



Софийски университет “Св. Климент Охридски”

Факултет по математика и информатика

Катедра “Софтуерни технологии”

Автореферат

на 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

1	Въведение	4
1.1	Мотивация и актуалност на темата	4
1.2	Цел и задачи на дисертационния труд	4
2	Обзор на предметната област	6
3	Нов корпус от данни за правни текстове и техните резюмета	9
3.1	Въведение	9
3.2	Данни на български език	10
3.3	Данни на други езици	12
4	Разработени и адаптирани подходи за резюмиране на текстове	13
4.1	Базов подход (избиране на първите изречения в текста)	13
4.2	Екстрактивен подход основан на TF-IDF	14
4.3	Екстрактивен подход TextRank	15
4.4	Екстрактивен подход основан на BERT и K-means	17
4.5	Екстрактивен подход основан на LegalBERT и K-means	19
4.6	Модел за резюмиране PreSumm	20
4.7	Езиков модел T5	22
4.8	Голям езиков модел GPT	23

4.9	StructExtSum - Нов екстрактивен подход основан на структура на документа	25
5	Резултати от проведените експерименти	30
5.1	Дизайн на експериментите	30
5.2	Резултати за резюмиране на правни текстове на английски . . .	30
5.3	Резултати за резюмиране на правни текстове на английски и български език чрез анализ на структурата	34
5.4	Резултати за резюмиране на правни текстове на 24 езика с използване на големи езикови модели	36
6	Заключение и бъдеща работа	38
6.1	Приноси на дисертацията	38
6.1.1	Научни приноси	38
6.1.2	Приложни приноси	39
6.1.3	Публикации по темата на дисертацията	40
6.2	Посоки за бъдещи изследвания	41
	Литература	43
	Декларация за оригиналност на резултатите	46

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. 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 текста, като същевременно гарантира, че смисълът е запазен. Това съкращава времето, необходимо за разбиране на дълги материали, без да се пропуска критична информация.

Такива, дълги и сложни за интерпретиране материали, са правните текстове и закони. Общото при тях е, че текстовете са дълги и не са с фиксирана дължина. При правните текстове например се наблюдават огромни вариации в дължината на текстовете и техните резюмета. Това прави поставената задача още по-предизвикателна, а бизнес ползата - по-голяма. Правилното и бързо интерпретиране на правни закони е от ключово значение за немалко бизнеси и автоматичното им резюмиране може да помогне това да се случва по-ефективно.

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 и разработване на подходи 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.

2 Обзор на предметната област

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 разговори с потребители и преобразуване на реч в текст (Khurana *et al.* 2017).

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 захванват 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 самоорганизируют 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 пред-

варителни познания в лингвистичната област.

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.

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.

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.

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 препоредена версия на оригиналния документ. При абстракт-

тното 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).

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.

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 данни¹, 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 съюз². 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.

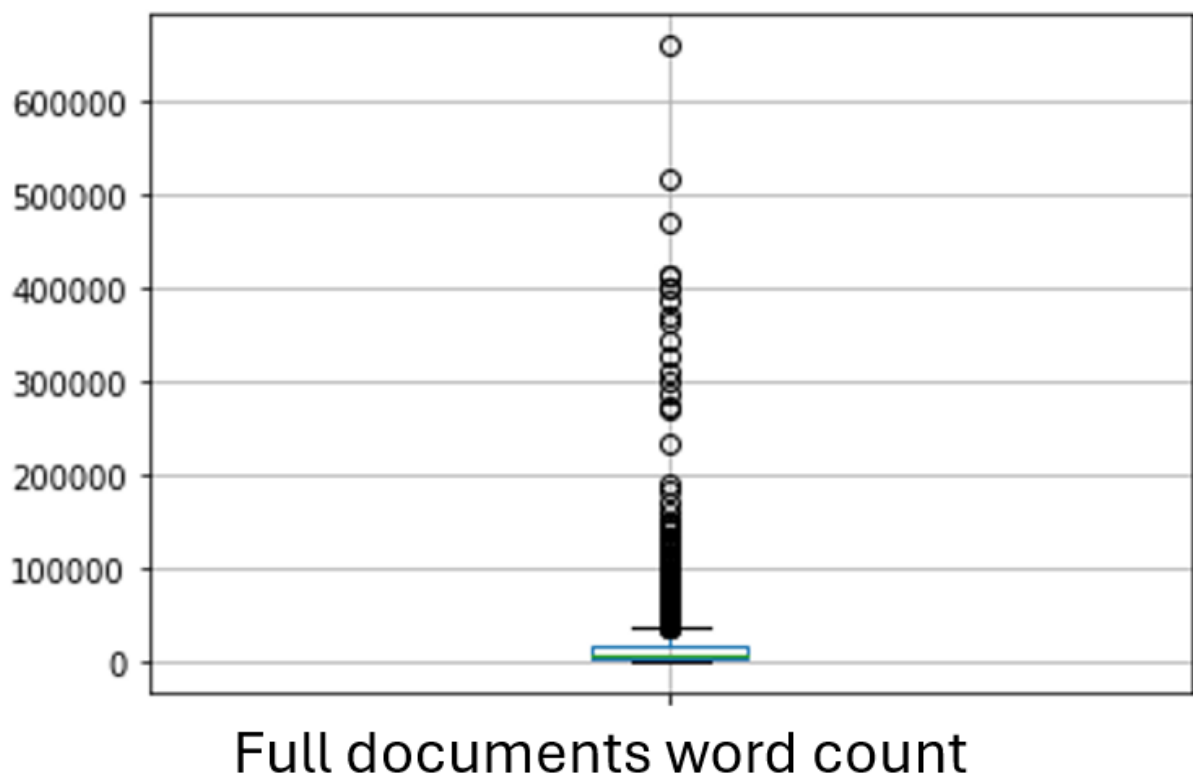
Има много характеристики на набора от данни, които го правят ценен за изследователската сфера. Той съдържа текст на специфична тематика, което е предизвикателство за претренираните модели на общ текст. Текстовете са с различна дължина, като някои документи са изключително дълги. Наличието им на различни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

¹https://huggingface.co/datasets/FMISummarization/FMI_Summarization

²<https://eur-lex.europa.eu/homepage.html>

3.2 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 средно 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 1).

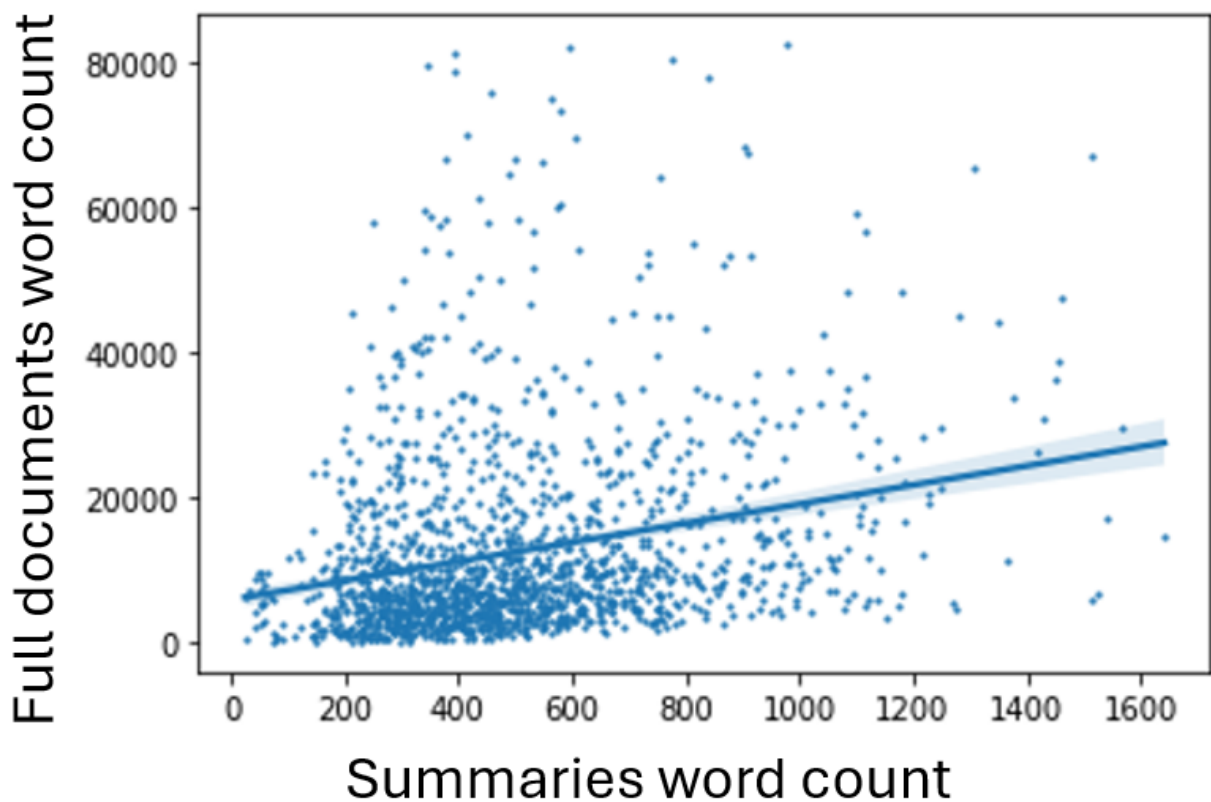


Фигура 1: Брой думи в пълните документи. Наличието на отклонения с над 100000 думи изкривяват статистиката за среден брой думи.

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 2 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 2: 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.

3.3 Данни на други езици

Използвайки същата методология бяха изтеглени данните на всички 24 налични езика. 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.

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 базирани препоръчителни системи в цифровите библиотеки използват tf-idf (Beel *et al.* 2015).

В нашия случай използваме TF-IDF на ниво изречение в текста спрямо всички 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-

ставяват 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”, “то”, “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

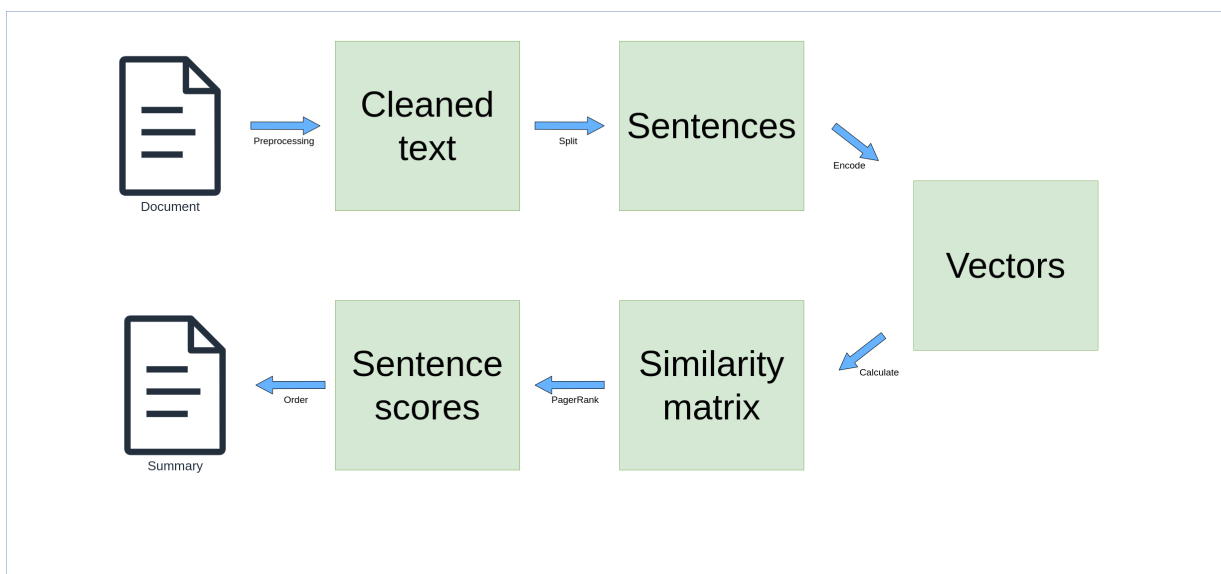
Преди да навлезем в детайли как работи TextRank (Mihalcea and Tarau 2004) алгоритъма, трябва да обясним PageRank (Page *et al.* 1999) алгоритъма, на който той е базиран. PageRank се използва предимно за подреждане на уеб страници в резултатите от онлайн търсене. Да предположим, че имаме 4 уеб страници — w_1 , w_2 , w_3 и w_4 . 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 1: Връзки между примерните уеб страници

Таблица 1 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 3: Начин на работа на алгоритъма TextRank

Фигура 3 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.* 2013). 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.

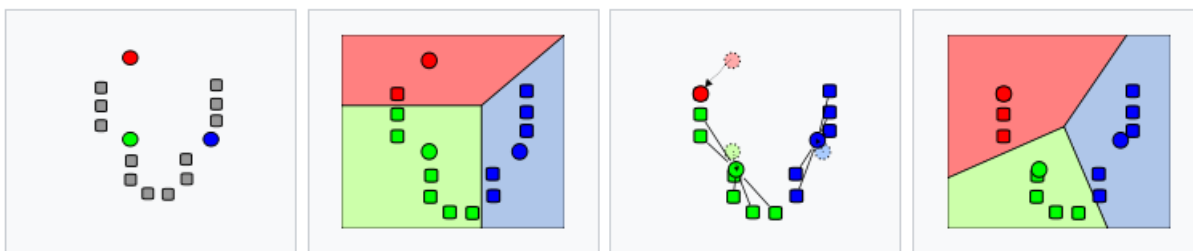
След като имаме оценките, подобно на предишния алгоритъм, сортираме изреченията по получените резултати в низходящ ред и конкатенираме изреченията поред докато не достигнем дължината на оригиналното резюме.

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 4).



Фигура 4: Начин на работа на алгоритъма K-means ³

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.

³Източник: https://en.wikipedia.org/wiki/K-means_clustering

Също както при всички екстрактивни подходи до момента, тук първо изчисляваме 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).

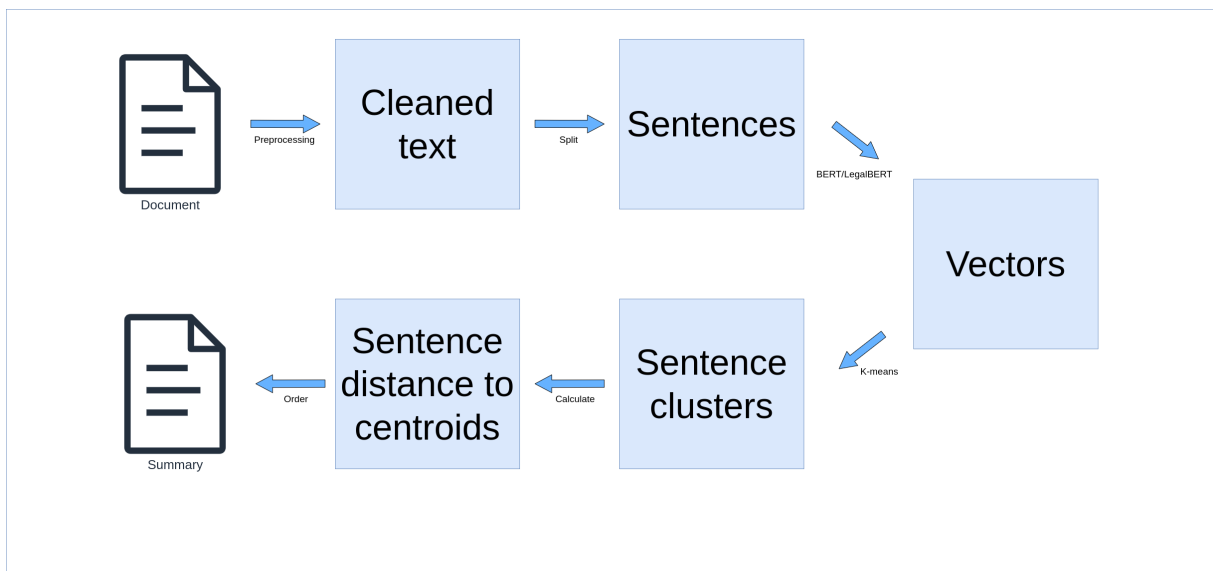
След това за всеки вектор на изречение изчисляваме разстоянието от центроида. Понякога центроидите са действителните изречения и в този случай разстоянието е нула. От всеки клъстер избираме едно или няколко изречения, които са най-близки до центроида (Фигура 5).

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.

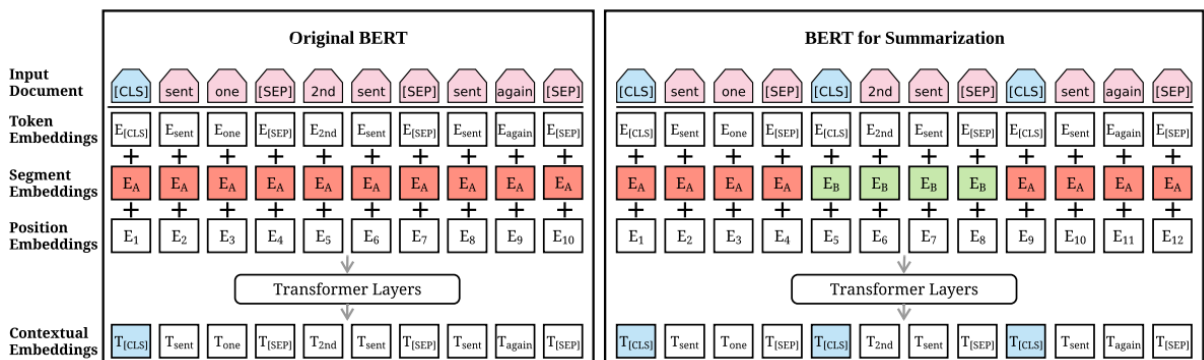


Фигура 5: Начин на резюмиране с BERT и K-means

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 6 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 представянния на изречения и използване на вграждания за сегментиране на интервали (илюстрирани в червен и зелен цвят) за разграничаване на множество изречения.



Фигура 6: Сравнение на стандартна употреба на BERT и употребата му за резюмиране ⁴

За целите на екстрактивно резюмиране, авторите използват вгражданията, генерирани от BertSum (архитектурата показана на Фигура 6). След това те използват допълнително вкаранияте 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,

⁴Източник: <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 по-дългите 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 ⁵. 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

⁵Източник: 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 G5 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 7). 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.

Контекст (описание на ситуация) - незадължително

Инструкция / въпрос / задача

Входни данни / условия / примери – незадължително

Изходни данни / указания / формат / тон / стил – незадължително

Ти си адвокат. Ти ще отговориш правилно на въпросите относно следния текст:

{Пълният текст на документа}

Резюмирай текста

без въвеждащи думи

Фигура 7: 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 StructExtSum - Нов екстрактивен подход основан на структурата 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 логи-

ческа 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 8 илюстрира примерна структура на документ с глава и членове.

ГЛАВА I - ПРИЛОЖНО ПОЛЕ И ОБЩИ РАЗПОРЕДБИ
Член 1 - Дефиниции
Член 2 - Международна гаранция
Член 3 - Приложно поле
Член 4 - Местонахождение на длъжника
Член 5 - Тълкуване и приложимо право
Член 6 - Връзка между конвенцията и протокола

Фигура 8: Пример за йерархична структура на документа ⁶

След анализ на HTML съдържанието можем да идентифицираме къде започва 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 резултатът 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.

⁶Източник: <https://eur-lex.europa.eu/legal-content/BG/TXT/?uri=celex:32009D0370>

- 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.

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. Метриците 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 подреденост (Longest Common Subsequence, LCS). Проблемът с най-дългата обща подреденост 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 2 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, показва 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*

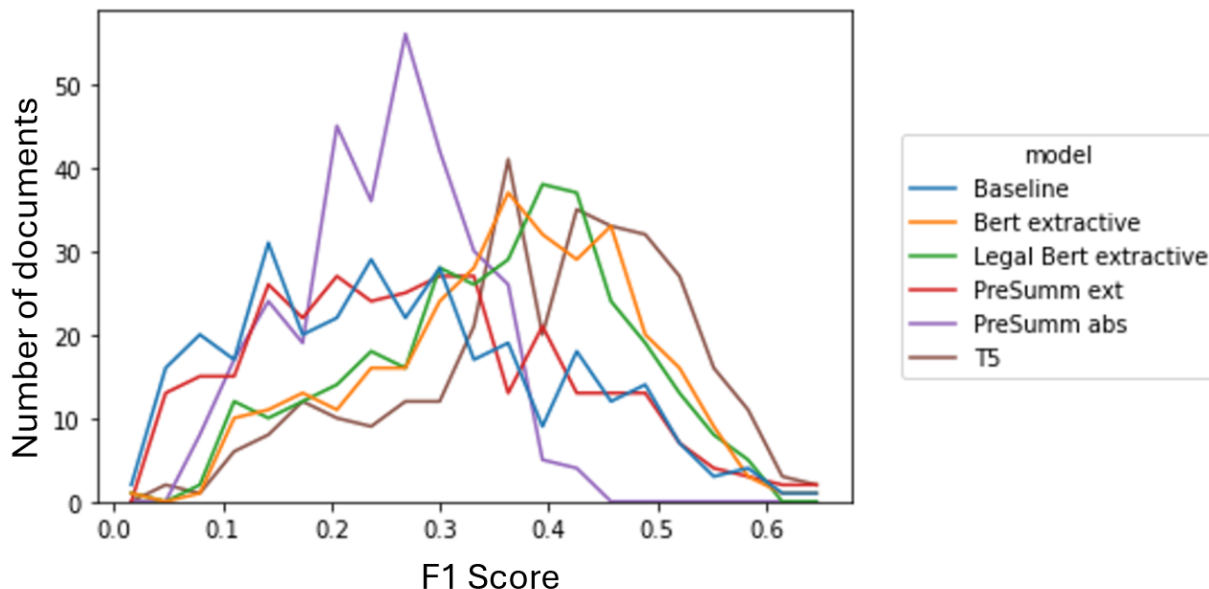
debts arising from joint ownership of property.

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 9 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 BERT и K-Means	екстрактивен	ROUGE-1	35.85	54.16	36.82
		ROUGE-2	11.57	17.14	11.65
		ROUGE-3	4.64	6.23	4.48
Резюмиране с LEGAL-BERT и K-Means	екстрактивен	ROUGE-1	34.06	56.45	36.06
		ROUGE-2	11.25	18.48	11.72
		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 2: Резултати от експерименталните модели. Фино настроеният модел *T5* генерира най-добрите резултати. Всички алгоритми подобриха базовия F1 резултат.



Фигура 9: Сравнителен анализ на резултатите, постигнати с различните модели.

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

Таблица 3 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 3: 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 4 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 4: Представяне на StructExtSum спрямо други алгоритми за текстове на английски език.

5.4 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 5 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 5: ROUGE-1 F1 стойности за всички 24 езика за всички типове експерименти.

6 Заключение и бъдеща работа

Автоматичното резюмиране на текстове е предизвикателна задача, която има значително приложение в различни сфери, включително правото, науката и индустрията. В рамките на настоящата дисертация беше проучено 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 пасажки. 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.

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>
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/>
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/>
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/>

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 метайнформация:** Добавянето на семантични етикети и метайнформация 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] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breiting. “Research-paper recommender systems: A literature survey”. *International Journal on Digital Libraries* (July 2015), pp. 1–34. DOI: 10.1007/s00799-015-0156-0.
- [2] 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.
- [3] 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].
- [4] 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>.
- [5] 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].
- [6] Daniel Jurafsky and James Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Vol. 2. Feb. 2008.
- [7] 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].

- [8] 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>.
- [9] 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].
- [10] Inderjeet Mani and Mark Maybury. “Automatic Summarization.” Jan. 2001, p. 5.
- [11] 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>.
- [12] 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].
- [13] Preslav Nakov. “Design and Evaluation of Inflectional Stemmer for Bulgarian” (Jan. 2003).
- [14] Ani Nenkova and Kathleen McKeown. *Automatic Summarization*. Vol. 5. June 2011. DOI: 10.1561/15000000015.
- [15] OpenAI *et al.* “GPT-4 Technical Report”. *arXiv e-prints* (2024). eprint: 2303.08774 (cs.CL). URL: <https://arxiv.org/abs/2303.08774>.
- [16] 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>.

- [17] 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>.

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: