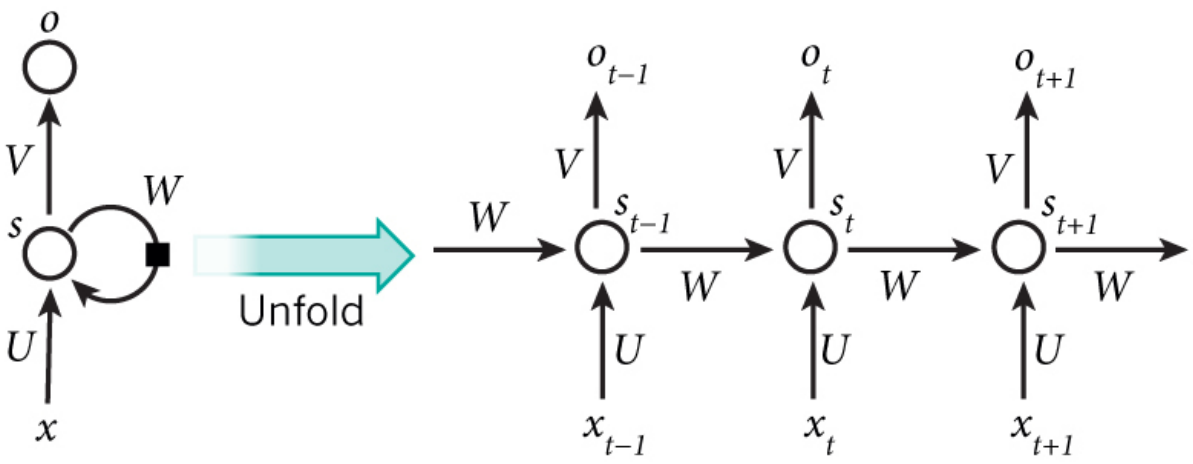
Word2vec създава вектори, които са разпределени числени представяния на характеристики на думи, характеристики като контекста на отделни думи. Прави го без човешка намеса.

Като се има предвид достатъчно данни, употреба и контекст, Word2vec може да направи много точни предположения за значението на дадена дума въз основа на минали изяви. Тези предположения могат да се използват за установяване на асоциация на дума с други думи (например “мъж” е за “момче”, какво “жена” е за “момиче”) или групиране на документи и класифициране по теми. Тези клъстери могат да формират основата за търсене, анализ на настроенията и препоръки в различни области като научни изследвания, правни открития, електронна търговия и управление на взаимоотношенията с клиенти.

Резултатът от невронната мрежа Word2vec е речник, в който всеки елемент има вектор, прикрепен към него, който може да бъде подаден в дълбока невронна мрежа или просто да бъде запитван за откриване на връзки между думите.

2.3.2 Рекурентни невронни мрежи

Рекурентната невронна мрежа (RNN) носи името си от специфична част в невронната мрежа, при която текущият вход (x_t) и нейното предишно скрито състояние (s_{t-1}) и тегло (W) се използват за прогнозиране на следващия изход (s_{t+1}) . RNN са повтарящи се, защото изпълняват една и съща операция на всеки последователен вход. Те дават добри резултати, когато входовете са последователни, като текст, реч и т.н. RNN могат да бъдат описани най-добре като невронни мрежи със състояния на паметта. Фигура 1 показва разгъната рекурентна невронна мрежа и тя е напълно свързана мрежа.



Фигура 1: Разгъната рекурентна невронна мрежа. [1]

Тази разгъната мрежа може да се използва за решаване на проблем с предсказване на следваща дума. Например, "Цветът на небето е XXX", така че за да предвидим "синьо", можем да използваме RNN. Последователността съдържа 5 думи в изречение, така че мрежата ще бъде разгъната в 5-слойна невронна мрежа, всеки невронен слой за една дума. Следващите стъпки описват характеристиките на тази рекурентна невронна мрежа.

- U и V са активационните функции, а W е вектор на теглата. U, V, и W са еднакви за всички слоеве.
- x_t е вход към мрежата, в момент t. Например x_1 може да бъде вход вектор, съответстващ на втората дума (например "цвят") в изречението.
- s_t е скрито състояние, на времева стъпка t. Това е състоянието на паметта на мрежата. Изчисляването на s_t се основава на текущия вход и входа от предишното скрито състояние. Формулата за изчисляване на s_t е следната:

$$S_t = f(x_t * U + W * s_{t-1})$$

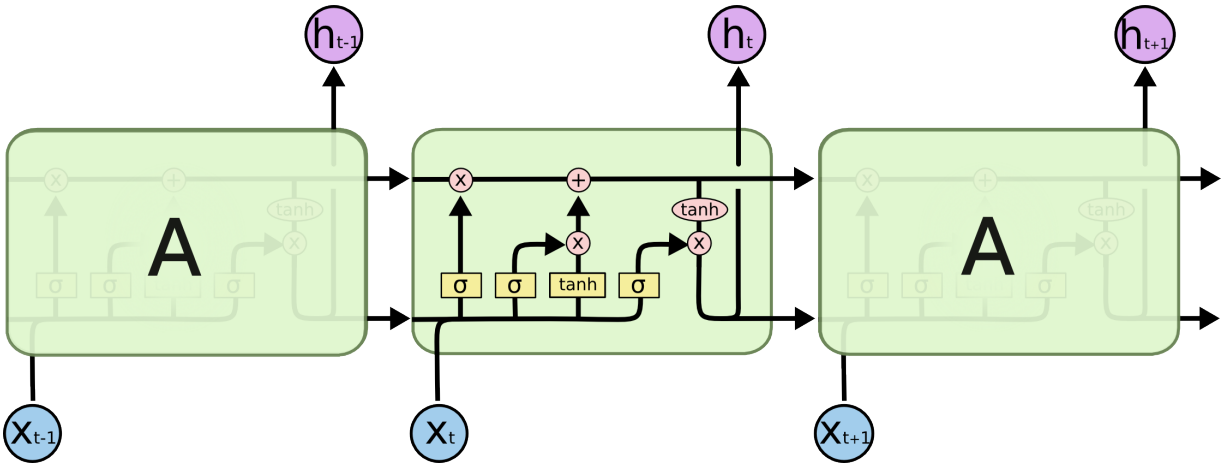
- f е нелинейна функция като ReLU и tanh. s_{t-1} се инициализира на нула за първото скрито състояние.

[1] Източник: https://dennybritz.com/posts/wildml/recurrent-neural-networks-tutorial-part-1/

- В момент t, o_t е изходното състояние.

$$o_t = softmax(V * s_t)$$

Long Short-Term Memory (LSTM): Дългата краткосрочна памет (LSTM) е специален вид рекурентна невронна мрежа (Hochreiter and Schmidhuber 1997). LSTM адресира ограничението на RNN, което се нарича "дългосрочни зависимости". Например, ако има последователност от думи в изречение, например "Птиците летят в небето", RNN очевидно ще може да предвиди, че последната дума е "небе". Ако обаче става дума за дълго изречение като "Роден съм в Америка, следователно говоря свободно английски". Сега тук, ако RNN иска да предскаже думата "английски" в изречението, трябва да има представа какво е съдържал текстът до момента, което RNN не помни поради недостатъчно количество памет. Този проблем се дефинира като дългосрочни зависимости. LSTM е създаден, за да реши този проблем чрез въвеждане на допълнителна клетка в мрежата, отговорна да пази дългосрочната памет. Фигура 2 илюстрира как работи.



Фигура 2: Long short-term memory (LSTM). [2]

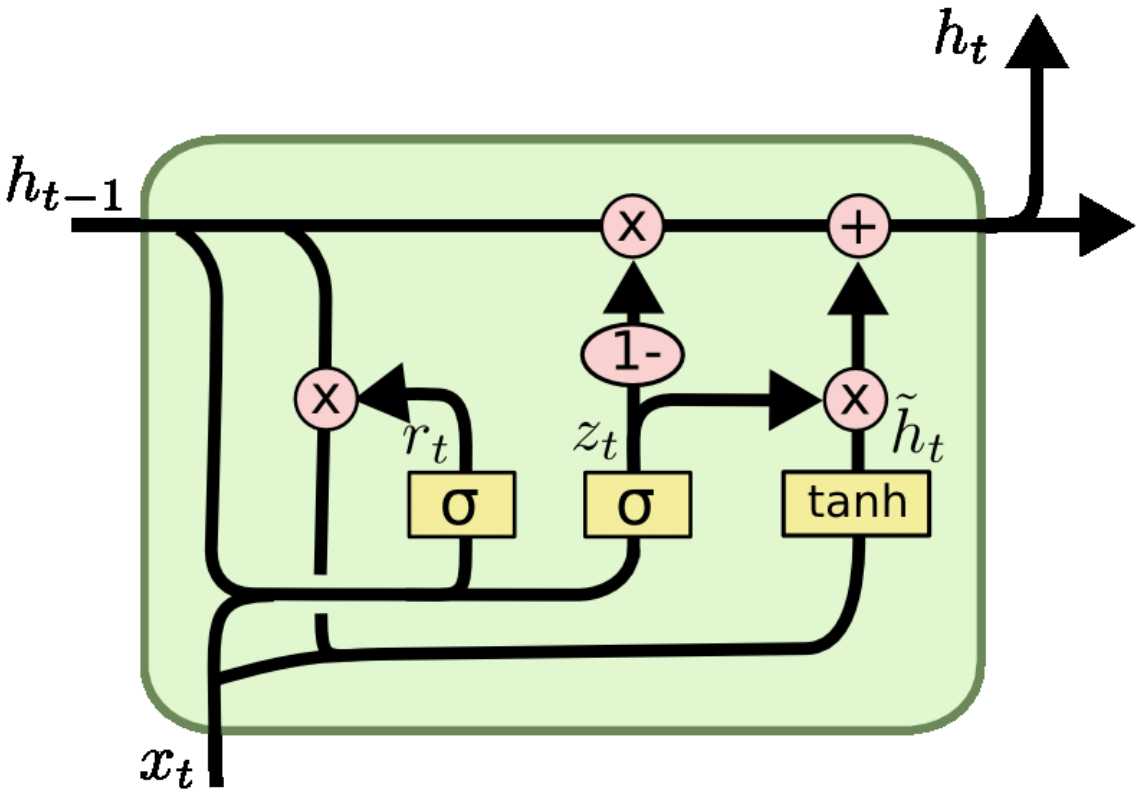
Например, нека се съсредоточим върху общата работа на LSTM. Всеки мрежов слой приема три входа. x_t е входът на времева стъпка t. h_{t-1} е изходът

[2]Източник: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

от предишната единица, а c_{t-1} е паметта на предишната единица и това е най-важната единица в LSTM, което я прави различна от RNN. h_t е изходът на текущия слой, а c_t е паметта на текущия слой.

Gated Recurrent Unit (GRU): Gated Recurrent Unit (GRU) (Cho *et al.* 2014) е затворен механизъм в рекурентната невронна мрежа (RNN), подобен на LSTM, който има опция за забравяне. GRU в определени случаи има предимства пред дългосрочната краткосрочна памет (LSTM). GRU използва по-малко памет и е по-бърз от LSTM, но LSTM е по-точен при използване на набори от данни с по-дълги последователности.

GRU използва два порта, порт за актуализиране и порт за нулиране. Тези портове решават каква информация се допуска до изхода и могат да бъдат обучени да задържат по-стара информация. Това позволява предаване на важна информация, за да се генерира по-добър изход. Фигура 3 илюстрира нагледно как работи.



Фигура 3: Gated recurrent unit (GRU). [3]

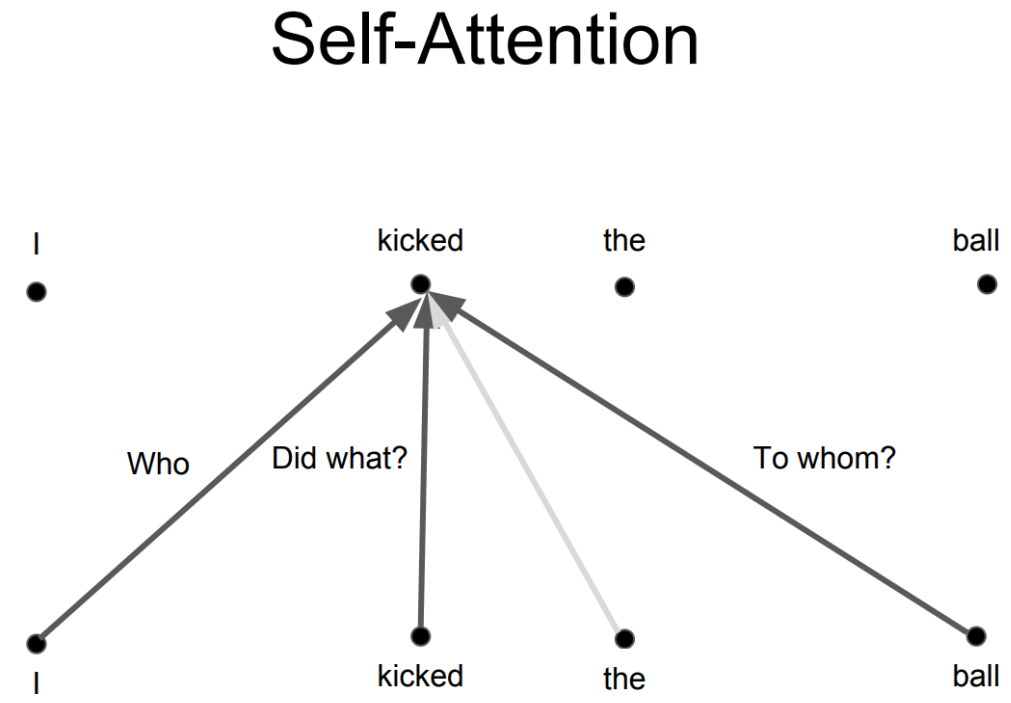
2.4 Архитектура на Трансформатор

Преди Vaswani *et al.* 2017 да представи Transformer архитектурата, рекурентните невронни мрежи бяха най-популярната и мощна архитектура за структурата Encoder-Decoder за решаване на проблеми от сферата на обработката на естествен език. В наши дни модерните и бързи изчислителни устройства като графични процесори (GPU) и тензорни процесори (TPU) разчитат на паралелни изчисления. Въпреки това, последователният характер на RNN принуждава изчислението да обработва изречения дума по дума, което прави RNN невъзможни за паралелизиране. За да преодолее този недостатък на RNN, трансформаторът използва механизъм на самоконтрол, който позволява на енкодера и декодера да отчитат всяка дума от цялата входна последователност.

2.4.1 Механизъм на самовнимание

Самовниманието (self-attention) е механизъм, който представя връзки между думите в рамките на едно изречение.

[3]Източник: https://www.researchgate.net/figure/Gated-Recurrent-Unit-GRU_fig4_328462205



Фигура 4: Механизъм на самовнимание (примерно изречение). [4]

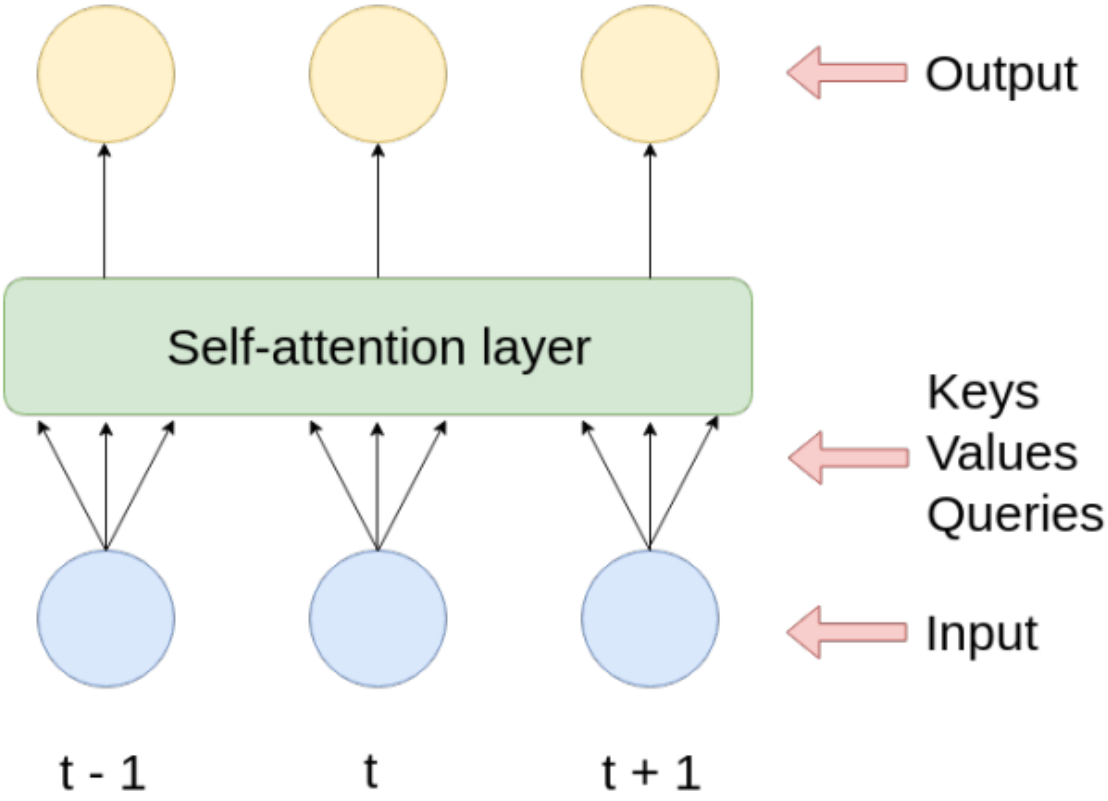
Фигура 4 показва пример за механизъм на самовнимание. В изречението "Аз ритнах топката." моделът може да разгледа другите думи, за да придобие по-добра представа за думата ритнах - Кой, направи Какво, на Кого. Механизмът на вниманието (Vaswani *et al.* 2017) може да се опише със следното уравнение:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{sqrtd_k})V$$

където Q е матрица, която съдържа набор от заявки, K и V са матрици с ключове и стойности. По отношение на енкодер-декодер, заявката и ключът обикновено са скритото състояние съответно на декодера и енкодера. Входовете

[4]Източник: https://towardsdatascience.com/transformers-141e32e69591

се използват като ключове и стойности. Стойността е нормализирано тегло, представящо колко внимание получава съответният ключ.



Фигура 5: Механизъм на самовнимание (архитектура).

От страна на енкодера, по-добро представяне на входа x_t на времева стъпка t се генерира чрез самовнимание, използвайки всички други входове x_1, x_2,..., x_n, където n е дължината на последователността. Тъй като тази работа може да се извърши за всички входни стъпки паралелно, трансформаторът е по-подходящ от рекурентните невронни мрежи за паралелизиране. Освен това, нивото на самовнимание свързва всички позиции със сложността на $O(1)$ брой последователни операции, по-евтино от $O(n)$ на RNN.

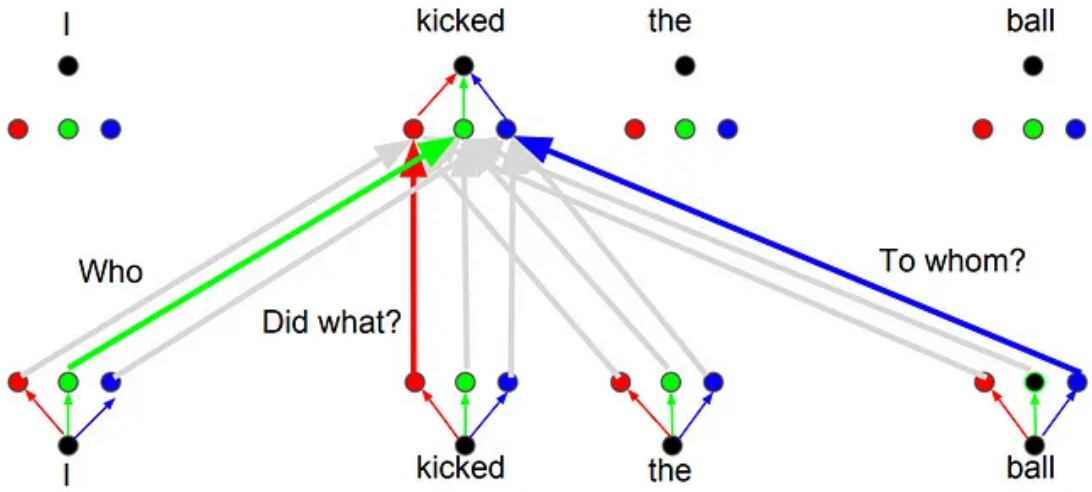
От страната на декодера, по време на стъпка t, x_t е текущият вход като заявка. Всички минали заявки x_1, x_2,..., x_{t-1}, се комбинират, за да бъдат ключове и стойности на слоя за самовнимание. Архитектурата на механизма на

самовнимание е представена схематично на Фигура 5.

2.4.2 Механизъм на внимание с много глави

Вместо една претеглена сума от стойностите, механизмът за внимание с много глави (Multi-Head Attention) изчислява множество претеглени суми, за да улови различни аспекти на входа.

За да научи различни представяния, всяка глава е уникална линейна трансформация на входното представяне като заявка, ключ и стойност. След това вниманието с мащабирани точки се изчислява h пъти успоредно, което го прави т.нар. много глави. След това изходите се свързват. Накрая се прилага една единствена линейна трансформация, както е показано на Фигура 6. Вниманието с много глави може да научи свързана информация от различни подпространства за представяне, тъй като всяка от тези заявки, ключове, набори от стойности се инициализира на случаен принцип.



Фигура 6: Механизъм на внимание с много глави (примерно изречение). [5]

[5]Източник: https://towardsdatascience.com/transformers-141e32e69591

2.4.3 Позиционно кодиране

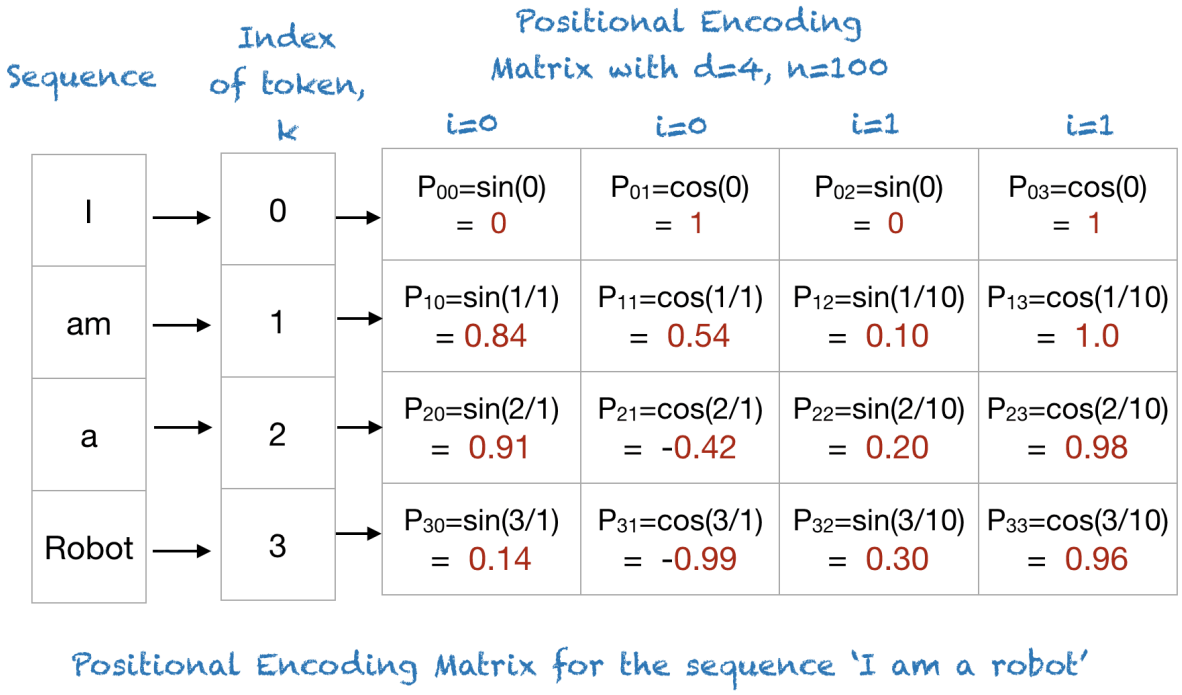
При рекурентните невронни мрежи, думите се подават последователно към модела, всеки токен е пореден. При вниманието с много глави изходът на всеки слой се изчислява последователно и независимо от реда на думите. Неефективно е да се моделира информацията за последователността без специален ред или позиция. За да отчете реда на думите във входната последователност, архитектурата Трансформатор (Transformer) добавя вектор към всеки входен вектор, наречено позиционно кодиране (Positional Encoding). Позиционното кодиране от модела на трансформатора се изчислява чрез синусови и косинусови функции на различни честоти като:

$$PE_{(pos,2i)=sin(\frac{pos}{10000^{\frac{2i}{d_{model}}}})}$$

$$PE_{(pos,2i+1)=cos(\frac{pos}{10000^{\frac{2i}{d_{model}}}})}$$

където i представлява векторния индекс, който разглеждаме, pos представлява позицията, d_{model} е измерението на входните вграждания. Всяко измерение на позиционното кодиране образува синусоида, което позволява на модела да се справи с по-дълги последователности от думи. PE_{pos+k} може да се изчисли чрез линейна функция на PE_{pos} с отместване k, така че относителната позиция между различни вграждания може да бъде изведена на ниска цена.

Фигура 7 описва пример за позиционно кодиране за входното изречение “Аз съм робот”. Всяка входна дума първо се превръща във вектор с помощта на алгоритъм за кодиране с размер 4. След това се прилага позиционно кодиране, за да се получи съответният изходен вектор със същия размер.



Фигура 7: Позиционно кодиране за изречението "Аз съм робот.". [6]

2.4.4 Архитектура на Трансформатор

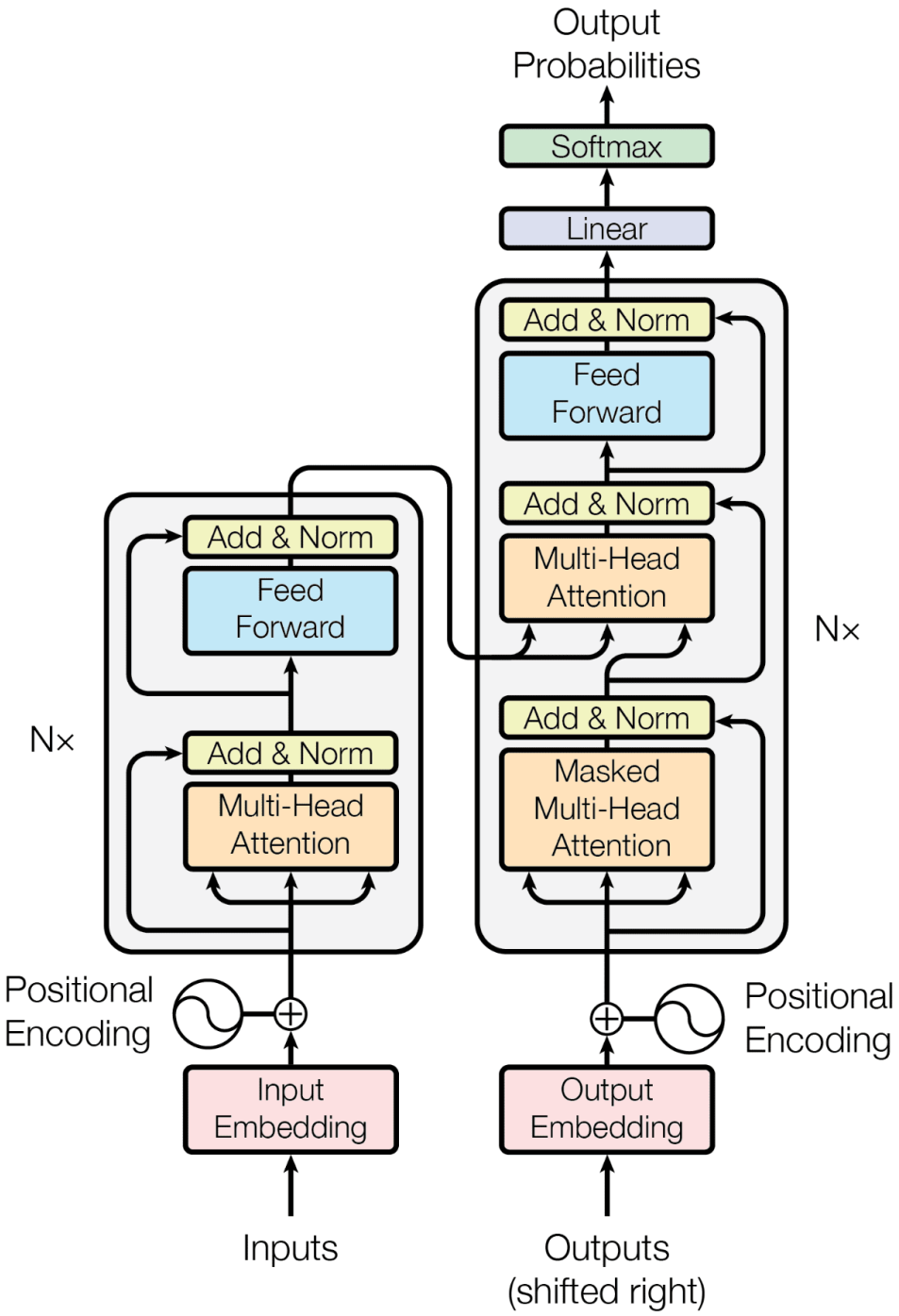
Моделът на трансформатора (Transformer) с неговите компоненти за енкодер и декодер е илюстриран на Фигура 8. И енкодерът, и декодерът могат да бъдат съставени от N_x идентични подчасти. Стекът на енкодера и стекът на декодера споделят един и същ брой N_x.

Блокът на енкодера е стек от N_x идентични слоеве. Всеки слой има подслой на механизма за самовнимание с множество глави, последван от стандартен напълно свързан подслой.

Декодерът също е стек от N_x идентични слоеве. В допълнение към двата подслоя във всеки слой на енкодера, декодерът има допълнителен подслой за

[6] Източник: https://machinelearningmastery.com/a-gentle-introduction-to-positional-encoding-in-transformer-models-part-1/

маскирано внимание с множество глави, за да избегне използването на информация, която още не е прочетена.



Фигура 8: Архитектура на Трансформатор (Transformer) с един слой енкодер (отляво) и един слой декодер (отдясно). [7]

[7]Източник: https://machinelearningmastery.com/the-transformer-model/

2.5 Предварително обучени езикови модели

Едно от най-важните предимства на архитектурата Трансформатор е, че предоставя възможност за пренасяне на учението (Transfer learning). Моделите могат да бъдат натренирани на голямо количество текст и научените тегла да бъдат използвани за други задачи с опцията част от модела да се тренира към специфични задачи.

2.5.1 BERT

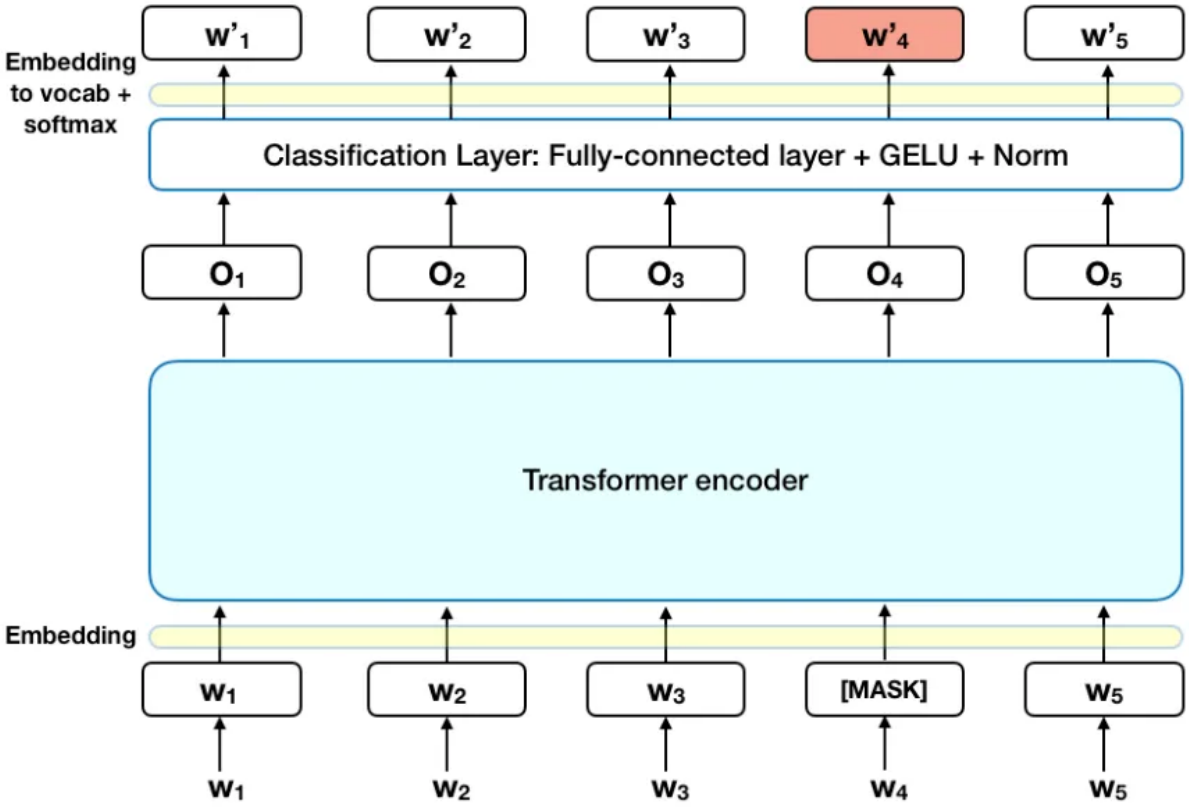
BERT (Devlin *et al.* 2018) използва архитектурата Трансформатор, механизъм на внимание, който научава контекстуални връзки между думи (или поддуми) в текст. В чистата си форма трансформаторът включва два отделни механизма — енкодер, който чете въведения текст, и декодер, който произвежда прогноза за задачата. Тъй като целта на BERT е да генерира езиков модел, е необходим само механизмът на енкодера.

За разлика от рекурентните модели, които четат въведения текст последователно (отляво надясно или отдясно наляво), енкодерът чете цялата последователност от думи наведнъж. Следователно се счита за двупосочен, въпреки че би било по-точно да се каже, че е непосочен. Тази характеристика позволява на модела да научи контекста на дадена дума въз основа на цялото ѝ обкръжение (отляво и отдясно на думата).

Фигура 9 показва енкодера Трансформатор. Входът е поредица от токени, които първо се вграждат във вектори и след това се обработват в невронната мрежа. Резултатът е поредица от вектори с размер H, в която всеки вектор съответства на входен токен със същия индекс.

Когато се обучават езикови модели, има предизвикателство да се дефинира цел за прогнозиране. Много модели предвиждат следващата дума в последователност (например “Детето се прибра от XXX”), насочващ подход, който по същество ограничава ученето на контекста. За да преодолее това предизвика-

телство, BERT използва две стратегии за обучение:



Фигура 9: Архитектура на BERT [8]

Езиково моделиране чрез маскиране

Преди подаването на последователности от думи в BERT, 15% от думите във всяка последователност се заменят с токен $[MASK]$. След това моделът се опитва да предскаже оригиналната стойност на маскираните думи въз основа на контекста, предоставен от другите немаскирани думи в последователността. От техническа гледна точка предвиждането на изходните думи изисква:

- Добавяне на класификационен слой върху изхода на енкодера.
- Умножаване на изходните вектори по матрицата за вграждане, трансформирането им в измерение на речника.

[8]Източник: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

- Изчисляване на вероятността за всяка дума в речника със $softmax$.

Функцията за загуба на BERT взема под внимание само предвиждането на маскираните стойности и игнорира предвиждането на немаскираните думи. Вследствие на това моделът се научава по-бавно от насочените модели, характеристика, която се компенсира от неговата повишена осведоменост за контекста.

Предсказване на следващото изречение

В процеса на обучение на BERT моделът получава двойки изречения като входни данни и се научава да прогнозира дали второто изречение в двойката е следващото изречение в оригиналния документ. По време на обучението 50% от входовете са двойка, в която второто изречение е следващото изречение в оригиналния документ, докато в останалите 50% произволно изречение от корпуса е избрано като второ изречение.

За да помогне на модела да различи двете изречения в обучението, входът се обработва по следния начин, преди да влезе в модела:

1. $[CLS]$ токен се вмъква в началото на първото изречение и $[SEP]$ токен се вмъква в края на всяко изречение.
2. Към всеки токен се добавя информация дали е част от изречение A или изречение B. Информацията за изреченията е подобно по концепция на вграждането на токени с две възможни стойности.
3. Към всеки токен се добавя информация за позицията му.

За да се предвиди дали второто изречение наистина е свързано с първото, се изпълняват следните стъпки:

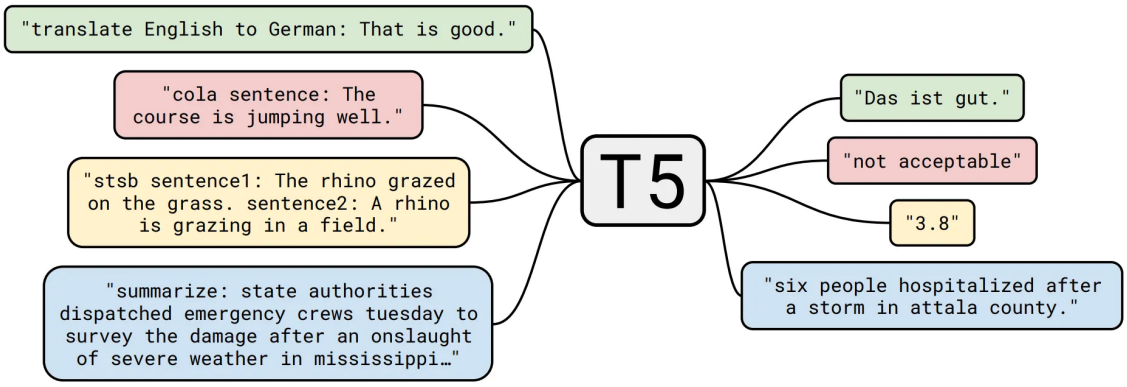
1. Цялата входна последователност преминава през модела Трансформатор.
2. Резултатът от токена $[CLS]$ се трансформира във вектор с форма $2x1$, като се използва прост класификационен слой.

3. Изчисляване на вероятността с помощта на $softmax$ функция.

При обучение на модела BERT, маскирането на думи и предсказването на следващо изречение се обучават заедно с цел минимизиране на комбинираната функция на загуба на двете стратегии.

2.5.2 Езиков модел T5

T5 (Raffel *et al.* 2019) Моделът за прехвърляне на текст към текст предлага преформатиране на всички задачи за обработка на естествен език в унифициран формат текст към текст, при който входът и изходът винаги са текстови низове. Това форматиране прави модела T5 подходящ за множество задачи. Както може да се види на Фигура 10, той приема като вход текст отляво за различни задачи и извежда текста за съответната задача. Авторите обозначават обучителния корпус като C4 (Colossal Clean Crawled Corpus), който представлява изчистена версия на Common Crawl с размер от приблизително 700 GB. Авторите описват почистването като извличане само на английски текст, премахване на кодови редове, дедупликация и други подобни. Това е висококачествен предварително обработен корпус на английски език, който е достъпен за изтегляне. Също така, моделът T5, предварително обучен на C4, постига най-добри резултати при много задачи, като същевременно е достатъчно гъвкав, за да бъде фино настроен за различни задачи от специфична област.



Фигура 10: Възможности на модела T5 [9]

T5 се обучава със същата цел като тази на BERT, която е маскиран езиков модел с малка модификация към него. Маскираните езикови модели са двупосочни модели, във всеки един момент t представянето на думата се извлича както от левия, така и от десния ѝ контекст. Тънката разлика, която T5 използва, е да замени множество последователни токени с една ключова дума $Mask$, за разлика от BERT, който използва токен $[MASK]$ за всяка дума. Оригиналният текст се трансформира във входни и изходни двойки. Тъй като крайната цел е да се обучи модел, който въвежда текст и извежда текст, целите са проектирани да произвеждат последователност, за разлика от BERT, който се опитва да изведе една дума.

Моделът е обучен на корпуса C4 със същата цел като част от предварителното обучение. След това е фино настроен за различни задачи като езиков превод, обобщение, сходство на изречения и други. Допълнителната настройка е направена чрез показване на модела на текстови двойки за вход и изход със специфичен за задачата префикс - текст, добавен към всеки вход. Например — translate English to German: <текст>, добавянето на такъв префикс дава възможност на модела да настрои теглото си за конкретна задача и да произведе очаквания резултат само за тази задача, като стесни обхвата на генериране.

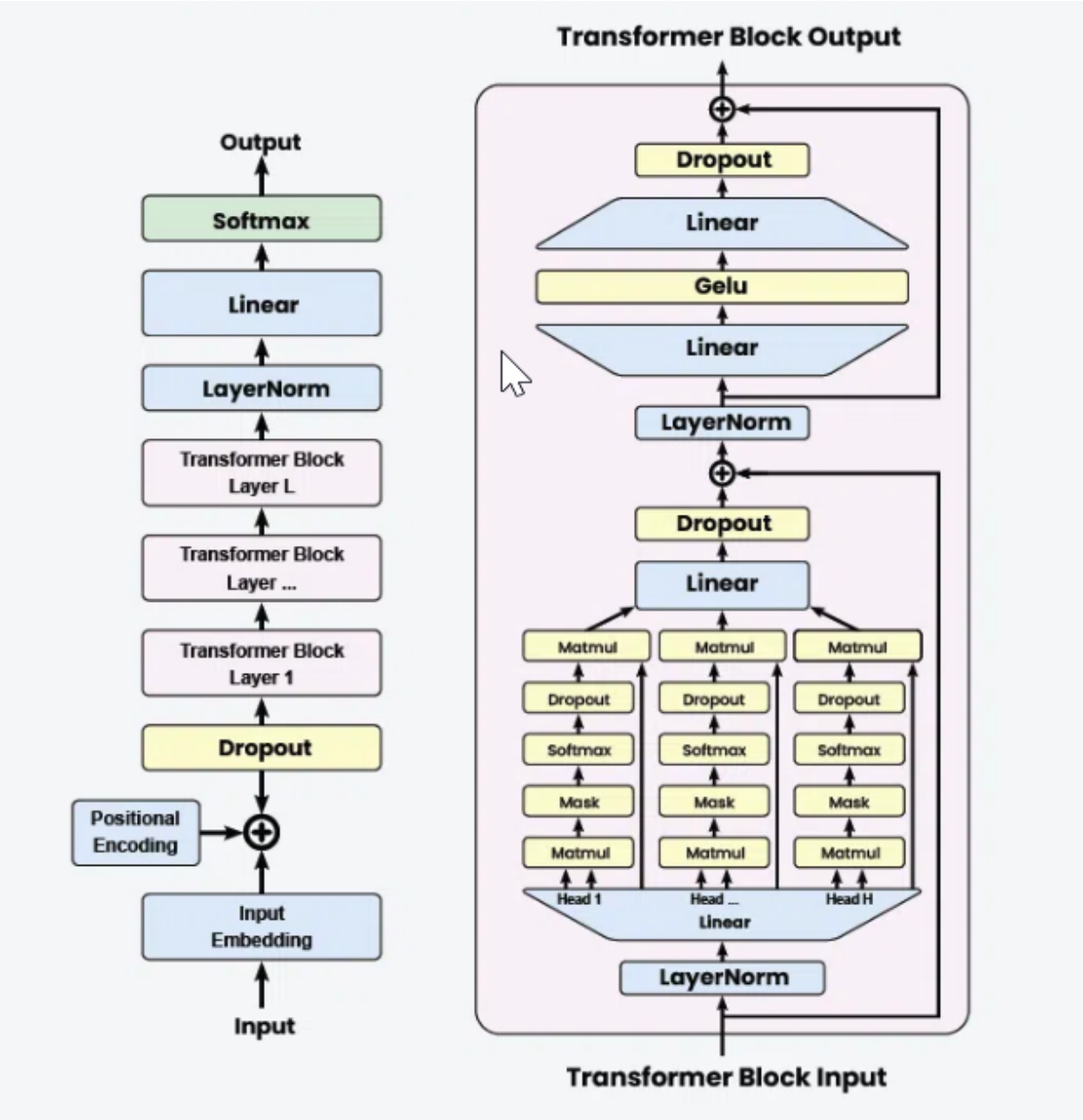
[9]Източник: https://towardsdatascience.com/understanding-t5-model-text-to-text-transfer-transformer-model-69ce4c165023

Всички задачи по същество споделят една и съща цел, процедура на обучение и процес на декодиране. Авторите също твърдят, че не са открили нито един случай, при който моделът да се е объркал и да е извел нещо напълно произволно или очаквано от друга задача.

2.5.3 Големи езикови модели, GPT

Големите езикови модели (Large Language Models, LLMs) са съвременни системи за обработка на естествен език, които са в основата на напредъка в редица задачи като генериране на текст, автоматичен превод, въпроси и отговори, както и резюмиране. Тези модели са базирани на архитектурата Трансформатор. За разлика от традиционните методи за обработка на текст, които разчитат на по-ограничени вектори или правила, трансформаторите позволяват моделиране на дългосрочни зависимости в текста, като това е особено важно при работа с дълги и комплексни документи.

Големите езикови модели обикновено преминават през обширен процес на предварително обучение върху разнообразни текстови корпуси. Това включва обработка на данни от много домейни и жанрове, което дава възможност на моделите да разбират различни стилове на писане и да се адаптират към специфични задачи. След това, чрез допълнително обучение или специализирано настройване, моделите се подготвят за конкретни цели, като се увеличава точността им и се подобрява съответствието с изискванията на задачата.



Фигура 11: Архитектура на GPT. [10]

С увеличаване на мащаба и сложността на тези модели, се увеличават и техните възможности за по-добро разбиране на текст и генериране на смислени и контекстуално правилни отговори. Съвременните LLM модели, като различните версии на GPT (OpenAI *et al.* 2024), са способни да обработват

[10]Източник: https://www.geeksforgeeks.org/introduction-to-generative-pre-trained-transformer-gpt/

текстове на множество езици и да поддържат качеството на езиковото разбиране и генерация дори при езикови комбинации, които не са традиционно добре представени в текстовите корпуси. Това прави LLMs подходящи за многоезикови задачи, при които има изискване за висока точност и естественост на генерирания текст.

Фигура 11 показва архитектурата на GPT. Основната част на архитектурата е енкодер-декодер структура, в която се използват механизми на вниманието, по-специално самовнимание (self-attention), за да улови значимите взаимовръзки между думите в даден контекст. Моделът се състои от множество слоеве от т.нар. глави за внимание (multi-head attention) и напълно свързани невронни мрежи, които работят в паралел, за да обработват различни аспекти на входния текст. Това позволява паралелна обработка и значително подобрява ефективността при генерирането на текст. Архитектурата включва и позиционно кодиране, което позволява на модела да взема предвид реда на думите, въпреки че обработката не е последователна. Тези характеристики позволяват на GPT моделите да обобщават и интерпретират текст, като създават смислово последователни и свързани фрагменти.

В настоящето изследване използвам GPT-3.5-turbo, който към момента на експериментите представлява балансиран избор между цена и производителност. Тази версия на модела предоставя усъвършенствани възможности за разбиране и генерация на текст, като същевременно е оптимизирана за приложения, изискващи бърза реакция и ефективно използване на ресурси.

2.6 Автоматично резюмиране на текстове

В настоящата теза се занимаваме с проблема за автоматичното резюмиране. Целта е въведеният текст да бъде намален и обобщен в по-кратка, компресирана версия, която улавя най-подходящите части от входните данни, без загуба на съществена информация (Nenkova and McKeown 2011). Например може да се резюмира дадена статия с цел да се извлече най-важната информация от нея. Друг пример е резюмиране на правни документи, които са дълги

и трудни за четене. Качественото автоматизирано обобщаване на текстове става още по-важно с всеки изминал ден тъй като количеството текст, което се продуцира ежедневно се увеличава с големи темпове.

Резюмирането е трудна задача дори за хората, а има много професионалисти в различни индустрии, които пишат резюмета като част от работата си (Mani 2002). Тази трудност идва от факта, че обобщението изисква множество сложни компоненти за разбиране на езика, които включват подбор, оценка, агрегиране и реорганизиране на информация, последвано от нейното компресиране, обобщение и/или перифразиране. В допълнение към това, това трябва да се случи на множество (евентуално абстрактни) нива, като изречение, параграф, раздел и документ. Съвременните системи могат предоставят полезни обобщения, които могат да спестят време, но има случаи като специфични области, непопулярни езици или корпуси с недостатъчно данни, върху които не успяват да сработят достатъчно добре. Оценяването на качеството на изходните резюмета е важен проблем поради субективния характер на тази задача, където множество не е безпристрастно дефинирано кое резюме е добро и кое не.

Автоматичното резюмиране води до редица практически приложения. Ранните примери включват резюмирането на Microsoft Word (Platt, Toutanova, and Yih 2010), системи, които обобщават съобщенията на гласовата поща за потребителите, за да им помогнат да определят приоритета на повикване (Koumpis and Renals 2005), както и системи, които могат да осигурят обобщение на потребителски форуми (Nenkova and Bagga 2003). Има приложени системи за резюмиране с цел автоматично генериране на заглавия на новинарски статии (Rush, Chopra, and Weston 2015).

2.6.1 Видове резюмиране

Различните видове резюмиране могат да бъдат класифицирани въз основа на целите, които се преследват, както и на характера на входните и изходните данни. В повечето случаи се прави резюме на един документ, въпреки това се

наблюдава резюмиране на множество от документи (Lebanoff, Song, and F. Liu 2018). Дисертационният труд представя резюмирането както на един, така и на няколко документи наведнъж.

Изходният формат на генерираното резюме може също да се различава по размер (например няколко ключови думи, заглавие, едно изречение, цял абзац или дори нов, по-кратък документ както е случаят с правните документи). Съдържанието на създаденото резюме може да бъде по-обширно или строго специфично, съдържащо само най-критичната информация. В допълнение към това могат да се добавят допълнителни ограничения, за да се генерират по-конкретни обобщения, например потребителско запитване или информация за целевата аудитория.

По отношение на формата на генерираните обобщения могат да бъдат идентифицирани два вида: екстрактивно и абстрактно (Mani and Maybury 2001) обобщение. Екстрактивният тип извлича и комбинира изречения и фрази точно както се показват в оригиналния документ. По този начин изходът е компресирана и преподредена версия на оригиналния документ. При абстрактното обобщение изходът обикновено се генерира с помощта на езиков модел, обучен върху колекция от резюмета, което допълнително зависи от входния текст. И така, обобщенията може да съдържат фрагменти, които не са част от оригиналния текст. Абстрактното резюмиране води до по-естествени, подобни на направени от човек резюмета, които могат да се позовават на концепции, които не са изрично споменати в оригиналния документ. Извличането на най-важните части от текста обаче е също толкова важен компонент, и е доказано, че комбинирането на двата подхода дава предимство в определени случаи (Rush, Chopra, and Weston 2015; Chen and Bansal 2018).

Може да се направи последно разграничение между общи и специфични за дадена област подходи за резюмиране. Общият подход има за цел да разработи единен подход за всеки тип въведен текст, докато специфичният използва наличната информация в конкретна сфера за подобряване на качеството на резюмиране. Тази информация може да е свързана със специфичния стил на писане или структурата на документа.

2.6.2 Подходи за резюмиране

Ранните изследвания на екстрактивното резюмиране се фокусират върху подреждане на изречения от оригиналния текст (Nenkova and McKeown 2011). Системите класират и избират няколко изречения от входния текст, използвайки лингвистични характеристики (например вероятността дадена дума да се среща във входния текст), които определят важността на всяко изречение. Изборът обикновено е последван от няколко стъпки за последваща обработка, които подобряват прехода между изреченията, за да се получи окончателното резюме.

След това започват да се прилагат контролирани статистически подходи за резюмиране, например системи, базирани на статистически машинен превод (Wubben, Bosch, and Krahmer 2012) или компресия на изречения (Filippova 2010). В основополагащата работа на (Rush, Chopra, and Weston 2015), те прилагат обучение от край до край като подход към абстрактно резюмиране, вдъхновен от скорошния успех на дълбоки модели за обучение на последователност за невронен машинен превод. Всички компоненти на тяхната система се обучават по начин, управляван от данни, като се използват двойки входни и изходни обобщения. Тяхната система може да генерира абстрактно заглавие на новинарска статия от първото изречение, използващо минимално количество езикова информация. Те сравняват своята система с няколко класически подхода и постигат отлични резултати върху наборите от данни Gigaword (Graff *et al.* 2003) и DUC (Schumann *et al.* 2020), които са често използвани данни за резюмиране.

2.6.3 Набори от данни за резюмиране

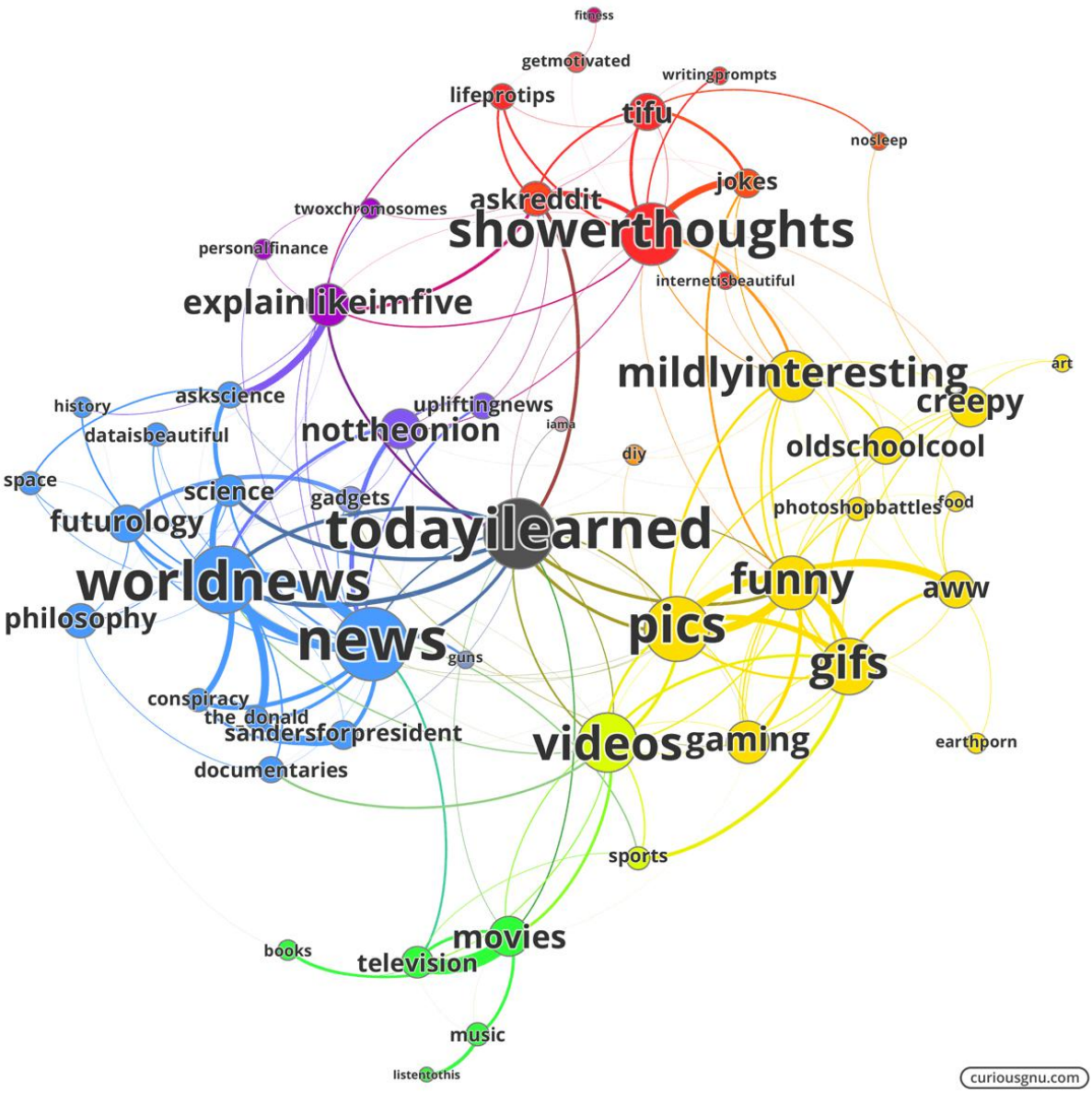
Повечето научни трудове в сферата на автоматичното резюмиране са съсредоточени върху малък брой и еднообразни набори от данни. Преобладава английският език и обикновено резюметата се състоят от едно или няколко изречения. Това дава възможност за подобряване на вече постигнати резултати, но моделите не са готови да се справят със задачата за резюмиране като цяло.

Освен това те трудно се нагаждат към конкретна област, по-малко популярен език или дълги текстове. Следващите 4 набори от данни са най-популярните към момента на писане на дисертационния труд според сайта за статии, които съдържат код[11]. Набори от данни и подходи, свързани с резюмиране на правни текстове, са разгледани в секция 2.7 – *Свързани разработки и съществуващи решения.*

Наборът от данни на Pubmed (Sen *et al.* 2008) се състои от 19717 научни публикации от базата данни на PubMed, отнасящи се до диабета, класифицирани в три класа. Мрежата за цитиране се състои от 44338 връзки. Всяка статия в набора от данни се описва от TF/IDF вектори на думи от речник с 500 уникални думи.

Reddit (Hamilton, Ying, and Leskovec 2017) е набор от данни със структура на граф от публикации в Reddit, направени през месец септември 2014г. Етикетите на ребрата в този случай е общността или "подреддит", към която принадлежи публикацията. Взети са проби от 50 големи общности, за да се изгради граф от публикация до публикация, свързваща публикации, ако един и същ потребител коментира и двете. Общо този набор от данни съдържа 232965 публикации със средна степен на свързаност 492. Първите 20 дни се използват за обучение, а останалите дни за тестване (като 30% се използват за валидиране). Визуализация на тази структура е представена на Фигура 12.

[11]Източник: https://paperswithcode.com/datasets?task=text-summarization&page=1



Фигура 12: Мрежа граф на публикации в Reddit. [12]

CNN/Daily Mail (Nallapati *et al.* 2016) е набор от данни за обобщаване на текст. Резюметата са абстрактни и са генерирани от човек. Те са от новинарски статии в уебсайтовете на CNN и Daily Mail. Общо корпусът има 286817 тренировъчни двойки, 13368 валидиращи двойки и 11 487 тестови двойки. Изходните документи в комплекта за обучение имат 766 думи, обхващащи средно 29.74

[12]Източник: https://paperswithcode.com/dataset/reddit

изречения, докато резюметата се състоят от 53 думи и 3.72 изречения.

CORD-19 (Wang *et al.* 2020) е набор от десетки хиляди научни статии за COVID-19, SARS-CoV-2 и свързаните с тях коронавируси за използване от световната изследователска общност.

2.6.4 Оценяване при автоматичното резюмиране

От първите дни на автоматичното резюмиране на текст, оценката е съществен проблем (Mani 2002). Отвореният и субективен характер на задачата прави практическата оценка на системите за обобщаване особено предизвикателен проблем. Ръчната оценка с помощта на човешка експертиза в момента е най-строгият подход към оценка на резюметата, тъй като измерва директно очакванията на потребителите в реалния живот. Ръчната оценка обаче често е много времеотнемаща и скъпа за изпълнение задача, което провокира изследователите да изобретяват автоматични показатели, които имат за цел да се приближат до човешките оценки.

По-долу обсъждаме накратко най-популярните методи за човешка и автоматична оценка на системите за резюмиране. Hardy, Narayan, and Vlachos 2019 съдържа по-задълбочено и сравняващо анализиране на широк спектър от подходи за оценка на резюмета.

Човешка оценка Ръчното оценяване на резюмета обикновено включва дефиниране на набор от въпроси, които се задават на човешките оценители. Някои въпроси може да измерват качествените аспекти на генерираните резюмета, като тяхната плавност или граматичност, докато други могат да измерват дали смисълът на оригиналния текст е запазен. Често абсолютна числова скала (Likert 1932), например между 1 и 5, се използва за такъв анализ, въпреки че базирани на предпочитания оценки, които директно карат субектите да избират между два изхода за резюмиране също са често срещани.

Автоматична оценка Добрата автоматична метрика за обобщаване трябва да взима предвид качеството на избраното съдържание от гледна точка на

информация, както и потенциално плавността му в смисъла на свързан текст. Оценяването на абстрактния текст създава допълнителни предизвикателства, тъй като изходното резюме може да съдържа думи, които не са част от входната статия. Въпреки тези предизвикателства има няколко автоматични показателя, които осигуряват бърз начин за обективно сравняване на различни подходи за резюмиране и дават известна представа за качеството на генерираните обобщения.

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) (Lin 2004) е метрика, базирана на пълнота, която обикновено се използва за обобщаваща оценка, тъй като постига добра корелация с човешки оценители. Има няколко варианта на ROUGE, като два варианта често се използват в литературата:

- **ROUGE-N** се основава на броя на съвместните срещания на N-gram (N последователни думи) между автоматично генерираното резюме и оригиналното, разделено на бройката от всички n-грамове, които се срещат в оригиналното резюме. ROUGE-1 и ROUGE-2 често се използват в литературата.

$$ROUGE_n = \frac{\sum_{gram_n \in input} count_{match}(gram_n)}{\sum_{gram_n \in input} count(gram_n)}$$

- **ROUGE-L** измерва дължината на най-дългата припокриваща се подпоследователност от думи във всяка двойка автоматично и оригинално резюме

Очевидно ограничение на ROUGE е, че изчислява силно лексикално припокриване с оригиналните резюмета. Следователно абстрактните системи могат да бъдат санкционирани, ако не използват същия речник като тях. Един от начините за облекчаване на това ограничение е да се изчисли ROUGE върху множество резюмета ако те са налични.

Bilingual Evaluation Understudy (BLEU) (Papineni *et al.* 2002) е често използван показател за автоматична оценка в системите за машинен превод.

Той е ориентиран към прецизността, като отчита процента на n-грами, които се срещат едновременно в генерирания превод и оригиналния текст. Въпреки че BLEU не се използва толкова често за оценка на резюмета, той може да бъде полезен като допълнение към ROUGE.

METEOR (Banerjee and Lavie 2005) е друга метрика, базирана на прецизност, често използвана при оценката на машинен превод. METEOR работи, като подравнява създадените от системата изречения към оригиналните, изчислявайки резултати за сходство на ниво изречение въз основа на точно съвпадение, синоним или парафраза на съвпадение на думи и фрази. Проучванията на METEOR са установили, че той корелира по-добре с човешките преводачи, отколкото BLEU. METEOR е използван в няколко скорошни проучвания за автоматично обобщаване, за да предостави различна гледна точка от ROUGE.

SARI (W. Xu *et al.* 2016) е по-нова метрика, която допълнително взема предвид общия брой промени (добавяния, изтривания) от входния текст към изхода. Въпреки че SARI първоначално е разработен за опростяване на изречения, той може също така да предостави проницателни резултати и за обобщаване на документи.

2.7 Свързани разработки и съществуващи решения

2.7.1 Автоматизирано тълкуване на правни документи чрез големи езикови модели

В статията "Automating Legal Document Interpretation with Large Language Models" (Luo *et al.* 2025) авторите представят ATRIE, иновативно решение, което използва големи езикови модели (LLMs) за автоматично извличане на информация, резюмиране на правни концепции и оценка на генерираните резюмета. Те демонстрират, че автоматичните резюмета, генерирани от ATRIE, са на същото ниво като тези, изготвени от правни експерти, с висока степен на изчерпателност и четимост.

Основни акценти от статията:

- **Методология:** ATRIE използва големи езикови модели за извличане на релевантна информация и генериране на резюмета на правни текстове.
- **Резултати:** Резултатите показват, че автоматичните резюмета на ATRIE са практически идентични с тези, изготвени от правни експерти, като значително подобряват ефективността на правния анализ и извличането на ключова информация от сложни правни текстове.
- **Предимства и недостатъци:** ATRIE предлага значителни предимства при обработка на големи обеми правни документи, но също така среща трудности при специфични и комплексни юридически концепции, които могат да изискват човешко разяснение.

2.7.2 Стандартизиране и оценка на модели за обобщаване на правни текстове

Статията "LexSumm: Benchmarking Legal Text Summarization" (Santosh, Weiss, and Grabmair 2024) представя LexSumm, комплексен модел за оценка на правни модели за резюмиране, който включва осем различни правни набори от данни, представляващи различни юрисдикции, и оценява техните ефективности чрез различни модели за резюмиране. Авторите също така представят LexT5, адаптирано решение за обобщаване на правни текстове, използващо архитектурата на T5 модела.

Основни акценти от статията:

- **LexSumm и набора от данни:** LexSumm е първият публично достъпен стандарт за правни текстове, включващ разнообразни документи от различни юрисдикции, които позволяват обективна оценка на различни алгоритми за резюмиране.
- **LexT5:** LexT5 е нов модел, специално проектиран за правни текстове. Той демонстрира обещаващи резултати при задачи за резюмиране, ка-

то същевременно показва по-добра способност да се справя с правните специфики в текстовете.

- **Оценка на резултатите:** Резултатите от оценката показват, че LexT5 превъзхожда много други методи за резюмиране, като дава по-точни и изчерпателни резюмета за правни документи, като същевременно демонстрира висока производителност при многоезични задачи.

2.7.3 Голям набор от съдебни решения на Върховния съд на САЩ за автоматично резюмиране

В статията "CaseSumm: A Dataset for Legal Document Summarization from US Supreme Court Cases" (Heddaya *et al.* 2024), авторите въвеждат CaseSumm, набор от данни с 25.6 хиляди решения на Върховния съд на САЩ и техните официални обобщения. Този набор от данни е най-големият публично достъпен ресурс за обобщаване на правни документи, съдържащ решения от Върховния съд и техните съответстващи резюмета. Статията също така представя нови методи за оценка на автоматични резюмета в правния контекст.

Основни акценти от статията:

- **CaseSumm набор от данни:** Наборът включва 25.6 хиляди решения на Върховния съд на САЩ и техните обобщения, които са изключително ценни за обучения и тестове на модели за автоматично резюмиране.
- **Оценка на резултатите:** Генерираните резюмета са оценявани както автоматично, така и чрез човешка оценка, като резултатите показват, че моделите с висок капацитет за обработка на правни текстове превъзхождат други методи за автоматично генериране на резюмета.
- **Бъдещи насоки:** Статията предлага идеи за бъдещи изследвания, включително за подобряване на моделите чрез обучение върху още по-големи и разнообразни правни набори от данни.

2.7.4 Многоезично резюмиране на съдебни решения от швейцарската съдебна система

В статията "Unlocking Legal Knowledge: Swiss Leading Decision Summarization" (Rolshoven *et al.* 2024), авторите представят нов набор от данни за резюмиране на съдебни решения от Швейцарския федерален съд, известен като Swiss Leading Decision Summarization (SLDS). Наборът съдържа 18 хиляди съдебни решения на три различни езика (немски, френски и италиански), а резюметата са предоставени на немски език.

Основни акценти от статията:

- **SLDS набор от данни:** Това е многоезичен набор от данни, който включва съдебни решения от различни правни юрисдикции в Швейцария, както и обобщения на тези решения на три езика.
- **Оценка на модели:** Моделите за обобщаване са тествани върху SLDS, като резултатите показват, че по-малките модели могат да предоставят по-добри резултати в правните контексти, отколкото по-големите.
- **Бъдещи насоки:** Статията предлага бъдещи изследвания в областта на многоезичното обобщаване, както и важността на локализирането на правните контексти за по-точни резюмета.

Тези статии показват текущото състояние и бъдещите насоки в областта на автоматизираното резюмиране на правни текстове. Разглеждат се нови модели и набори от данни, които предлагат значителни възможности за напредък в автоматичното генериране на правни резюмета и подобряване на ефективността в правния сектор чрез изкуствен интелект.

3 Нов корпус от данни за правни текстове и техните резюмета

3.1 Въведение

Като част от дисертацията бе избран конкретен проблем с юридически текстове. Освен изготвянето на нов набор от данни[13], който да послужи и на други изследователи, тези данни бяха използвани за различни експерименти с автоматичното резюмиране.

Данните са взети от официалния сайт на Европейския съюз[14]. Те съдържат юридически текстове на различни теми, включително закони, актове и други. Всеки документ има резюме, генерирано от експерт в областта. Всички документи и резюмета, освен на английски език, са преведени на до 23 други езика. Има резюмета, които резюмират повече от 1 документ. Към момента на взимане на данните има 1776 резюмета и съответстващите им пълни документи на български език. Сравнение между данните на различни езици е направено в следващите секции.

Има много характеристики на набора от данни, които го правят ценен за изследователската сфера. Той съдържа текст на специфична тематика, което е предизвикателство за претренираните модели на общ текст. Текстовете са с различна дължина, като някои документи са изключително дълги. Наличието им на различни езици предоставя възможност за сравнение на резултатите и валидация дали експериментите работят само в частен случай. Малкият брой на документите допълнително затруднява задачата.

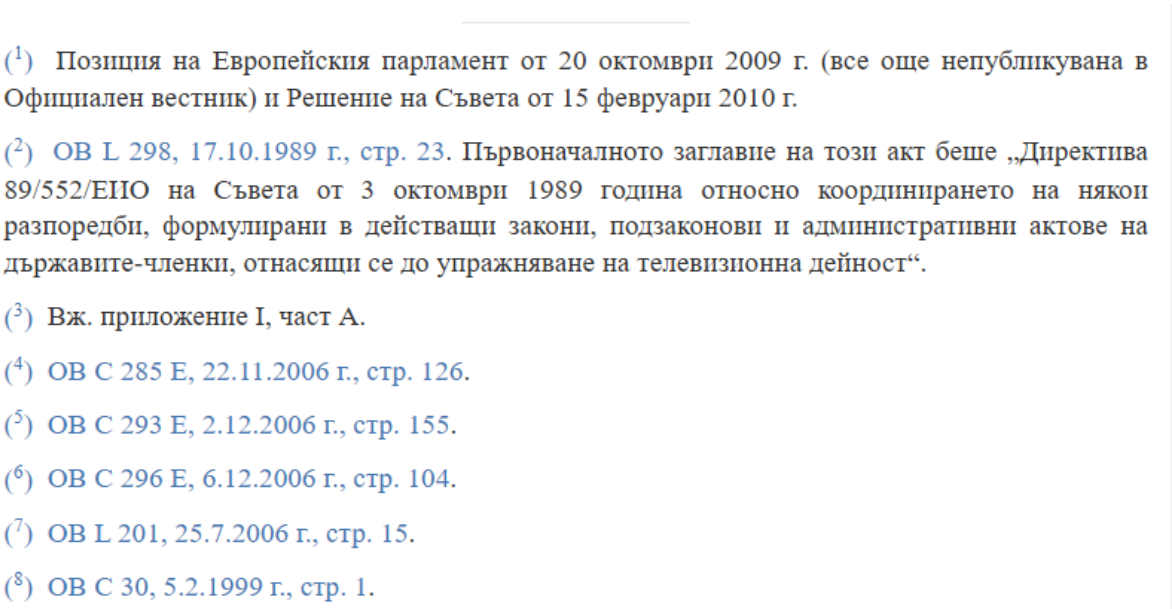
[13]https://huggingface.co/datasets/FMISummarization/FMI_Summarization
[14]https://eur-lex.europa.eu/homepage.html

3.2 Събиране и изчистване на данните

За всеки един документ и резюме се изтегля пълното HTML съдържание. Всички документи се обхождат като се взимат резултатите от търсенето страница по страница и за всеки един резултат се запазват уникалните идентификатори на пълните документи и резюметата. В случаите, в които има няма превод на търсения език за съответната двойка резюме, документи информацията не се запазва.

Премахват се заглавната част (Фигура 13) и списък с референции от найдолната част (Фигура 14). Има случаи, в които няколко документа съответстват на един линк. При тях посочените действия се повтарят за всеки един документ и съдържанието се конкатенира.

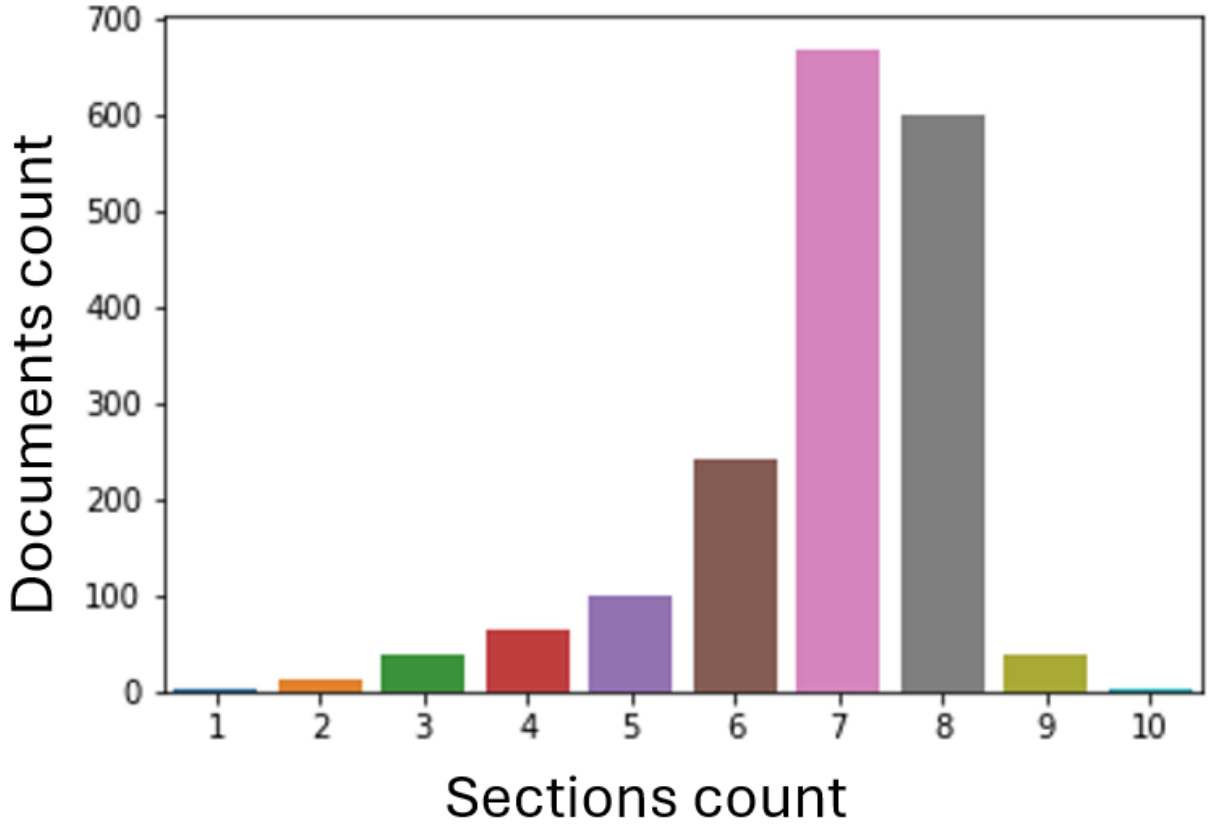
Фигура 13: Заглавна част на документ.



Фигура 14: Референции на документ.

Резюметата са разделени на секции. Фигура 15 показва броя документи с

определен брой секции. Най-много са документите с 6,7 и 8 секции. След подробен анализ на секциите, се оказва, че голяма част от тях не носят същинска информация за резюмето. Идентифицирани са 408 различни заглавия на секции. Таблица 1 показва най-популярните 10 имена на секции и броя документи, в които се срещат.

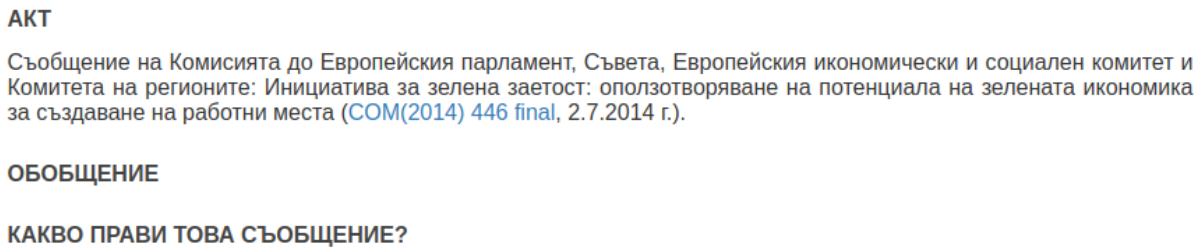


Фигура 15: Брой документи с определен брой секции. Фигурата показва, че най-голямата част от документите съдържат 6,7 или 8 секции.

За резюме считаме текстът, който се съдържа в една от следните секции: ОСНОВНИ АСПЕКТИ, ОБОБЩЕНИЕ. Има резюмета, които не съдържат нито една от тези секции. След ръчен преглед се оказа, че те наистина са невалидни (пример Фигура 16). Цялата съществена информация се съдържа в тези секции, докато останалите съдържат съпътстваща информация, която не е от ключово значение за резюмето или вече е описана в основната секция.

Други документи, които бяха премахнати от набора данни са такива, при

които няма превод на съответния език. Освен това има документи, при които текстът на резюмето е по-дълъг от пълния документ. След ръчна проверка отново се оказаха невалидни документи, които бяха премахнати.


АКТ

Съобщение на Комисията до Европейския парламент, Съвета, Европейския икономически и социален комитет и Комитета на регионите: Инициатива за зелена заетост: оползотворяване на потенциала на зелената икономика за създаване на работни места (COM(2014) 446 final, 2.7.2014 г.).

ОБОБЩЕНИЕ

КАКВО ПРАВИ ТОВА СЪОБЩЕНИЕ?

Фигура 16: Резюме с липсващ текст.

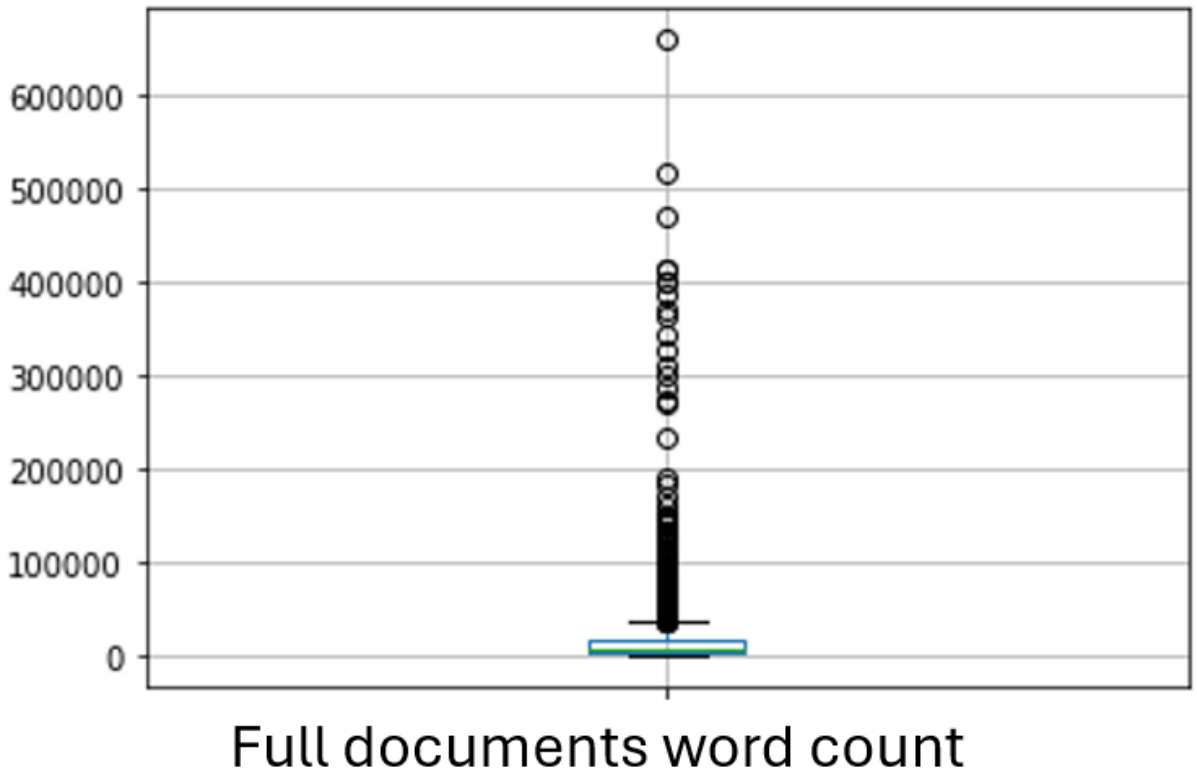
Име на секция	**Брой документи, в които се среща**
ОСНОВНИ АСПЕКТИ	1598
ОБЩА ИНФОРМАЦИЯ	1513
РЕЗЮМЕ НА:	1508
СВЪРЗАНИ ДОКУМЕНТИ	1013
ОСНОВЕН ДОКУМЕНТ	986
ОСНОВНИ ПОНЯТИЯ	682
ОТКОГА СЕ ПРИЛАГА РЕГЛАМЕНТЪТ?	525
КАКВА Е ЦЕЛТА НА РЕГЛАМЕНТА?	435
ОБОБЩЕНИЕ	405
АКТ	402

Таблица 1: Имена на секции и брой документи, в които се срещат (10 най-популярни).

3.3 Данни на български език

Крайният резултат е 1776 резюмета със съответните им документи. Средната дължина на пълните документи е 21152 думи, а на резюметата - 543 думи. Съотношението в броя думи между пълните документи и резюметата е сред-

но 45.7. Важно е да се спомене, че дължината на текстовете е неравномерно разпределена с наличието на много отклонения. Например има резюмета с дължина над 5000 думи (10 пъти над средното) и пълни текстове с дължина над 600000 думи (30 пъти над средното) (Фигура 17).

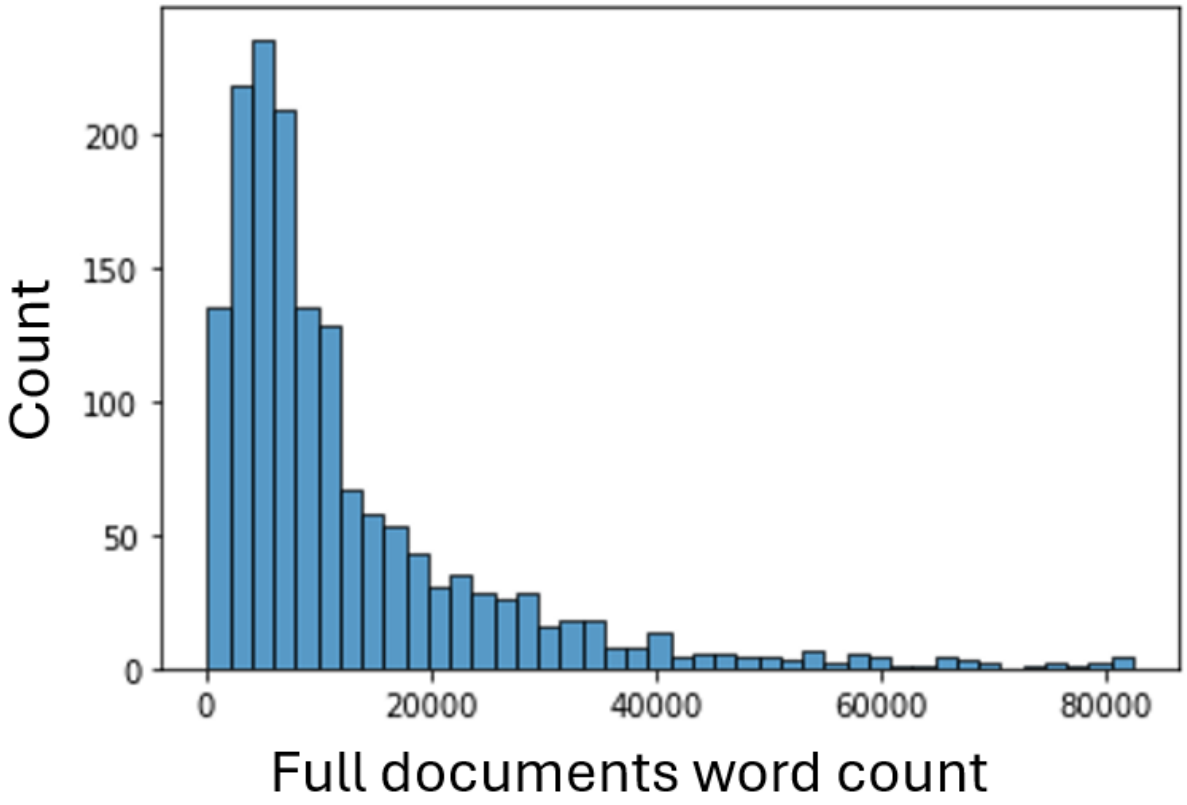


Фигура 17: Брой думи в пълните документи. Наличието на отклонения с над 100000 думи изкривяват статистиката за среден брой думи.

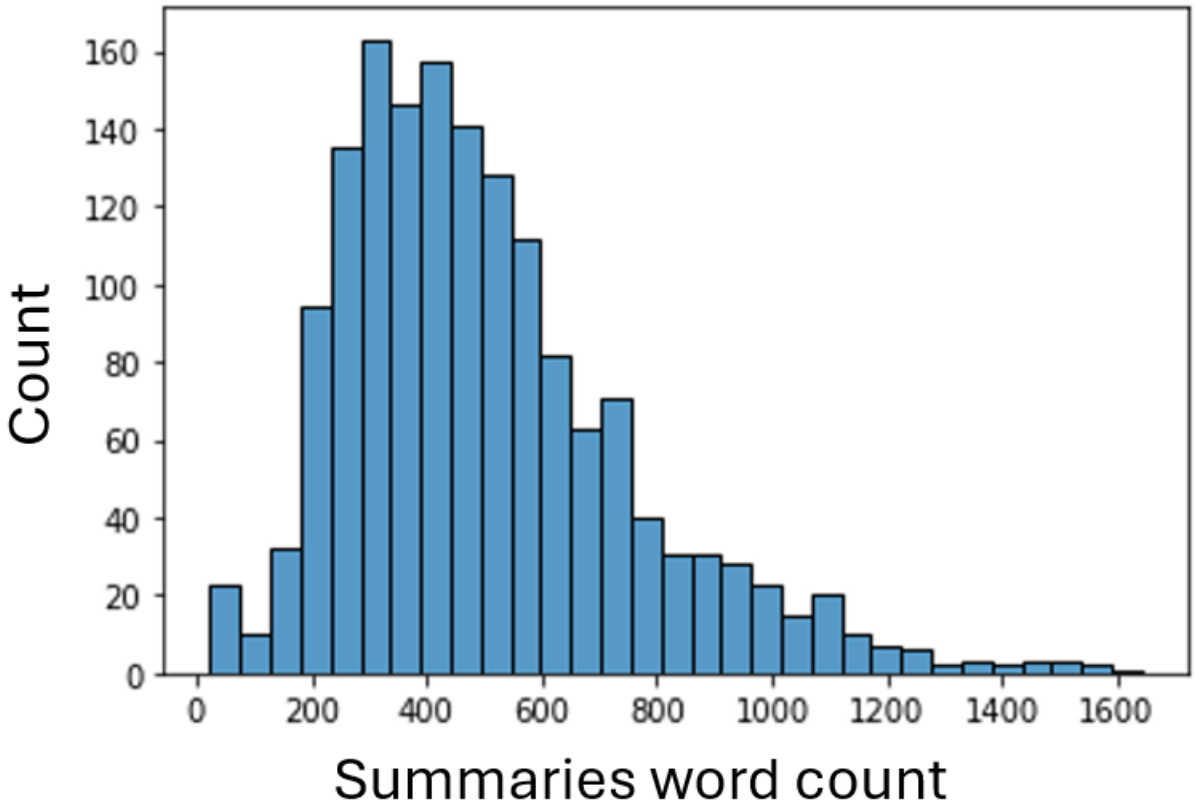
С цел по-добра визуализация и придобиване на по-добра представа за разпределението на данните предоставените фигури са с изчистени отклонения. Само за целите на диаграмите са премахнати:

- Пълни документи с повече от 84000 думи.
- Резюмета с повече от 2000 думи.
- Съотношение на думите по-голямо от 250.

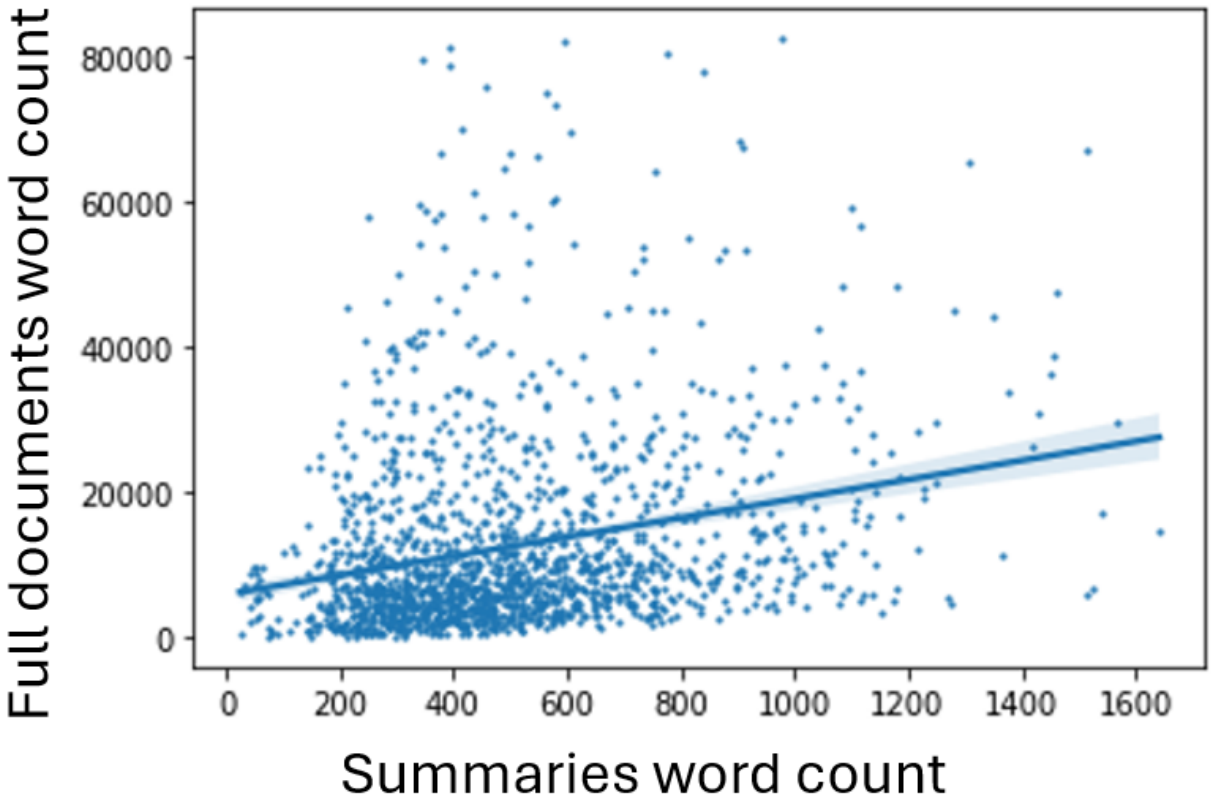
Както се вижда на Фигура 20 съотношението на броя думи между пълните документи и резюметата не е фиксирано и варира много в различните примери. Това прави задачата за автоматично резюмиране още по-предизвикателна. Фигура 18 и Фигура 19 допълнително илюстрират разпределението на броя думи в пълните документи и резюметата след премахване на значителните отклонения.



Фигура 18: Разпределение на броя думи на пълните документи след премахване на отклоненията.



Фигура 19: Разпределение на броя думи на резюметата след премахване на отклоненията.



Фигура 20: Всяка точка представлява данните за броя думи на едно резюме и съответния брой думи на пълния документ след премахване на отклоненията. Линията и пространството около нея показва модел на линейна регресия, който се доближава най-много до съотношението в броя думи между резюметата и пълните документи.

Тъй като правните документи могат да бъдат от различно естество, направихме категоризация на пълните документи. Тя е на база на първата дума в текста. Категоризацията не се използва по време на генериране на резюметата, а е по-скоро за добиване на по-добра представа за естеството на данните. Таблица 2 показва разпределението на категориите.

Категория	Брой документи
РЕГЛАМЕНТ	614
ДИРЕКТИВА	389
РЕШЕНИЕ	229
СЪОБЩЕНИЕ	132
ДРУГИ/НЕДЕФИНИРАНИ	109
СПОРАЗУМЕНИЕ	49
ЧЛЕН	40
ПРЕПОРЪКА	37
ЗАКЛЮЧЕНИЯ	29
КОНВЕНЦИЯ	15
ИЗВЕСТИЕ	14

Таблица 2: Брой документи по категории на база на първата дума в текста.

3.4 Данни на други езици

Използвайки същата методология бяха изтеглени данните на всички 24 налични езика. Таблица 3 показва разликите между данните на различните езици. Броят на документите на различните езици варира, като очаквано на английски има най-много (1854), а на ирландски най-малко (518). Съотношението в броя думи между пълните документи и резюметата е сравнително постоянно при всички езици с леки вариации, които се дължат на спецификите на езика и на документите, които имат превод.

За всеки език бяха идентифицирани ключовите секции, произхождайки от имената на английски и български и анализирайки преводите на съответните езици и брой документи със секциите.

3.5 Публикуване на набора от данни

За да бъде максимално полезен за научната общност, разработеният набор от данни беше публикуван в платформата Hugging Face [15]. Публикуването на научни ресурси в отворен достъп е от съществено значение за напредъка на изследванията, тъй като позволява на други специалисти лесен достъп до данни, необходими за развитие и тестване на модели. Наличието на такъв набор от данни в публично достъпна платформа насърчава сътрудничество, споделяне на знания и повторна употреба на данните за различни експериментални цели. Това осигурява възможност на изследователите в сферата на автоматичното резюмиране да сравняват своите резултати и да подобряват методите си въз основа на унифициран набор от данни.

Чрез публикуването на този набор от данни се поставя акцент върху значението на прозрачността и достъпността в научните изследвания. Той може да служи като основа за бъдещи проучвания, които да надграждат постигнатото, и значително улеснява експериментите на други изследователи, особено когато става дума за специфични домейни като правната сфера, където данните обикновено са труднодостъпни и сложни за обработка.

[15] https://huggingface.co/datasets/FMISummarization/FMI_Summarization

Код	Език	Брой документи	Брой думи в пълните документи (средно)	Брой думи в резюметата (средно)	Секции за резюмиране
BG	Български	1776	21152	543	ОСНОВНИ АСПЕКТИ ОБОБЩЕНИЕ
CS	Чешки	1793	19492	474	KLÍČOVÉ BODY PŘEHLED
DA	Датски	1838	19766	475	HOVEDPUNKTER RESUMÉ
DE	Немски	1844	19886	509	WICHTIGE ECKPUNKTE ZUSAMMENFASSUNG
EL	Гръцки	1732	22557	565	ΒΑΣΙΚΑ ΣΗΜΕΙΑ ΣΥΝΟΨΗ
EN	Английски	1854	21872	509	KEY POINTS SUMMARY
ES	Испански	1842	23711	608	PUNTOS CLAVE SÍNTESIS
ET	Естонски	1796	16700	393	PÕHIPUNKTID KOKKUVÕTE
FI	Финландски	1830	16323	390	TÄRKEIMMÄT KOHDAT TIIVISTELMÄ
FR	Френски	1833	23884	663	POINTS CLÉS SYNTHÈSE
GA	Ирландски	518	28014	680	PRÍOMHPHOINTÍ ACHOIMRE
HR	Хърватски	1657	17085	468	KLJUČNE TOČKE SAŽETAK
HU	Унгарски	1795	19330	480	FŐBB PONTOK ÖSSZEFOGLALÓ
IT	Италиански	1836	21860	589	PUNTI CHIAVE SINTESI
LT	Литовски	1796	18555	454	PAGRINDINIAI ASPEKTAI SANTRAUKA

LV	Латвийски	1796	18642	457	SVARĪGĀKIE ASPEKTI KOPSAVILKUMS
MT	Малтийски	1765	19371	488	PUNTI EWLENIN SOMMARJU
NL	Нидерландски	1832	21433	541	KERNPUNTEN SAMENVATTING
PL	Полски	1797	19366	490	KLUCZOWE ZAGADNIENIA STRESZCZENIE
PT	Португалски	1840	21887	577	PONTOS-CHAVE SÍNTESE
RO	Румънски	1789	21676	547	ASPECTE-CHEIE SINTEZĂ
SK	Словашки	1788	18964	476	HLAVNÉ BODY SÚHRN
SL	Словенски	1797	19203	478	KLJUČNE TOČKE POVZETEK
SV	Шведски	1827	18014	470	VIKTIGA PUNKTER SAMMANFATTNING

Таблица 3: Сравнение между данните на всички езици

4 Разработени и адаптирани подходи за резюмиране на текстове

4.1 Базов подход (избиране на първите изречения в текста)

При всички експерименти в света на изкуствения интелект се избира базов подход, спрямо който се сравняват постигнатите резултати. Целта на експериментите е естествено да го подобрят. Базовият подход трябва да бъде лесен за имплементиране, бърз от гледна точка на брой изчисления и въпреки това да е смислен за конкретната задача.

За целите на автоматичното резюмиране ние избрахме като базов подход взимане на първите изречения в текста. Важно уточнение е, че по време на експериментите ние се стремим да генерираме резюмета със същата дължина както генерираните от експертите. Причина затова е големите разлики в големините на резюметата и в съотношенията между броя думи в пълните текстове и резюметата. Това също така позволява метриките прецизност и пълнота да са относително еднакви и полученият F1 резултат да е достатъчно показателен.

Дефинираме подходът по следния начин: взимаме последователно изречения от пълния текст, започвайки от началото. Спираме когато броят думи в изреченията е по-голям или равен на броя думи в оригиналното резюме.

Подходът освен изключително лесен за имплементиране и бърз за смятане е и много подходящ за конкретния проблем. Обикновено най-съществената информация в даден текст се намира в самото му начало и този подход се използва много често в сферата. Целта пред всички останали подходи е да подобрят резултатите на базовия подход.

4.2 Екстрактивен подход основан на TF-IDF

Първият подход, който цели да подобри базовият се базира на TF-IDF и не изисква обучение. TF-IDF (Term Frequency–Inverse Document Frequency) е цифрова информация, която има за цел да отрази колко важна е дадена дума за документ в колекция или корпус. Често се използва като тегловен фактор при търсения за извличане на информация. Стойността TF-IDF се увеличава пропорционално на броя пъти, в които дадена дума се появява в документа и се компенсира от броя на документите в корпуса, които съдържат думата, което помага да се коригира фактът, че някои думи се появяват по-често като цяло. TF-IDF е един от най-популярните подходи за претегляне на значимостта на дума. Проучване, проведено през 2015г., показва, че 83% от текстово базираните препоръчителни системи в цифровите библиотеки използват tf–idf (Beel *et al.* 2015).

В нашия случай използваме TF-IDF на ниво изречение в текста спрямо всичко останали изречения. Подходът цели да идентифицира изречения с думи, които са ключови за текста и се срещат често в него. След това конкатенирането на тези изречения е генерираното резюме.

Алгоритъмът се изпълнява за всеки пълен текст и е изцяло независим от генерираното резюме. Първата стъпка е всички думи да се приведат в основната си форма (stemming). В лингвистичната морфология и извличане на информация свеждането на дума до основната си форма (stemming) е процесът на редуциране на склонени (или понякога производни) думи до тяхната основна или коренна форма. Основата не трябва да е идентична с морфологичния корен на думата; обикновено е достатъчно свързаните думи да се съпоставят с една и съща основа, дори ако тази основа сама по себе си не е валиден корен. Алгоритмите за stemming се изучават в компютърните науки от 60-те години на миналия век. Процесът е специфичен за всеки език, заради различните окончания и формулиране на думите. Nakov 2003 представя стемер на български език, който е използван в дисертационния труд.

Следващата стъпка е премахване на стопиращи думи (stop words). Те пред-

ставляват набор от често използвани думи за конкретен език, които сами не носят смисъл, а се използват за добавяне на контекст към други ключови думи. Премахването им за целта на TF-IDF оценката ни позволява да се съсредоточим върху ключовите думи в текста. Стопиращи думи на български са “си”, “го”, “ако”, “и” и други подобни. Те отново са специфични за всеки език и за разлика от stemming, тук логиката е много проста - списък от думи за всеки език, които се изключват от текста.

След като сме привели всички думи в основна форма и сме премахнали стопиращите думи изчисляваме за всяка дума колко пъти се среща в текста. След това разделяме намираме думата, която се среща най-много пъти и разделяме всички стойности на това число. Това позволява всички стойности да са в диапазона (0-1].

За финалната стъпка разделяме документа на изречения. За всяко изречение определяме резултатът му като съберем TF-IDF за всяка от думите му. Накрая сортираме изреченията по получените резултати в низходящ ред и конкатенираме изреченията поред докато не достигнем дължината на оригиналното резюме.

4.3 Екстрактивен подход TextRank

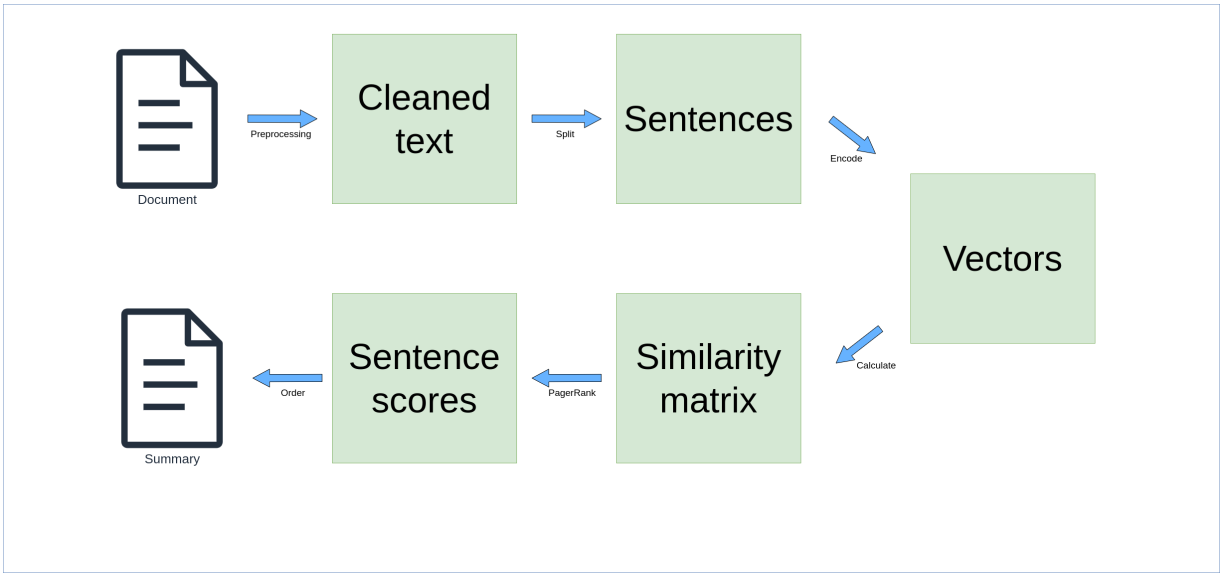
Преди да навлезем в детайли как работи TextRank (Mihalcea and Tarau 2004) алгоритъма, трябва да обясним PageRank (Page *et al.* 1999) алгоритъма, на който той е базиран. PageRank се използва предимно за подреждане на уеб страници в резултатите от онлайн търсене. Да предположим, че имаме 4 уеб страници — w1, w2, w3 и w4. Тези страници съдържат връзки, сочещи една към друга. Някои страници може да нямат връзка – те се наричат висящи страници.

Уеб страница	Връзки
w1	[w4, w2]
w2	[w3, w1]
w3	[]
w4	[w1]

Таблица 4: Връзки между примерните уеб страници

Таблица 4 показва връзките. w1 има връзки към w2 и w4. w2 има връзки към w3 и w1. w4 има връзка само към w1, а w3 няма връзки към други страници. PageRank резултатът се формира под формата на матрица и е вероятността дадена страница да бъде посетена от друга.

За целите на TextRank алгоритъма и автоматичното резюмиране вместо уеб страници използваме изречения. Сходност между 2 всеки две изречения в текста се използва вместо вероятността за отиване на страница. Стойностите на сходност се пазят в матрица с еднакъв брой редове и колони равни на броя на изреченията в текста. TextRank е екстрактивен подход за резюмиране, който няма нужда от трениране спрямо генерирано резюме от експерт.



Фигура 21: Начин на работа на алгоритъма TextRank

Фигура 21 показва как работи алгоритъма. Също както предходния алгоритъм имаме стъпка на предварителна обработка. Тя включва премахването на стопиращи думи и привеждането на всички думи в основната им форма (stemming). След това разделяме текста на изречения.

За всяко изречение изготвяме вектор на вграждания с помощта на Word2Vec (Mikolov *et al.* 2013b). Word2Vec е съвременен алгоритъм за генериране на разпределено векторно представяне с фиксирана дължина на всички думи в огромен корпус. Ефективността на Word2Vec се дължи на две причини — едната е използването на вектори с фиксиран размер, което означава, че размерът на вектора не зависи от броя на уникалните думи в корпуса. Второ, включване на семантична информация във векторните представяния. Векторите Word2Vec са много ефективни при групирането на подобни думи заедно. Алгоритъмът може да прави силни оценки въз основа на позицията на думата в корпуса. Например "красив" и "хубав" са подобни и следователно тяхното векторно представяне ще бъде много подобно. Получените вектори ни позволяват да представим всяко изречение като съвкупност от вектори за всяка дума в тях. С цел да получим вектори с еднакви размери за всички изречения, допълваме с нули всички изречения, освен най-дългото.

След като имаме вектори, които представляват всяко изречение изготвяме матрица на сходствата, която е квадратна с брой колони и редове равни на броя на изреченията в текста. Върху нея прилагаме PageRank алгоритъма, който ни дава оценка за всяко изречение.

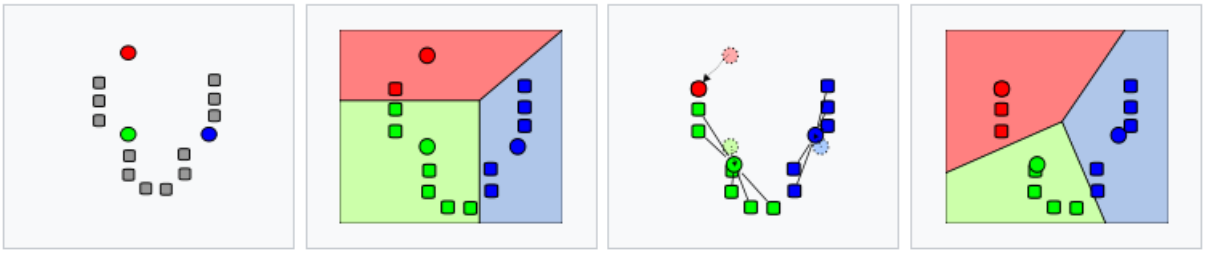
След като имаме оценките, подобно на предишния алгоритъм, сортираме изреченията по получените резултати в низходящ ред и конкатенираме изреченията поред докато не достигнем дължината на оригиналното резюме.

4.4 Екстрактивен подход основан на BERT и K-means

Следващият изпробван подход използва BERT (Devlin *et al.* 2018) като основа за изготвяне на вграждания за изреченията и алгоритъмът за групи-

ране K-means.

K-means е метод за векторно клъстериране, при който се търси "среда" на всяка една група (наричана още център/центроид на клъстера) и всяко наблюдение се причислява към най-близкия до него център. Алгоритъмът е много популярен с широко приложение, първоначално започнат с идеята за използване за обработка на сигнали. При K-means броят клъстери е предварително зададен (k). Намирането на центровете на тези клъстери става чрез итеративен подход (виж Фигура 22).



Фигура 22: Начин на работа на алгоритъма K-means [16]

Той се състои от следните стъпки:

1. Първоначален избор на центрове на клъстерите.
2. Причисляваме всяка точка от данните към най-близкия център.
3. Обновяваме центъра да е в "средата" на всички точки, причислени към него.
4. Повтаряме 2. и 3. докато центърът спре да се движи.

Алгоритъмът е с трудност NP. Има различни стратегии за първоначален избор на центровете, като избор на произволни точки от пространството, избор на някои конкретни данни да служат като център и други. Алгоритъмът може да попадне на локален минимум и да не намери оптималното решение. Това зависи от първоначалният избор на центрове.

[16]Източник: https://en.wikipedia.org/wiki/K-means_clustering

Също както при всички екстрактивни подходи до момента, тук първо изчистваме текста и го разделяме на изречения. С помощта на BERT генерираме вектори за всички изречения.

Като следваща стъпка прилагаме K-means алгоритъма върху генерираните вграждания. Идеята е да се групират изреченията, които са контекстуално подобни едно на друго, и да се избере по едно изречение от всеки клъстер, което е най-близко до средната стойност (центроида).

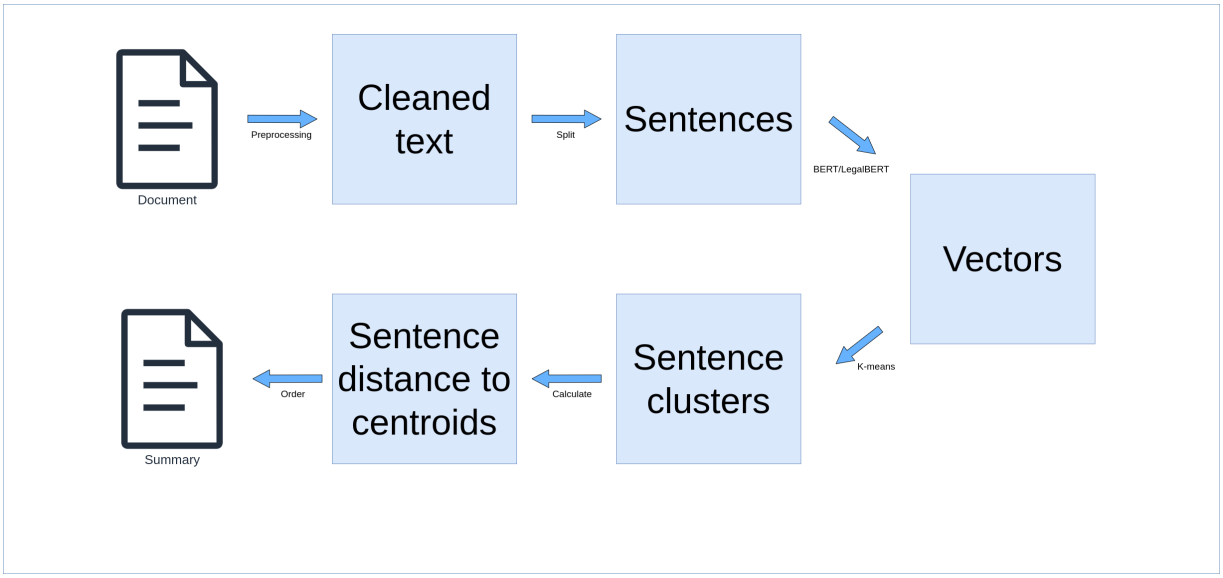
След това за всеки вектор на изречение изчисляваме разстоянието от центроида. Понякога центроидите са действителните изречения и в този случай разстоянието е нула. От всеки клъстър избираме едно или няколко изречения, които са най-близки до центроида (Фигура 23).

Заради ограничения на BERT, този подход е изпробван само на английски език.

4.5 Екстрактивен подход основан на LegalBERT и K-means

Като следваща стъпка повторихме предните експерименти с една промяна. За генериране на вектори (вграждания на думи) вместо BERT, използвахме LegalBERT (Chalkidis *et al.* 2020).

LegalBERT е семейство модели на BERT за правната област, предназначени да подпомагат изследвания за обработка на текст в правната сфера, изчислителното право и приложенията на правни технологии. За да се обучат предварително различните варианти на LegalBERT са събрани 12 GB разнообразен английски правен текст от няколко области (например законодателство, съдебни дела, договори), извлечени от публично достъпни ресурси. LegalBERT се представя по-добре от използването на стандартния BERT модел за задачи, специфични за областта. Предлага се и малък модел (33% от размера на BERT-BASE), предварително обучен от нулата върху правни данни.

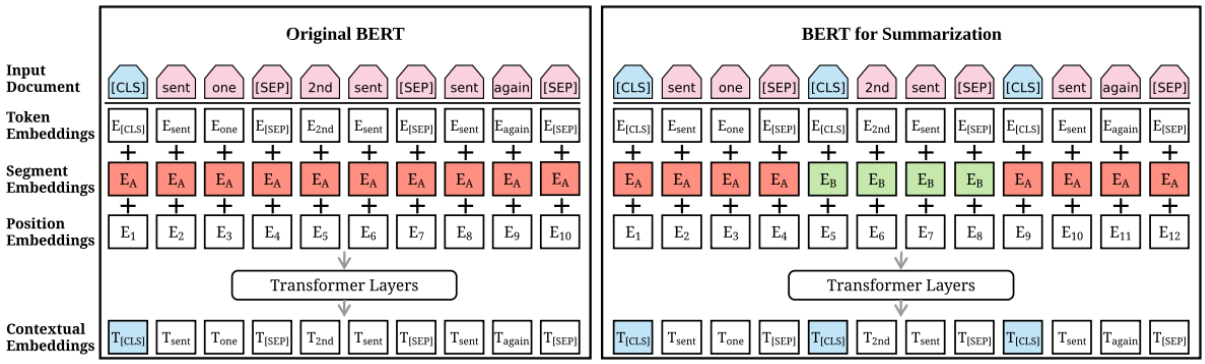


Фигура 23: Начин на резюмиране с BERT и K-means

4.6 Модел за резюмиране PreSumm

Към момента на писане на дисертационния труд, най-добри резултати в сферата върху редица известни набори от данни постига алгоритъмът PreSumm (Y. Liu and Lapata 2019). Авторите предлагат нов подход за обучение, който разделя оптимизаторите на енкодера и декодера, за да се съобрази с факта, че енкодерът е предварително обучен, докато декодерът трябва да бъде обучен от нулата. Използва се BERT за превръщане на текста във вектори и се предлага нов независим подход за превръщане на векторите в резюме.

Фигура 24 сравнява стандартната употреба на BERT и използването му за автоматично резюмиране. Първият ред е входният документ, последван от сумирането на три вида вграждания за всеки токен. Сумираните вектори се използват като входни вграждания към няколко двупосочни трансформаторни слоя, генерирайки контекстуални вектори за всеки токен. BERTSUM разширява BERT чрез вмъкване на множество символи $[CLS]$ за изучаване на представяния на изречения и използване на вграждания за сегментиране на интервали (илюстрирани в червен и зелен цвят) за разграничаване на множество изречения.



Фигура 24: Сравнение на стандартна употреба на BERT и употребата му за резюмиране [17]

За целите на екстрактивно резюмиране, авторите използват вгражданията, генерирани от BertSum (архитектурата показана на Фигура 24). След това те използват допълнително вкараните от тях [CLS] токени, чрез които да разграничат изреченията. Използват невронна мрежа, чиято цел е за всяко изречение да предвиди дали да бъде част от резюмето или не.

Авторите използват стандартна архитектура за енкодер-декодер за абстрактно резюмиране. Енкодерът е предварително обучен BERTSUM, а декодерът е 6-слоен модел с архитектура на трансформатор, инициализиран на случаен принцип. Възможно е да има несъответствие между енкодера и декодера, тъй като енкодерът е предварително обучен, докато декодерът трябва да бъде обучен от нулата. Това може да направи претренирането нестабилно. Например, енкодерът може да надхвърли данните, докато декодерът да не пасне, или обратното. За да се заобиколи това, авторите проектират нова възможност за фина настройка, която разделя оптимизаторите на енкодера и декодера.

Освен това има възможност за двуетапен подход, при който първо се тренира екстрактивен модел и след това се прилага абстрактивен подход. Забележете също, че този двуетапен подход е концептуално много прост, моделът може да се възползва от информацията, споделена между тези две задачи,

[17] Източник: https://arxiv.org/pdf/1908.08345.pdf

без фундаментално да променя своята архитектура. Абстрактивният модел по подразбиране се нарича BERTSUMABS, а двустепенния настроен модел BERTSUMEXTABS.

За целите на експерименти с правни текстове, претренирахме моделът върху новите данни. Кодът от статията трябваше да бъде сериозно променен, за да се адаптира за нуждите. Основно предизвикателство представляваха подългите текстове както за пълните текстове така и за резюметата, поради които текстът се подава на парчета с големина най-много 512 токена.

Заради ограничения на BERT, този подход е изпробван само на английски език.

4.7 Езиков модел T5

Моделът T5 (Text-To-Text Transfer Transformer) е един от най-мощните абстрактивни модели за генериране и резюмиране на текст. В този проект T5 е използван в стандартната си версия, предоставена от библиотеката Hugging Face, в "base" размер [18]. Важно предимство на T5 е неговата гъвкавост, която позволява допълнително претрениране на модела, за да бъде адаптиран към специфични задачи, като резюмиране на правни текстове.

Първоначално са направени опити за фино настройване на T5 върху специфични правни данни с цел подобряване на неговата ефективност при работа с дълги правни документи. При тези експерименти обаче не се наблюдават значителни подобрения в резултатите в сравнение с готовия модел. Причините за липсата на подобрение могат да се дължат на ограниченията на модела, свързани с обработката на по-дълги текстове. Моделът T5 е оптимизиран за обобщаване на документи с ограничена дължина на входа, което ограничава директната му приложимост за по-големи документи, каквито са правните текстове.

Също така, експериментите с T5 са проведени единствено на английски

[18]Източник: https://huggingface.co/docs/transformers/model_doc/t5

език поради липсата на мултиезикови версии на модела, способни да обработват документи с по-голяма сложност и обем. Въпреки тези ограничения, моделът T5 демонстрира конкурентни резултати в резюмирането на текстове, които попадат в неговите параметрични ограничения.

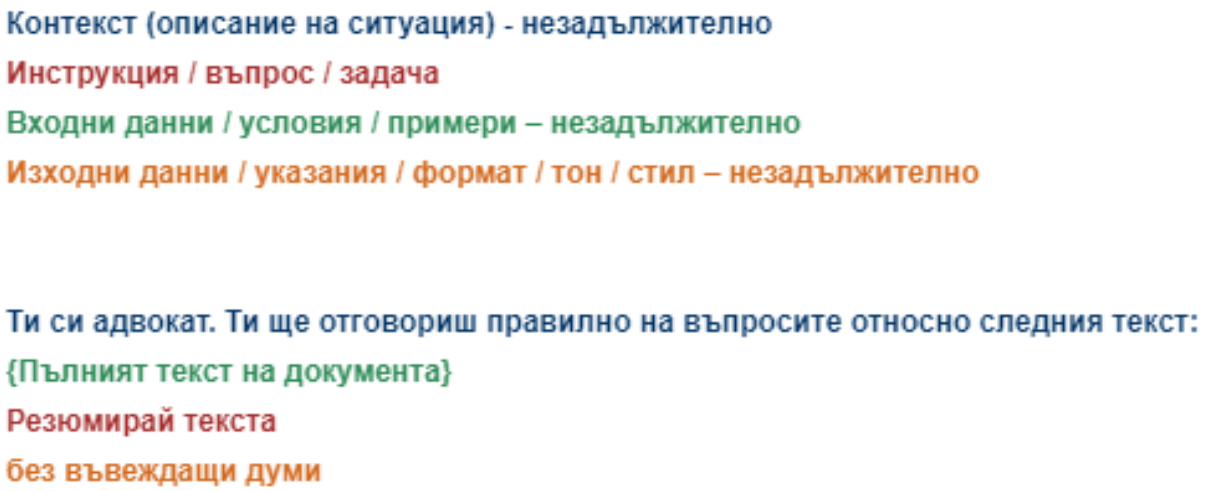
4.8 Голям езиков модел GPT

Използвахме най-новите постижения в областта на машинното обучение за генериране на резюмета, като използвахме GPT API на OpenAI.

Експериментирахме с различни промптове. Окончателният системен промпт, който използвахме, е "Ти си адвокат." Целта е да насочим агента да говори сякаш е адвокат, което съответства на профила на хората, създали резюметата. Промптът за документа е "Ти ще отговориш правилно на въпросите относно следния текст:", последван от пълния текст. Промптът за финалната задача за GPT API е "Резюмирай текста без въвеждащи думи." Причината да добавим завършек като "без въвеждащи думи" е следната: GPT API има ограничения за максималния брой токени, които не са достатъчни за повечето текстове. Поради това се наложи текстът да бъде разделен на части. Без това уточнение в промпта, всички генерирани текстове започват с въведение като "Текстът очертава" или "Най-важните части от текста са". Тези фрази се повтарят за всяка част, а такива въведения не присъстват в истинските резюмета. Промптовете са преведени автоматично на 24 различни езика.

Структурата на промптовете обикновено включва няколко основни компонента, които могат да бъдат адаптирани според конкретната задача (Фигура 25). Един от тези компоненти е контекстът, който предоставя допълнителна информация за ситуацията или ролята, в която трябва да влезе моделът, като цяло не е задължителен, но може да помогне за насочване на отговора в правилната посока. Следва инструкцията, въпросът или задачата, които ясно формулират какво се очаква от модела. Промптът може също така да съдържа входни данни, условия или примери, които да осигурят необходимата информация за изпълнение на задачата, както и указания относно изхода, които

уточняват желаните формат, тон и стил на отговора. Комбинирането на тези елементи позволява създаването на ефективни и целенасочени промптове, които водят до по-прецизни и уместни резултати.



Фигура 25: Структура на промпт

Цената за използването на GPT API зависи от броя на използваните токени. Към момента на писане на този труд, цената за модела е $0.0015 за 1000 токена за вход и $0.002 за 1000 токена за изход. Изчисляването на всички резюмета за всички езици би струвало $600 с този модел и $14,000 за модела GPT-4. Заради ценовите съображения използвахме модела GPT-3.5-turbo и генерирахме резюмета за 10% от текстовете на всички езици, което ни струваше около $60 с допълнителни разходи от около $40 за различни експерименти.

Следващото предизвикателство беше съотношението на броя думи между пълния текст и резюмето. Както бе споменато, целим да генерираме резюмета със същата дължина като оригиналните резюмета. От друга страна, при изхода от GPT не можем да контролираме точно дължината на текста, а само максималната дължина. Когато извикваме GPT API, имаме максимален контекст от 4 000 токена, който включва както входа, така и изхода. След като извадим броя токени на промпта, ни остават по-малко от 4 000 за общия вход и изход. Първоначално опитахме да определим съотношението между оригиналния пълен текст и резюмето. Да кажем, че то е 9:1. Тогава бихме разделили

пълния текст на части, които заемат около 90% от разрешените токени и оставят 10% за изхода. Проблемът при тази стратегия беше, че в много случаи максималният брой токени за изход е толкова малък, че моделът не успява да завърши отговора.

След експерименти с различни максимални броеве токени, решихме да фиксираме желаното съотношение между резюме и пълен текст на 1:5 — приблизително 600 токена за резюмето и 3 000 за пълния текст. Умножаваме дължината на резюмето по 5. Допълнително умножаваме тази стойност по 1.5, тъй като не можем да зададем точния брой токени, а само максимума, което означава, че целим входните текстове да са 7.5 пъти по-дълги от резюметата, които се опитваме да генерираме. Ако оригиналният пълен текст е по-дълъг от това, добавяме стъпка за предварителна обработка за намаляване на съдържанието му. Изчисляваме оценките по TextRank за всички изречения в пълния текст и премахваме изреченията с най-ниски оценки, докато достигнем до това съотношение. Така подходът ни комбинира екстрактивни и абстрактивни методологии за постигане на целите си при по-дълги текстове. Ако съотношението е по-ниско от 7.5, не се извършва предварителна обработка.

След изпълнението на стъпката за предварителна обработка, текстът се разделя на части. Всяка част представлява списък от цели изречения, които не надвишават максималния разрешен лимит за токени. Всяка част се резюмира и резултатите се конкатенират.

4.9 Предизвикателства на различните езици

Първото предизвикателство за езиците, различни от английски език е, че популярните големи модели, готови за употреба са тренирани основно на английски и в някои случаи са напълно неизползваеми за други езици. За някои от тях, като например T5, в последствие излизат версии за множество езици, но количеството информация на други езици е пренебрежимо малко спрямо на английски. Причините за това са разнообразни, вариращи от наличието на повече текстове на английски до това, че инвестицията на английски е по-

смислена от бизнес гледна точка, защото има повече ползватели.

Следващото сериозно предизвикателство е, че има стъпки от процесите, които са общи за всички езици, но изискват различна имплементация за всеки език. Един такъв пример са секциите, от които се изкарва резюмето. Таблица 3 показва имената на секции за различните езици. Друго подобно предизвикателство е стемирането (свеждането на думата до нейна основа). То е специфично на всеки език, защото окончанията и формирането на думите е различно. Поради ненамиране на добро, готово за използване решение на някои езици, там не се прилага тази стъпка. Това може да доведе до по-лошо представяне на тези езици. Друга сходна тема са стопиращите думи. Те са различни за всеки език.

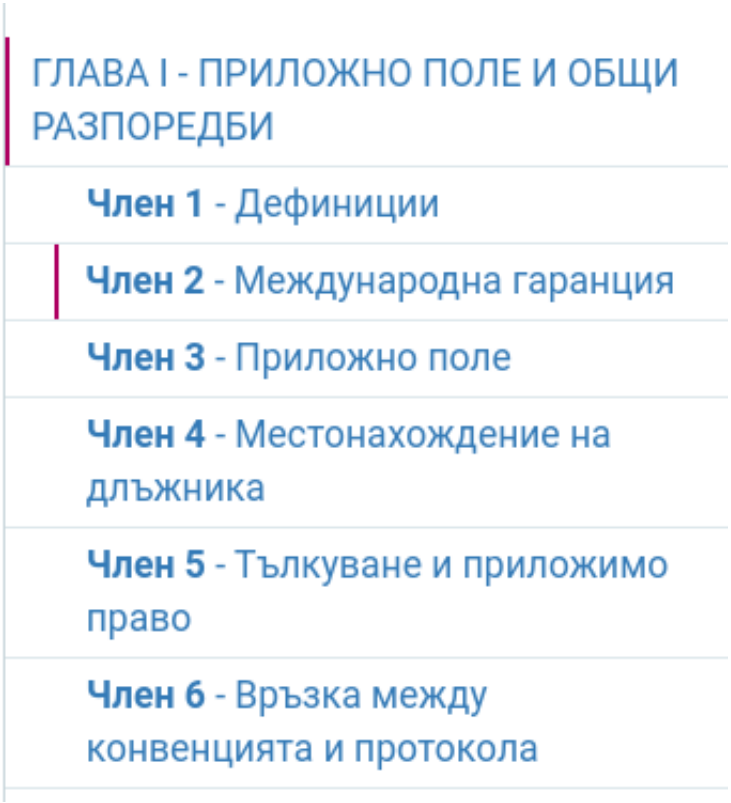
Друга много важна тема за разнообразието от езици е липсата на разбиране от страна на разработващите. При познание на конкретен език (български, английски) можем лесно да направим ръчна проверка на генерираните резултати. При другите езици това е много трудно, защото трябва да се използват автоматични средства за превод, а те могат да изменят текста.

4.10 StructExtSum - Нов екстрактивен подход основан на структура на документа

StructExtSum е нов екстрактивен подход за автоматично резюмиране, който използва структурните характеристики на правните документи с цел подобряване на качеството на генерираните резюмета. Разработването на този метод е мотивирано от спецификите, наблюдавани в новосъздадения корпус от правни текстове на български и английски език. Правните документи често имат отчетлива вътрешна организация – с абзаци, заглавия, подточки и логическа последователност, която стандартните модели не използват пълноценно. StructExtSum цели да използва тази структура, за да избира по-информативни и представителни сегменти от текста.

Една от характерните особености на използвания корпус, която

StructExtSum използва, е наличието на ясно изразена йерархична структура в документите. Пълните текстове са организирани чрез три вида структурни единици: Заглавие, Глава и Член. Въпреки че тяхната поява не е напълно консистентна между документите – някои съдържат само заглавия, други само глави или членове – тази структура предоставя възможност за логическо групиране и разграничаване на различни части от съдържанието. Заглавията обикновено въвеждат нов документ и понякога маркират границата между множество документи в един файл. Главите обединяват тематично свързани секции, а членовете представят отделни законодателни разпоредби. Всеки от тези елементи има номер и наименование (напр. ГЛАВА I ОБЩИ РАЗПОРЕДБИ, Член 1 Цел). Фигура 26 илюстрира примерна структура на документ с глава и членове.



Фигура 26: Пример за йерархична структура на документа [19]

След анализ на HTML съдържанието можем да идентифицираме къде за-

[19]Източник: https://eur-lex.europa.eu/legal-content/BG/TXT/?uri=celex:32009D0370

почва и свършва всеки структурен елемент. Това ни позволява да извлечем полезна информация за всяко изречение. Освен тях пресмятаме и други езикови характеристики на ниво изречение.

Доводите за използване на тази информация включват хипотези, че резюмето трябва да съдържа изречение от всяка част или че първите части и изречения са с по-важна информация.

Идентифицираме следните характеристики:

- Пореден номер на изречение – редът на всяко изречение се разделя на общия брой изречения в пълния текст.
- Пореден номер на Глава - въз основа на структурата на документите, те са разделени на глави. Всяко изречение е част от една глава. Тази характеристика показва номера на главата, част от която е изречението, разделен на общия брой глави.
- Пореден номер на изречението в рамките на главата - пореден номер на изречението за конкретната глава разделен на общия брой изречения в главата.
- Пореден номер на Член - въз основа на структурата на документите, те са разделени на членове. Всяко изречение е част от един член. Тази характеристика показва номера на члена, част от който е изречението, разделен на общия брой членове.
- Пореден номер на изречението в рамките на члена - пореден номер на изречението за конкретния член разделен на общия брой изречения в члена.
- Съотношение на стоп думите - Броят на стоп думите в изречението, разделен на общия брой думи в документа.
- Вече показаният TF-IDF екстрактивен подход разчита на тази функция. Тя показва резултата TF-IDF за всяко изречение по отношение на другите изречения в пълния текст.

- Стойността, която е резултат от TextRank алгоритъма.
- Броят на думите - броят на думите в изречението, разделен на броя на думите в най-дългото изречение в текста.

Всички характеристики имат стойности между нула и едно.

За всяко изречение изчисляваме ROUGE-1 като го сравним с оригиналното резюме. Формулираме задачата като дефинираме двойки - вектор от характеристики на изречение и целева стойност за предсказване от модела (ROUGE-1 на изречението). Нормализираме тези стойности, като ги разделим на максималния резултат. След това тренираме линейна регресия, за да предскажем ROUGE резултата на база на извлечените характеристики.

Разглеждаме задачата на ниво език и правим крос валидация с 10 части (folds). Разделяме текстовете на 10 равни части избрани на случаен принцип, тренираме моделът върху 90% от тях и предсказваме останалите 10%. Това се повтаря 10 пъти за всеки език. По този начин за всеки текст в набора ни от данни имаме резюме.

4.11 Техническа имплементация

Разработката на системата за автоматично създаване на резюмета на правни текстове е реализирана в *Google Colaboratory (Colab)*[20], което предоставя достъп до изчислителни ресурси (включително GPU) и удобна интеграция с Google Cloud Storage и GitHub. Проектът е структуриран в отделни модули, организирани под формата на Jupyter тетрадки (notebooks), което позволява лесно повторение и модификация на процеса. Цялостният процес по автоматично резюмиране е илюстриран чрез следните основни технологични стъпки – от първоначалното изтегляне на данни до анализа на резултатите:

- **Извличане на HTML документи (EUR-Lex)** – първоначално се изтеглят нужните правни документи от платформата EUR-Lex.

- **Съхранение в Google Cloud Storage** – изтеглените HTML документи се съхраняват в облачното хранилище.
- **Извличане на текст от HTML съдържанието** – от HTML файловете се извлича основният текст.
- **Генериране на резюмета (екстрактивни и абстрактивни методи)** – прилага се автоматизирано резюмиране.
- **Съхранение на резюметата в GCS** – генерираните резюмета също се запазват в облака.
- **Оценка с ROUGE** – извършва се автоматична оценка на качеството на резюметата с метриката ROUGE.
- **Анализ и визуализация на резултатите** – резултатите се анализират и визуализират с цел изводи и сравнения.

4.11.1 Архитектура на процеса

Целият работен поток (pipeline) е разделен на няколко ключови етапа:

1. **Извличане на данни**: Изходната информация се извлича от публично достъпния портал на Европейския съюз EUR-Lex[21]. Чрез Python скрипт, HTML съдържанието на страниците се изтегля и запазва в Google Cloud Storage (GCS).
2. **Предобработка на текст**: С използване на библиотеката `BeautifulSoup` се извлича само релевантният текст от HTML документите. Резултатите също се записват в GCS във вид на `.txt` файлове.
3. **Генериране на резюмета**: Системата използва над 10 различни подхода за обобщение – както екстрактивни (напр. TextRank, LexRank), така и абстрактивни (включително езикови модели като GPT). Всеки подход е имплементиран в отделна тетрадка, за да се осигури модуларност.

[20]Източник: `https://colab.research.google.com`

4. **Оценка на качеството**: Оценката се извършва с помощта на метриката ROUGE, използвайки библиотеката `rouge-score` на Google. Изчисляват се показатели като `ROUGE-1`, `ROUGE-2` и `ROUGE-L`, които сравняват автоматичното резюме с оригиналното резюме, предоставено от EUR-Lex.

5. **Анализ на резултатите**: Генерираните резултати от метриката се анализират статистически с помощта на `pandas`, `matplotlib` и `seaborn`.

4.11.2 Използвани технологии и библиотеки

Проектът е изцяло реализиран на езика Python. Основните използвани библиотеки включват:

- **Набавяне и обработка на данни:** `requests`, `BeautifulSoup4`, `html5lib`
- **Обработка на естествен език (NLP):** `nltk`, `spacy`, `sumy`, `transformers`
- **Генериране на резюмета:** `networkx` (за TextRank), `openai`, `huggingface/transformers`
- **Оценка:** `rouge-score`
- **Съхранение и хостинг:** `google-cloud-storage`
- **Анализ и визуализация:** `pandas`, `matplotlib`, `seaborn`

4.11.3 Инфраструктура и съхранение на данни

Всички файлове – оригинални HTML документи, извлечени текстове, генерирани резюмета и резултати от метриките – се съхраняват в Google Cloud Storage, като проектът използва структура от директории, съобразена с отделните етапи от анализа. Чрез интеграция с `gcsfs` и `google-cloud-storage`, четенето и записът на файлове се осъществяват директно от Colab.

[21] Източник: `https://eur-lex.europa.eu`

Изходният код е хостнат в GitHub хранилище (репозитори), което позволява лесен достъп, контрол на версиите и колаборация. Всички компоненти са документирани и организирани с цел максимална възпроизводимост.

4.11.4 Отворен код и възпроизводимост

Един от основните принципи в реализацията на проекта е отворен достъп и възможност за повторно използване. Всички скриптове и тетрадки са достъпни публично чрез GitHub, с ясно описание на необходимите зависимости и инструкции за стартиране. Проектът е структуриран така, че да позволява лесна адаптация, разширение и интеграция в различни изследователски и практически сценарии, свързани с резюмирането на правни документи.

5 Резултати от проведените експерименти

5.1 Дизайн на експериментите

Най-широко използваната метрика за оценяване на автоматично резюмиране на текстове е ROUGE (Recall-Oriented Understudy for Gisting Evaluation). ROUGE е набор от метрики, използвани за оценка на автоматично резюмиране и софтуер за машинен превод. Метриките сравняват автоматично създадено резюме с резюме, създадено от човек. ROUGE-N се отнася до съвпаденията на n-грамите между системните и референтните резюмета. ROUGE-L се основава на статистика за най-дългата обща подредености (Longest Common Subsequence, LCS). Проблемът с най-дългата обща подредености естествено разглежда структурното сходство на изреченията и автоматично идентифицира най-дългите последователни n-грами. В проведените експерименти бяха използвани по-специално ROUGE-1, ROUGE-2 и ROUGE-L F1 стойности.

5.2 Резултати за резюмиране на правни текстове на английски

Бяха проведени експерименти с различни алгоритми върху един и същ набор от данни. Те включват както екстрактивни, така и абстрактивни подходи. Някои от тях не изискват обучение, докато други се обучават от нулата или се донастройват върху набора от данни.

Алгоритмите T5 и PreSumm (на основата на BERT) имат ограничения за броя на токените в подадените и изходните данни. За да може данните да бъдат обработвани от тези алгоритми, пълните текстове и резюметата бяха разделени на части: пълните текстове съдържат по 1024 токена, а резюметата – 128 токена. Използвахме същата метрика ROUGE, за да оценим всяка част от резюметата спрямо съответната част от пълните текстове. По време на обучението всяка част от пълния текст беше свързвана с най-подходящата част от резюмето. По време на оценяването бяха генерирани резюмета за всички части на пълния текст, които след това бяха съединени, а резултатът беше оценен

спрямо оригиналното резюме.

Таблица 5 представя резултатите от проведените експерименти. Всички екстрактивни подходи превъзхождат базовия алгоритъм. *PreSumm* подобрява единствено *ROUGE-1* резултата от 26.52 (базов) на 29.25. Екстрактивният метод за резюмиране с *BERT* и *K-Means* дава най-добрите екстрактивни резултати – 36.82 *ROUGE-1* резултат. Когато използвахме същия алгоритъм, но заменихме *BERT* с предварително обучен *LEGAL-BERT*, настроен за правни текстове, резултатите бяха леко по-ниски – 36.06 *ROUGE-1*. Вземайки предвид, че оригиналните резюмета са създадени абстрактивно от експерти, резултат от 36.82 може да се счита за значителен успех.

И двата абстрактивни алгоритъма (*PreSumm* и *T5*) имат ограничения върху размера на входа и изхода. Пълните текстове и резюметата бяха разделени на части с размери 1024 и 128 съответно. По време на обучението всяка част от текста беше сдвоена с най-добрата част от резюмето с най-висок *ROUGE-1 F1* резултат. По време на оценката генерираните резюмета от всички части на текста бяха обединени и сравнени с оригиналното резюме. И двата подхода дадоха по-добри резултати от базовия метод.

Фино настроеният абстрактивен модел *T5-base* с модифицирана реализация за обработка на дълги текстове, разделени на части, показа най-добрите общи резултати. Ето първия параграф от резюме с най-висок резултат (0.65 *ROUGE-1 F1*):

Оригинал: *The European order for payment (EOP) procedure applies to all civil and commercial matters in cases where at least one of the parties lives in an EU country different from the one where the application for an order is made. The procedure does not apply to certain issues: revenue, customs or administrative matters, state liability for acts and omissions in the exercise of state authority, matrimonial property regimes, bankruptcy, proceedings relating to the winding-up of insolvent companies or other legal persons, and judicial arrangements, social security, claims arising from non-contractual obligations, unless there was an agreement between the parties or an admission of debt or they relate to liquidated*

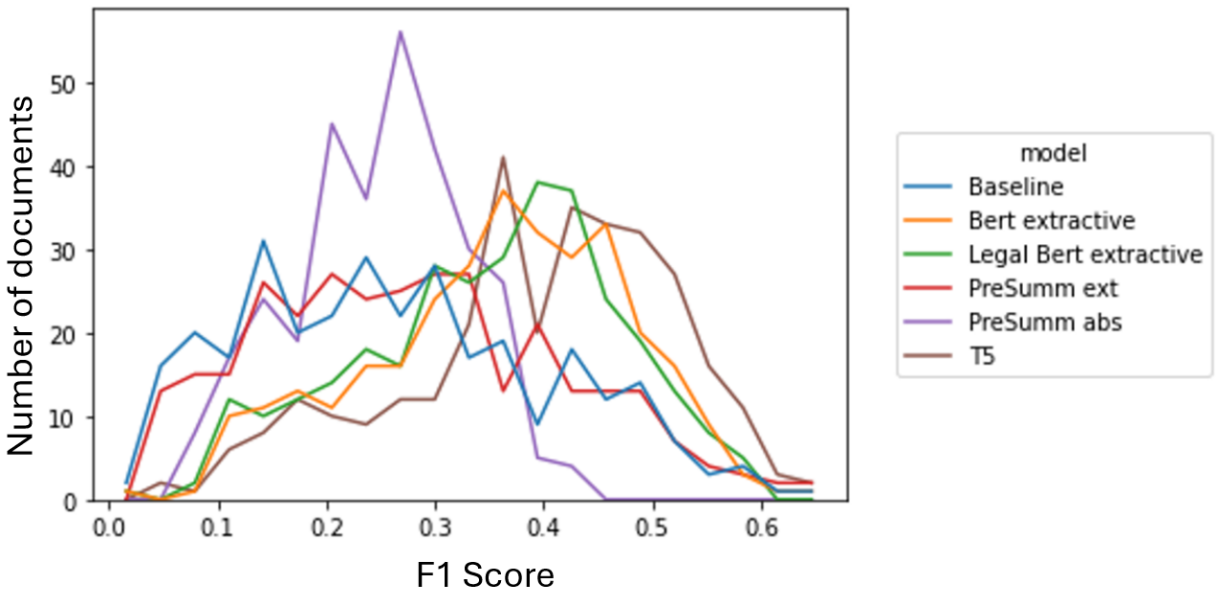
debts arising from joint ownership of property.

Генерирано: *a European order for payment procedure is established in the EU country where the claimant lives. Its purpose is to ensure that creditors and debtors have equal access to justice throughout the EU. The regulation also establishes an electronic system for determining which courts have jurisdiction to issue an order for payment, as well as a mechanism for the recovery of uncontested pecuniary claims.*

Фигура 27 показва разпределението на *ROUGE-1 F1* резултатите за всички изследвани модели. Кривата на абстрактивния модел *PreSumm* е най-близка до нормалното разпределение. Можем също така да наблюдаваме подобно поведение между кривата на базовия екстрактивен метод и *PreSumm* екстрактивен. Останалите имат сходни форми, а моделът *T5* се отличава.

Модел	**Тип**	**Метрика**	**Прецизност**	**Пълнота**	**F1**
Базов	екстрактивен	ROUGE-1	19.22	73.14	26.52
		ROUGE-2	7.41	29.94	10.41
		ROUGE-3	3.46	13.30	4.83
Резюмиране с	екстрактивен	ROUGE-1	35.85	54.16	36.82
BERT и K-Means		ROUGE-2	11.57	17.14	11.65
		ROUGE-3	4.64	6.23	4.48
Резюмиране с	екстрактивен	ROUGE-1	34.06	56.45	36.06
LEGAL-BERT		ROUGE-2	11.25	18.48	11.72
и K-Means		ROUGE-3	4.63	6.95	4.63
PreSumm	екстрактивен	ROUGE-1	22.64	71.80	29.25
		ROUGE-2	8.19	28.20	10.85
		ROUGE-3	3.47	11.52	4.57
PreSumm	абстрактивен	ROUGE-1	33.30	25.09	28.46
		ROUGE-2	5.41	4.08	4.63
		ROUGE-3	1.29	0.99	1.11
T5	абстрактивен	ROUGE-1	42.89	52.25	**39.27**
		ROUGE-2	15.94	18.97	**14.17**
		ROUGE-3	7.28	8.07	**6.28**

Таблица 5: Резултати от експерименталните модели. Фино настроеният модел *T5* генерира най-добрите резултати. Всички алгоритми подобриха базовия F1 резултат.



Фигура 27: Сравнителен анализ на резултатите, постигнати с различните модели.

5.3 Резултати за резюмиране на научни статии на английски

С цел експериментиране в други области бяха направени експерименти с резюмиране и върху корпуса от данни ScisummNet за научни статии (Yasunaga, Zhang, *et al.* 2017, Yasunaga, Kasai, *et al.* 2019). Той се състои от 1000 статии от сферата на компютърната лингвистика, тяхната мрежа на цитиране и техни ръчно направени резюмета. Оригиналният текст е в структуриран XML формат. Той позволява да разберем кой текст е част от въведението (Abstract) или от коя секция е. Като предварителна обработка на текстовете секцията Abstract беше премахната и алгоритмите са тренирани да генерират както резюметата генерирани от експерти, така и секцията Abstract.

При експериментите с научни статии бяха изпробвани 3 подхода - Екстрактивен подход основан на BERT и K-means, Pegasus и T5. Таблица 6 показва резултатите от проведените експерименти.

Алгоритъм	Цел на резюмиране	ROUGE-1	ROUGE-2	ROUGE-3
Bert-K-means	Abstract	**28.52**	7.46	3.38
Pegasus	Abstract	22.55	**14.63**	**6.60**
T5	Abstract	26.49	10.20	6.22
Bert-K-means	Резюме	**34.43**	8.43	3.54
Pegasus	Резюме	27.28	**16.43**	**7.56**
T5	Резюме	30.11	11.24	7.45

Таблица 6: Резултатите при резюмиране на статии с различни подходи. Генерираното резюме се сравнява както със секцията Abstract, така и с ръчно генерираните резюмета.

5.4 Резултати за резюмиране на правни текстове на английски и български език чрез анализ на структурата

Таблица 7 показва резултатите от различните подходи. Подходът с линейна регресия на основата на структурни характеристики (StructExtSum), подобрява екстрактивния подход основан на TF-IDF с 2.5% и базовия с 6.5%. Докато базовият и екстрактивният подход не изискват обучение и са тествани върху всички документи, последният подход е тестван чрез 10-кратно кръстосано валидиране върху всички документи.

Алгоритъм	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**
Baseline (first N sentences)	28.22	9.97	22.87
TF-IDF	29.29	10.91	23.00
StructExtSum	**30.04**	**11.15**	**23.51**
SlavicBERT and K-Means	19.69	7.33	17.53

Таблица 7: Представяне на StructExtSum спрямо други алгоритми за текстове на български език.

StructExtSum може лесно да бъде приложен към други езици, ако могат да бъдат извлечени характеристиките за всяко изречение. За да потвърдим допълнително ефективността на алгоритъма, създадохме набор от правни данни на английски език. Използвахме същия подход за събиране, предварителна обработка, извличане на характеристики и генериране на резюмета. Таблица 8 показва, че StructExtSum отново успява да превъзхожда базовия подход и този на основата на TF-IDF.

Алгоритъм	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**
Baseline (first N sentences)	36.23	11.10	29.05
TF-IDF	37.31	11.77	29.32
StructExtSum	**38.99**	**12.20**	**30.16**
SlavicBERT and K-Means	36.82	11.65	28.53

Таблица 8: Представяне на StructExtSum спрямо други алгоритми за текстове на английски език.

5.5 Резултати за резюмиране на правни текстове на 24 езика с използване на големи езикови модели

В тази секция са представени експерименти, проектирани за цялостно оценяване и сравнение на изброените подходи за резюмиране на текстове на 24 езика, използвайки набора от данни за правни документи.

Започнахме с генериране на резюмета, които имат същата дължина като оригиналното резюме. Така прецизността (precision) и пълнотата (recall) са идентични. По този начин избягваме проблема с промяна на F-стойността поради генериране на по-дълги или по-къси резюмета и се фокусираме изцяло върху релевантността на изреченията, а не върху дължината на резюмето. За всички изпробвани подходи избрахме само изречения с поне 3 думи.

Таблица 9 показва резултатите от различните подходи. Подходът TF-IDF подобрява базовия за всички езици. TextRank и GPT превъзхождат TF-IDF за всички езици. За 8 от езиците GPT превъзхожда TextRank, докато за останалите TextRank е най-добрият.

Език	Базов подход	TF-IDF	TextRank	GPT
BG	0.284	0.297	0.318	**0.323**
CS	0.248	0.269	0.285	**0.287**
DA	0.338	0.350	0.379	**0.389**
DE	0.326	0.338	**0.368**	0.364
EL	0.279	0.286	0.309	**0.316**
EN	0.364	0.384	**0.416**	0.405
ES	0.381	0.389	**0.414**	0.408
ET	0.194	0.209	**0.232**	0.216
FI	0.252	0.252	**0.286**	0.269
FR	0.375	0.385	**0.416**	0.396
GA	0.335	0.328	**0.348**	0.324
HR	0.235	0.243	**0.268**	0.268
HU	0.290	0.290	**0.320**	0.318
IT	0.378	0.375	**0.412**	0.399
LT	0.235	0.242	**0.262**	0.252
LV	0.238	0.238	**0.265**	0.261
MT	0.229	0.232	**0.265**	0.242
NL	0.302	0.308	0.336	**0.339**
PL	0.254	0.260	**0.279**	0.271
PT	0.383	0.391	**0.423**	0.388
RO	0.359	0.377	**0.402**	0.400
SK	0.237	0.251	0.269	**0.270**
SL	0.249	0.253	0.284	**0.288**
SV	0.330	0.327	0.370	**0.374**

Таблица 9: ROUGE-1 F1 стойности за всички 24 езика за всички типове експерименти.

5.6 Човешка оценка на качеството на резюметата

В допълнение към количествените метрики, които са използвани за автоматична оценка на резултатите от различните подходи за автоматично резюмиране, важно е да се разгледа и човешка оценка. Такава оценка цели да представи субективно, но реалистично възприятие върху генерираните резюмета. Това е от съществено значение при обработката на правни документи, където точността на формулировките и запазването на логическите зависимости имат критично значение.

Следващият анализ е базиран на наблюдение и сравнение на реално генерирани резюмета по всяка от разглежданите техники. За всеки подход е направена преценка на силните и слабите му страни в контекста на юридически и научни текстове, като акцентът е поставен върху пригодността му за реални приложения.

5.6.1 Екстрактивни подходи

Резюметата, които са генерирани с екстрактивни подходи, са по-подходящи за четене от експерти в сферата. Целта е да могат да се съсредоточат върху важните части, без да има нужда те да са логически свързани. Следват 2 примера на резюмета, които са генерирани с екстрактивни подходи. В различна степен те успяват да уловят ключовите части на документа, но са изключително трудни за четене от хора, които не са експерти.

Пример 1 *Поради това е целесъобразно да се предвиди възможност лицата, освободени от санкция в съответствие с решение на орган за защита на конкуренцията съгласно програма за освобождаване от санкция или намаляване на санкции, да бъдат защитени от неоправданото им излагане на искове за обезщетение за вреди, като се има предвид, че решението на органа за защита на конкуренцията, с което се установява нарушението, може да се превърне в окончателно за лицето, освободено от санкция, по-рано, откол-*

което за другите предприятия, които не са били освободени от санкция, като по този начин потенциално прави лицето, освободено от санкция, предпочитана цел при съдебно производство. „иск за обезщетение за вреди“ означава иск съгласно националното право, с който пред национален съд се предявява претенция за претърпени вреди от лице, което твърди, че е увредено или от лице, което действа от името на едно или няколко лица, за които се твърди, че са увредени, когато правото на Съюза или националното право предоставя тази възможност, или от физическо или юридическо лице, което е правоприемник на лице, за което се твърди, че е увредено, включително и лицето, придобило правото на иск;

Пример 2 *да преодолеят мисленето, почиващо върху идеята за секторна изолация, в традиционните области на политиката, като включат в по-голяма степен изкуството, културата и творчеството в стратегиите за икономически растеж, в социалните политики, в градското и регионалното развитие и устойчивото развитие; да повишават осведомеността на всички заинтересовани страни, по-специално лицата, отговорни за формулирането на политиките, творците, отраслите, предприятията и инвеститорите за културната, икономическата и обществената стойност на творчеството и иновациите в рамките на междусекторното сътрудничество. Където е уместно, да насърчават междусекторните подходи между различните области на обучението в рамките на висшите учебни заведения, например чрез съвместни програми по култура и изкуства, наука, инженерни науки, технологии, стопанска дейност и други подходящи области;*

Базов подход (избиране на първите изречения) Този подход се характеризира с ниска сложност. Основното му ограничение е, че началните изречения на текстовете рядко съдържат обобщената същност на съдържанието, особено при правни документи, където ключовата информация обикновено се появява по-късно в текста. Генерираните резюмета са формално коректни, но често неинформативни и нецелесъобразни.

Екстрактивен подход основан на TF-IDF Този подход предлага подобрение спрямо базовия чрез селекция на по-информативни изречения. Все пак, липсата на разбиране за контекста води до включване на изолирани фрази, които не са свързани логически помежду си. Често липсва плавност между изреченията, което се отразява негативно на възприемането на резюмето.

TextRank Алгоритъмът TextRank показва подобрени резултати при текстове със стабилна структура. Избраните изречения обикновено представляват обобщения на ключови пасажи. Независимо от това, ограниченията на този модел не позволяват дълбоко разбиране на съдържанието, поради което някои важни аспекти могат да бъдат изпуснати.

BERT + K-means Използването на предварително обучен BERT модел с клъстеризация води до селекция на семантично разнородни изречения, които по-добре обхващат съдържанието на документа. Генерираните резюмета са по-балансирани и смислово богати, но в някои случаи могат да съдържат повтаряща се информация от различни клъстери.

LegalBERT + K-means LegalBERT предоставя по-добро разбиране на терминологията и стилистиката на юридически документи. Резюметата, създадени с този модел, демонстрират по-висока точност при запазване на правната логика и формулировки. Това ги прави предпочитан избор за юридически контекст.

StructExtSum StructExtSum използва структурата на документа. Подходът се представя отлично при документи с ясни сегменти като "Факти "Правна обосновка"и "Изводи". Резултатите са структурирани, последователни и добре обхващат логиката на оригиналния текст. Този метод е особено подходящ за правни документи, съдържащи формализирана структура.

5.6.2 Абстрактивни подходи

Резюметата, които са генерирани с абстрактивни подходи, са по-подходящи за четене от хора, които не са експерти. При тях текстът е логически свързан и последователен. Следват 3 примера на резюмета, които са генерирани с абстрактивни подходи. Тук има риск да бъдат променени ключови части от текста, но като цяло той е много по-разбираем и лесен за четене.

Пример 1 (GPT) *РЕГЛАМЕНТ (ЕС) 2015/1588 НА СЪВЕТА, като взе предвид Договора за функционирането на Европейския съюз, определя прилагането на членове 107 и 108 от Договора, включващи категории на държавна помощ. Комисията оценява съвместимостта на помощта с вътрешния пазар. Регламентите за групово освобождаване гарантират правна сигурност. Държавната помощ се определя в член 107, параграф 1 от Договора. Комисията има правомощия за групово освобождаване само за мерки, които отговарят на всички критерии по член 107, параграф 1 от Договора. Иновациите са приоритет на политиката на Съюза. Мерките за помощ в областта на културата и спорта трябва да бъдат съвместими с вътрешния пазар, като се вземат предвид основната компетентност на държавите членки и специалната закрила на културното многообразие. Помощта за широколентова инфраструктура и финансирането на инфраструктура може да бъде освободена от задължението за уведомяване. Мерките трябва да се съобразяват с целите на Съюза и да ограничават нарушаването на конкуренцията. Праговете за помощта могат да бъдат определени като интензитет на помощта или максимален размер. Текстът разглежда важността на мерките за постигане на целите на Съюза и възможността за освобождаване от тях. Предлага определяне на прагове за държавно подпомагане и насърчаване на прозрачността чрез публикуване на информация на уебсайта на Комисията. Текстът подчертава необходимостта от контрол върху предоставянето на помощи и предлага преразглеждане на категориите помощи. Комисията трябва да може да отменя или изменя регламентите при промени на обстоятелствата. Осигурява сътрудничество между Комисията и държа-*

вите членки чрез консултации и предоставяне на информация. Подчертава прозрачността и контрола по прилагането на груповите освобождавания. Регламентите се прилагат за определен срок и могат да бъдат отменяни или изменяни при промени на обстоятелствата. Комисията представя доклад за прилагането на регламента всяки пет години. В Комисията се публикува проект на регламент, за да се даде възможност на заинтересованите лица да представят възраженията си. Създава се Консултативен комитет по държавна помощ. Проектите и документите се публикуват на уебсайта на Комисията. Срокът за представяне на възраженията не може да бъде по-кратък от един месец. Срокът може да се съкращава или удължава за консултации, по спешност или при определени случаи. Комитетът излиза със становище по проекта, а всяка държава членка може да поиска вписване на нейната позиция в протокола. Отменя се Регламент (ЕО) № 994/98 и настоящият регламент е задължителен и се прилага във всички държави членки.

Пример 2 (T5) *requirements apply to all new types of vehicles and their payloads. Manufacturers must ensure that the sound level of each vehicle in decibels (dB) is displayed on the vehicle's front and rear windows, as well as on consumer information. Manufacturers must also ensure that the sound level of each vehicle in decibels (dB) is measured in decibels (dB) and on labelling conditions applicable to air and noise pollution. The regulation does not apply to hybrid electric and pure electric vehicles of category N1 if one of the following conditions is met: engine capacity does not exceed 660 cm*

Пример 3 (PreSumm) *the organisation for economic co-operation and development for use in a biogas or composting plant .<q>in 2013 , the legislation was amended to give mayotte , as a new outermost region of the eu , until 1 january 2021 to comply with its rules . the organisation for economic co-operation and development for use in a biogas or composting plant .<q>in 2013 , the legislation was amended to give mayotte , as a new outermost region of the eu , until 1 january 2021 to comply with its rules . the organisation for economic co-operation and*

development for use in a biogas or composting plant .<q>in 2013 , the legislation was amended to give mayotte , as a new outermost region of the eu , until 1 january 2021 to comply with its rules .

PreSumm PreSumm комбинира екстрактивен и абстрактивен компонент, като така се стреми да извлече смислово богати сегменти и да ги организира в резюме, наподобяващо човешка формулировка. В някои ситуации PreSumm не успява да генерира смислен текст и генерира повтарящ се безсмислен такъв (показан пример).

T5 Абстрактивният модел T5 демонстрира способност за създаване на плавни и естествени резюмета. Предимството му е в стилистичната гладкост, но понякога резюметата са прекалено обобщени и пропускат детайли, които са от значение за юридическия контекст. При научни текстове представянето е по-добро, тъй като структурата на оригинала често следва установени шаблони. Генерираният текст не е граматически добре оформен.

GPT Моделите от семейство GPT се открояват със способността си да създават смислени и граматично правилни текстове. Генерираните резюмета често наподобяват стил на човешки автор. Основен риск е появата на хипотетични или невалидни изводи, особено при липса на достатъчен контекст. Текстовете са разбираеми и удобни за четене.

Човешката оценка подчертава разликите между подходите, които автоматичните метрики не винаги успяват да разкрият. Макар екстрактивните методи със специализирано обучение да остават предпочитани в контексти, изискващи висока прецизност на терминологията и логическа последователност, абстрактивните модели отбелязват значителен напредък и се оформят като по-достъпна и интуитивна алтернатива за широка аудитория, особено с оглед на тяхната способност за обобщение и адаптация.

6 Заключение и бъдеща работа

В рамките на настоящата дисертация беше изследвано как съвременните методи за обработка на естествен език могат да бъдат използвани за подобряване на качеството на резюмирането, особено в контекста на дълги и сложни текстове. Разработеният модел StructExtSum демонстрира, че структурираната информация в документите може да се използва ефективно за извличане на релевантни пасажи. Освен това, създаденият многоезиков корпус за правни текстове предлага нови възможности за изследвания в областта на автоматичното резюмиране и обработката на естествен език. В следващите раздели са представени основните научни и приложни приноси на дисертацията.

6.1 Приноси на дисертацията

Настоящата дисертация предоставя значителни приноси както в теоретичен, така и в практически аспект, свързани с областта на автоматичното резюмиране на текстове. Резултатите от изследването разширяват разбирането за приложението на съвременни методи за обработка на естествен език в контекста на сложни и дълги текстове, като правни документи и научни статии.

6.1.1 Научни приноси

Разработване на нов метод за екстрактивно резюмиране: Предложен е моделът StructExtSum, който използва структурата на документите за идентифициране на ключови елементи в правни текстове. Моделът е тестван върху многоезикови данни и демонстрира конкурентни резултати спрямо съществуващите подходи.

Анализ на многоезиковото автоматично резюмиране: Дисертацията изследва ефективността на съществуващите езикови модели върху множество езици, като отчита специфичните предизвикателства на нискоресурсни езици

като български.

Сравнителен анализ на предварително обучени модели: Проведените експерименти с BERT, T5 и GPT показват предимствата и ограниченията на тези модели при обработката на правни текстове. Анализът предоставя насоки за избора на подходящ модел в зависимост от специфичния контекст.

Създаване на нов корпус за правни текстове: Изграден е многоезиков набор от данни за резюмиране на правни документи, включващ 24 езика. Този ресурс е публично достъпен и осигурява ценна база за бъдещи изследвания и разработка на нови модели.

6.1.2 Приложни приноси

Автоматизирано обобщаване на правни документи: Разработените методи могат да бъдат интегрирани в правни системи за автоматично извличане на съществени части от текстове, улеснявайки работата на юристи и адвокати.

Оптимизация на достъпа до информация: Чрез подобреното резюмиране на дълги документи потребителите могат бързо да получат най-важната информация, без да е необходимо да четат изцяло съдържанието.

Интеграция в съществуващи решения: Разработените алгоритми могат да бъдат внедрени в софтуерни платформи за управление на документи, подпомагайки автоматизацията на процеси като търсене, анализ и синтез на информация.

Подобрена обработка на нискоресурсни езици: Създаденият многоезиков корпус и проведените анализи допринасят за развитието на инструменти за обработка на български език и други нискоресурсни езици, които често са недостатъчно представени в съществуващите системи за обработка на естествен език.

6.1.3 Публикации по темата на дисертацията

В рамките на изследователската работа по темата на дисертацията бяха публикувани четири научни статии, които допринасят за разширяването на познанията и дискусиите в областта. Тези публикации отразяват основните аспекти на изследването, неговите теоретични и приложни измерения, както и получените резултати. Следва списък на публикациите, свързани с темата на дисертацията, които са публикувани в рецензирани научни издания и конференции:

1. Valentin Zmiycharov, Milen Chechev, Gergana Lazarova, Todor Tsonkov, and Ivan Koychev. "A Comparative Study on Abstractive and Extractive Approaches in Summarization of European Legislation Documents." Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP). 2021. `https://aclanthology.org/2021.ranlp-1.184`

   В тази публикация се изследва ефективността на различни методи за автоматично резюмиране на правни текстове на английски език от Европейското законодателство. Представен е сравнителен анализ между базови, екстрактивни и абстрактивни модели (PreSumm и T5). Статията допринася с оценка на представянето на съвременни езикови модели в юридически контекст.

2. Todor Tsonkov, Gergana Lazarova, Valentin Zmiycharov, and Ivan Koychev. "A Comparative Study of Extractive and Abstractive Approaches for Automatic Text Summarization on Scientific Texts." Proceedings of the International Conference on Education and Research in the Information Society (ERIS). 2021. `https://ceur-ws.org/Vol-3061/`

   Публикацията представя сравнително изследване на екстрактивен подход с BERT и абстрактивни (Pegasus и T5) методи за автоматично резюмиране на научни текстове на английски език. Проучването анализира качеството на резюметата по различни метрики и оценява приложимостта на различните техники в контекста на академични публикации.

3. Valentin Zmiycharov, Gergana Lazarova, Todor Tsonkov, and Ivan Koychev. "StructExtSum – Bulgarian Legislation Text Extractive Summarization by Structure Understanding." Proceedings of the International Conference on Education and Research in the Information Society (ERIS). 2022. `https://ceur-ws.org/Vol-3372/`

   Публикацията въвежда иновативен екстрактивен подход — StructExtSum — за автоматично резюмиране на правни текстове, основан на анализ на документната структура. За първи път са включени текстове на български език, като се разглежда и приложението на метода върху правни документи на английски език. Резултатите показват значително подобрение при използване на структурна информация.

4. Valentin Zmiycharov, Ivan Koychev, and Todor Tsonkov. "EurLexSummarization – A New Text Summarization Dataset on EU Legislation in 24 Languages with GPT Evaluation." Proceedings of the International Conference on Computational Linguistics in Bulgaria (CLIB). 2024. `https://dcl.bas.bg/clib/proceedings/`

   Публикацията представя новоразработен многоезичен корпус за автоматично резюмиране на правни документи с текстове и експертно създадени резюмета на 24 езика. Върху този набор са проведени експерименти с големия езиков модел GPT, като се оценява неговата ефективност при генериране на резюмета, сравними по качество с човешки. Представен е метод за създаване на промптове, съобразени със специфичния правен стил, както и обстойни резултати от сравнение с други техники за резюмиране (TF-IDF, TextRank и др.).

6.2 Посоки за бъдещи изследвания

Настоящото изследване предлага множество насоки за бъдеща работа, които могат да разширят неговия обхват и да адресират някои от ограниченията, идентифицирани в рамките на дисертацията. Те включват както подобрения на използваните модели, така и проучвания в нови и недостатъчно изследвани

области.

- **Включване на нови езици**: Разширяването на набора от данни с допълнителни езици би позволило валидиране на предложените модели в още по-разнообразен многоезиков контекст.
- **Обогатяване на данните с метаинформация**: Добавянето на семантични етикети и метаинформация към текстовете би могло да улесни обучението на по-сложни модели, които използват контекстуална информация.
- **Комбиниране на екстрактивни и абстрактивни подходи**: Въпреки фокуса върху екстрактивното резюмиране, бъдещи изследвания могат да се насочат към комбинирани модели, които съчетават силните страни на двата подхода.
- **Интеграция на домейн-специфични знания**: Разработването на модели, които използват специфични за правната област знания, би могло да подобри качеството на резюметата, като същевременно запази тяхната прецизност.
- **Резюмиране на мултимодални данни**: Успешното резюмиране на документи, които съдържат както текст, така и изображения, таблици или графики, представлява интересна и важна насока за бъдещи изследвания.
- **Изследвания върху динамични текстове**: Разглеждане на възможности за резюмиране на текстове в реално време, като например правни новини или потоци от съдебни решения.
- **Проверка на точността**: Бъдещи проучвания могат да се фокусират върху минимизиране на грешките в резюметата, които могат да доведат до погрешна интерпретация на правни текстове.
- **Прозрачност и обяснимост на моделите**: Разработване на подходи, които осигуряват обясними резултати, би могло да повиши доверието към автоматизираните системи за резюмиране.

- **Разработване на нови метрики**: Настоящите метрики като ROUGE имат ограничения. Създаването на нови показатели, които по-добре оценяват качеството на резюметата, би могло значително да подпомогне областта.

- **Човешка оценка**: Повишаване на ролята на човешките експерти в оценяването на автоматично генерирани резюмета, което ще допринесе за по-добро съответствие с очакванията на потребителите.

Литература

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". *arXiv e-prints*, arXiv:1409.0473 (Sept. 2014), arXiv:1409.0473. DOI: 10.48550/arXiv.1409.0473. arXiv: 1409.0473 [cs.CL].

[2] Satanjeev Banerjee and Alon Lavie. "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments". *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, June 2005, pp. 65–72. URL: https://aclanthology.org/W05-0909.

[3] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. "Research-paper recommender systems: A literature survey". *International Journal on Digital Libraries* (July 2015), pp. 1–34. DOI: 10.1007/s00799-015-0156-0.

[4] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. "A large annotated corpus for learning natural language inference". *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015, pp. 632–642. DOI: 10.18653/v1/D15-1075. URL: https://aclanthology.org/D15-1075.

[5] Erik Cambria and Bebo White. "Jumping NLP Curves: A Review of Natural Language Processing Research [Review Article]". *IEEE Computational Intelligence Magazine* 9.2 (2014), pp. 48–57. DOI: 10.1109/MCI.2014.2307227.

[6] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. "LEGAL-BERT: The Muppets straight out of Law School". *arXiv e-prints*, arXiv:2010.02559 (Oct. 2020), arXiv:2010.02559. DOI: 10.48550/arXiv.2010.02559. arXiv: 2010.02559 [cs.CL].

[7] Yen-Chun Chen and Mohit Bansal. "Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting". *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 675–686. DOI: 10.18653/v1/P18-1063. URL: https://aclanthology.org/P18-1063.

[8] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches". *arXiv e-prints*, arXiv:1409.1259 (Sept. 2014), arXiv:1409.1259. DOI: 10.48550/arXiv.1409.1259. arXiv: 1409.1259 [cs.CL].

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". *arXiv e-prints*, arXiv:1810.04805 (Oct. 2018), arXiv:1810.04805. DOI: 10.48550/arXiv.1810.04805. arXiv: 1810.04805 [cs.CL].

[10] Katja Filippova. "Multi-Sentence Compression: Finding Shortest Paths in Word Graphs". *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Beijing, China: Coling 2010 Organizing Committee, Aug. 2010, pp. 322–330. URL: https://aclanthology.org/C10-1037.

[11] David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. "English gigaword". *Linguistic Data Consortium, Philadelphia* 4.1 (2003), p. 34.

[12] William Hamilton, Rex Ying, and Jure Leskovec. "Inductive Representation Learning on Large Graphs" (June 2017).

[13] Hardy Hardy, Shashi Narayan, and Andreas Vlachos. "HighRES: Highlight-based Reference-less Evaluation of Summarization". *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3381–3392. DOI: 10.18653/v1/P19-1330. URL: https://aclanthology.org/P19-1330.

[14] Mourad Heddaya, Kyle MacMillan, Anup Malani, Hongyuan Mei, and Chenhao Tan. *CaseSumm: A Large-Scale Dataset for Long-Context Summarization from U.S. Supreme Court Opinions*. 2024. eprint: 2501.00097 (cs.CL).

[15] Julia Hirschberg and Christopher Manning. "Advances in natural language processing". *Science (New York, N.Y.)* 349 (July 2015), pp. 261–266. DOI: 10.1126/science.aaa8685.

[16] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". *Neural Computation* 9.8 (Nov. 1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735. eprint: https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

[17] Daniel Jurafsky and James Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Vol. 2. Feb. 2008.

[18] Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. "Natural Language Processing: State of The Art, Current Trends and Challenges". *arXiv e-prints*, arXiv:1708.05148 (Aug. 2017), arXiv:1708.05148. DOI: 10.48550/arXiv.1708.05148. arXiv: 1708.05148 [cs.CL].

[19] K. Koumpis and S. Renals. "Content-based access to spoken audio". *IEEE Signal Processing Magazine* 22.5 (2005), pp. 61–69. DOI: 10.1109/MSP.2005.1511824.

[20] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. "From Word Embeddings To Document Distances". *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 957–966. URL: https://proceedings.mlr.press/v37/kusnerb15.html.

[21] Quoc V. Le and Tomas Mikolov. "Distributed Representations of Sentences and Documents". *arXiv e-prints*, arXiv:1405.4053 (May 2014), arXiv:1405.4053. DOI: 10.48550/arXiv.1405.4053. arXiv: 1405.4053 [cs.CL].

[22] Logan Lebanoff, Kaiqiang Song, and Fei Liu. "Adapting the Neural Encoder-Decoder Framework from Single to Multi-Document Summarization". *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 4131–4141. DOI: 10.18653/v1/D18-1446. URL: https://aclanthology.org/D18-1446.

[23] R. Likert. "A technique for the measurement of attitudes." *Archives of Psychology* (1932).

[24] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. URL: https://aclanthology.org/W04-1013.

[25] Yang Liu and Mirella Lapata. "Text Summarization with Pretrained Encoders". *arXiv e-prints*, arXiv:1908.08345 (Aug. 2019), arXiv:1908.08345. DOI: 10.48550/arXiv.1908.08345. arXiv: 1908.08345 [cs.CL].

[26] Kangcheng Luo, Quzhe Huang, Cong Jiang, and Yansong Feng. *Automating Legal Concept Interpretation with LLMs: Retrieval, Generation, and Evaluation*. 2025. eprint: 2501.01743 (cs.CL).

[27] Polina Mamoshina, Armando Vieira, Evgeny Putin, and Alexander Zhavoronkov. "Applications of Deep Learning in Biomedicine". *Molecular pharmaceutics* 13 (Mar. 2016). DOI: 10.1021/acs.molpharmaceut.5b00982.

[28] Inderjeet Mani. "Summarization Evaluation: An Overview" (June 2002).

[29] Inderjeet Mani and Mark Maybury. "Automatic Summarization." Jan. 2001, p. 5.

[30] Rada Mihalcea and Paul Tarau. "TextRank: Bringing Order into Text". *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Ed. by Dekang Lin and Dekai Wu. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 404–411. URL: https://aclanthology.org/W04-3252.

[31] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space". *arXiv e-prints*, arXiv:1301.3781 (Jan. 2013), arXiv:1301.3781. DOI: 10.48550/arXiv.1301.3781. arXiv: 1301.3781 [cs.CL].

[32] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient Estimation of Word Representations in Vector Space". *arXiv e-prints*, arXiv:1301.3781 (Jan. 2013), arXiv:1301.3781. DOI: 10.48550/arXiv.1301.3781. arXiv: 1301.3781 [cs.CL].

[33] Preslav Nakov. "Design and Evaluation of Inflectional Stemmer for Bulgarian" (Jan. 2003).

[34] Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos santos, Caglar Gulcehre, and Bing Xiang. "Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond". *arXiv e-prints*, arXiv:1602.06023 (Feb. 2016), arXiv:1602.06023. DOI: 10.48550/arXiv.1602.06023. arXiv: 1602.06023 [cs.CL].

[35] Ani Nenkova and Amit Bagga. "Facilitating email thread access by extractive summary generation". Jan. 2003, pp. 287–296. DOI: 10.1075/cilt.260.32nen.

[36] Ani Nenkova and Kathleen McKeown. *Automatic Summarization*. Vol. 5. June 2011. DOI: 10.1561/1500000015.

[37] Nikola I. Nikolov and Richard Hahnloser. "Large-Scale Hierarchical Alignment for Data-driven Text Rewriting". *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*. Varna, Bulgaria: INCOMA Ltd., Sept. 2019, pp. 844–853. DOI: 10.26615/978-954-452-056-4_098. URL: https://aclanthology.org/R19-1098.

[38] OpenAI *et al.* "GPT-4 Technical Report". *arXiv e-prints* (2024). eprint: 2303.08774 (cs.CL). URL: https://arxiv.org/abs/2303.08774.

[39] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. "The PageRank Citation Ranking : Bringing Order to the Web". *The Web Conference*. 1999. URL: https://api.semanticscholar.org/CorpusID:1508503.

[40] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. "Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features". *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, June 2018, pp. 528–540. DOI: 10.18653/v1/N18-1049. URL: https://aclanthology.org/N18-1049.

[41] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "Bleu: a Method for Automatic Evaluation of Machine Translation". *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, July 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://aclanthology.org/P02-1040.

[42] Jeffrey Pennington, Richard Socher, and Christopher Manning. "GloVe: Global Vectors for Word Representation". *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: https://aclanthology.org/D14-1162.

[43] John Platt, Kristina Toutanova, and Wen-tau Yih. "Translingual Document Representations from Discriminative Projections". *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Cambridge, MA: Association for Computational Linguistics, Oct. 2010, pp. 251–261. URL: https://aclanthology.org/D10-1025.

[44] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". *arXiv e-prints*, arXiv:1910.10683 (Oct. 2019), arXiv:1910.10683. DOI: 10.48550/arXiv.1910.10683. arXiv: 1910.10683 [cs.LG].

[45] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. "SQuAD: 100,000+ Questions for Machine Comprehension of Text". *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, Nov.

2016, pp. 2383–2392. DOI: 10.18653/v1/D16-1264. URL: https://aclanthology.org/D16-1264.

[46] Luca Rolshoven, Vishvaksenan Rasiah, Srinanda Brügger Bose, Matthias Stürmer, and Joel Niklaus. *Unlocking Legal Knowledge: A Multilingual Dataset for Judicial Summarization in Switzerland*. 2024. eprint: 2410.13456 (cs.CL).

[47] Alexander M. Rush, Sumit Chopra, and Jason Weston. "A Neural Attention Model for Abstractive Sentence Summarization". *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015, pp. 379–389. DOI: 10.18653/v1/D15-1044. URL: https://aclanthology.org/D15-1044.

[48] T. Y. S. S. Santosh, Cornelius Weiss, and Matthias Grabmair. *LexSumm and LexT5: Benchmarking and Modeling Legal Summarization Tasks in English*. 2024. eprint: 2410.09527 (cs.CL).

[49] Raphael Schumann, Lili Mou, Yao Lu, Olga Vechtomova, and Katja Markert. "Discrete Optimization for Unsupervised Sentence Summarization with Word-Level Extraction". *arXiv e-prints*, arXiv:2005.01791 (May 2020), arXiv:2005.01791. DOI: 10.48550/arXiv.2005.01791. arXiv: 2005.01791 [cs.CL].

[50] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. "Collective Classification in Network Data". *AI Magazine* 29.3 (Sept. 2008), p. 93. DOI: 10.1609/aimag.v29i3.2157. URL: https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2157.

[51] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank". *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, Oct. 2013, pp. 1631–1642. URL: https://aclanthology.org/D13-1170.

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. "Attention Is All You Need". *arXiv e-prints*, arXiv:1706.03762 (June 2017), arXiv:1706.03762. DOI: 10.48550/arXiv.1706.03762. arXiv: 1706.03762 [cs.CL].

[53] Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex Wade, Kuansan Wang, Nancy Xin Ru Wang, Chris Wilhelm, Boya Xie, Douglas Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. "CORD-19: The COVID-19 Open Research Dataset". *arXiv e-prints*, arXiv:2004.10706 (Apr. 2020), arXiv:2004.10706. DOI: 10.48550/arXiv.2004.10706. arXiv: 2004.10706 [cs.DL].

[54] Sander Wubben, Antal van den Bosch, and Emiel Krahmer. "Sentence Simplification by Monolingual Machine Translation". *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 1015–1024. URL: https://aclanthology.org/P12-1107.

[55] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention". *arXiv e-prints*, arXiv:1502.03044 (Feb. 2015), arXiv:1502.03044. DOI: 10.48550/arXiv.1502.03044. arXiv: 1502.03044 [cs.LG].

[56] Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. "Optimizing Statistical Machine Translation for Text Simplification". *Transactions of the Association for Computational Linguistics* 4 (Dec. 2016), pp. 401–415. DOI: 10.1162/tacl_a_00107.

[57] Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander Fabbri, Irene Li, Dan Friedman, and Dragomir Radev. "ScisummNet: A Large Annotated Corpus and Content-Impact Models for Scientific Paper Summarization with Citation Networks". *Proceedings of AAAI 2019*. 2019.

[58] Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir R. Radev. "Graph-based Neural Multi-Document Summarization". *Proceedings of CoNLL 2017*. 2017.

Списък на фигурите

1 Разгъната рекурентна невронна мрежа. [22] 19

2 Long short-term memory (LSTM). [23] 20

3 Gated recurrent unit (GRU). [24] 21

4 Механизъм на самовнимание (примерно изречение). [25] 23

5 Механизъм на самовнимание (архитектура). 24

6 Механизъм на внимание с много глави (примерно изречение). [26] 25

7 Позиционно кодиране за изречението “Аз съм робот.”. [27] 27

8 Архитектура на Трансформатор (Transformer) с един слой енкодер (отляво) и един слой декодер (отдясно). [28] 29

9 Архитектура на BERT [29] 31

10 Възможности на модела T5 [30] 34

11 Архитектура на GPT. [31] 36

12 Мрежа граф на публикации в Reddit. [32] 42

13 Заглавна част на документ. 50

14 Референции на документ. 50

15 Брой документи с определен брой секции. Фигурата показва, че най-голямата част от документите съдържат 6,7 или 8 секции. . 51

16 Резюме с липсващ текст. 52

17 Брой думи в пълните документи. Наличието на отклонения с над 100000 думи изкривяват статистиката за среден брой думи. . . . 53

18 Разпределение на броя думи на пълните документи след премахване на отклоненията. 54

19 Разпределение на броя думи на резюметата след премахване на отклоненията. 55

20 Всяка точка представлява данните за броя думи на едно резюме и съответния брой думи на пълния документ след премахване на отклоненията. Линията и пространството около нея показва модел на линейна регресия, който се доближава най-много до съотношението в броя думи между резюметата и пълните документи. 56

21 Начин на работа на алгоритъма TextRank 64

22 Начин на работа на алгоритъма K-means [33] 66

23 Начин на резюмиране с BERT и K-means 68

24 Сравнение на стандартна употреба на BERT и употребата му за резюмиране [34] 69

25 Структура на промпт 72

26 Пример за йерархична структура на документа [35] 75

27 Сравнителен анализ на резултатите, постигнати с различните модели. 85

Списък на таблиците

1 Имена на секции и брой документи, в които се срещат (10 най-популярни). 52

2 Брой документи по категории на база на първата дума в текста. 57

3 Сравнение между данните на всички езици 60

4 Връзки между примерните уеб страници 64

5 Резултати от експерименталните модели. Фино настроеният модел *T5* генерира най-добрите резултати. Всички алгоритми подобриха базовия F1 резултат. 84

6 Резултатите при резюмиране на статии с различни подходи. Генерираното резюме се сравнява както със секцията Abstract, така и с ръчно генерираните резюмета. 86

7 Представяне на StructExtSum спрямо други алгоритми за текстове на български език. 87

8 Представяне на StructExtSum спрямо други алгоритми за текстове на английски език. 87

9 ROUGE-1 F1 стойности за всички 24 езика за всички типове експерименти. 89

ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ НА РЕЗУЛТАТИТЕ

Декларирам, че настоящата дисертация съдържа оригинални резултати, получени при проведени от мен научни изследвания. Резултатите, които са получени, описани и/или публикувани от други учени, са надлежно и подробно цитирани в библиографията.

Настоящата дисертация не е прилагана за придобиване на научна степен в друго висше училище, университет или научен институт.

Подпис: