

**Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky**

**Využitie veľkých jazykových modelov pre
analýzu rozhodnutí slovenských súdov**

Diplomová práca

**Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky**

**Využitie veľkých jazykových modelov pre
analýzu rozhodnutí slovenských súdov**

Diplomová práca

Študijný program:	Hospodárska informatika
Študijný odbor:	Informatika
Školiteľ:	doc. Ing. Peter Bednár, PhD.
Školiace pracovisko:	Ústav umelej inteligencie (ÚUI)

Košice 2026

Bc. Bianka Šimková

Abstrakt v slovenčine

Táto práca spája najnovšie metódy umelej inteligencie s právnou doménou slovenskej judikatúry. Cieľom tejto práce je využiť veľké jazykové modely na analýzu rozhodnutí slovenských súdov. V rámci implementácie boli texty rozhodnutí predspracované a rozdelené na logické časti s využitím regulárnych výrazov. Následne bolo porovnávaných deväť jazykových modelov pomocou metrík Perplexity, BLEU a ROUGE na úlohu generovania výroku rozhodnutia. Na základe tohto porovnania boli vybrané a dotrénované dva modely Qwen3-8B a mistral-sk-7b s cieľom adaptácie na slovenskú právnu doménu, a taktiež na úlohu predikcie úspešnosti odvolania (zrušenia, resp. zmenu rozhodnutia) prístupom zero-shot a few-shot, ako aj pridaním klasifikačnej vrstvy. Na tento účel bola použitá vybraná vzorka z 4,1 miliónov digitalizovaných rozhodnutí slovenských súdov do roku 2023. Kvantitatívne aj kvalitatívne vyhodnotenie potvrdzuje zlepšenie jazykovej reprezentácie po doladení modelov. Pri predikcii zrušenia rozhodnutia dosiahol najvyššiu úspešnosť model s klasifikačnou vrstvou, a to vyššiu ako 80%. Vytvorené modely predstavujú dobrý základ pre ďalší budúci výskum v oblasti použitia veľkých jazykových modelov v slovenskej judikatúre. Z praktického hľadiska môžu výstupy práce slúžiť ako podpora rozhodovania pre právnikov alebo aj bežných občanov, a to najmä pri posudzovaní pravdepodobnosti úspešnosti odvolania, ktoré v prípade neúspešnosti predstavuje ďalšie náklady a záťaž nielen pre občana, ale aj pre súdny systém.

Kľúčové slová v slovenčine

spracovanie prirodzeného jazyka, súdne rozhodnutia, právny text, veľké jazykové modely, hlboké učenie, predikčný model, úspešnosť súdneho odvolania

Abstrakt v angličtine

This work combines the latest artificial intelligence methods with the legal domain of Slovak jurisprudence. The aim of this work is to use large language models to analyze decisions of Slovak courts. As part of the implementation, the texts of decisions were pre-processed and divided into logical parts using regular expressions. Subsequently, nine language models were compared using the Perplexity, BLEU and ROUGE metrics for the task of generating the court statement. Based on this comparison, two models Qwen3-8B and mistral-sk-7b were selected and fine-tuned for slovak legal domain adaptation, and also for the task of predicting the success of an appeal (reversal or change of the decision) using the zero-shot and few-shot methods, as well as by adding a classification layer. For this purpose, a selected sample from 4.1 million digitized decisions of Slovak courts until 2023 was used. Both quantitative and qualitative evaluations confirm the improvement of language representation after fine-tuning the models. In predicting the reversal of a decision, the model with a classification layer achieved the highest success rate, exceeding 80%. The created models represent a good basis for further future research in the field of the use of large language models in Slovak jurisprudence. From a practical point of view, the outputs of the work can serve as decision-making support for lawyers or ordinary citizens, especially when assessing the probability of success of an appeal, which in case of failure represents additional costs and burden not only for the citizen but also for the judicial system.

Kľúčové slová v angličtine

Natural Language Processing, Court Decisions, Legal Text, Large Language Models, Deep Learning, Prediction Model, Court Appeal Success

Bibliografická citácia

ŠIMKOVÁ, Bianka. *Využitie veľkých jazykových modelov pre analýzu rozhodnutí slovenských súdov*. Košice: Technická univerzita v Košiciach, Fakulta elektrotechniky a informatiky, 2026. 79 s. Vedúci práce: doc. Ing. Peter Bednár, PhD.

TECHNICKÁ UNIVERZITA V KOŠICIACH

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Katedra kybernetiky a umelej inteligencie

ZADANIE DIPLOMOVEJ PRÁCE

Študijný odbor: **Informatika**

Študijný program: **Hospodárska informatika**

Názov práce:

Využitie veľkých jazykových modelov pre analýzu rozhodnutí slovenských súdov

Using large language models for analyzing decisions of Slovak courts

Študent:

Bc. Bianka Šimková

Školiteľ:

doc. Ing. Peter Bednár, PhD.

Školiace pracovisko:

Katedra kybernetiky a umelej inteligencie

Konzultant práce:

Pracovisko konzultanta:

Pokyny na vypracovanie diplomovej práce:

1. Podať teoretický prehľad problematiky využitia jazykových modelov pre analýzu súdnych rozhodnutí
2. Navrhnuť a implementovať vhodné jazykové modely pre analýzu súdnych rozhodnutí
3. Vhodne spracovať dostupné verejné dáta zo súdnych rozhodnutí
4. Otestovať a vyhodnotiť implementované metódy na vytvorených testovacích dátach
5. Vypracovať dokumentáciu podľa pokynov katedry a vedúceho práce

Jazyk, v ktorom sa práca vypracuje:

slovenský

Termín pre odovzdanie práce:

24.04.2026

Dátum zadania diplomovej práce:

31.10.2025




.....
prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som záverečnú prácu vypracovala samostatne s použitím uvedenej odbornej literatúry.

Podakovanie

Týmto by som chcela poďakovať vedúcemu mojej diplomovej práce, doc. Ing. Petrovi Bednárovi, PhD., za jeho odborné vedenie, rešpektujúci a podporujúci prístup, ochotu vždy si nájsť čas na moje zoznamy otázok k práci a za promptné reakcie. Jeho podpora a cenné rady mi veľmi pomohli a vďaka jeho vedeniu som získala množstvo cenných poznatkov.

Zároveň by som chcela poďakovať mojej sestre, Ing. Nikole Šimkovej, PhD., za jej podporu, cenné podnety, právne rady, pracovitosť a húževnatosť, ktorá je pre mňa v živote veľkou inšpiráciou.

V neposlednom rade chcem poďakovať mojej mamke za neustálu podporu, pochopenie a povzbudenie počas celého štúdia a aj mimo neho.

Obsah

1 Úvod	1
1.1 Motivácia	1
1.2 Formulácia úlohy	2
2 Analýza	4
2.1 Štruktúra súdov v SR	4
2.2 Prehľad súvisiacich štúdií	5
2.3 Metódy a technológie	10
2.3.1 Spracovanie prirodzeného jazyka	10
2.3.2 Neurónové siete	14
2.3.3 Veľké jazykové modely	24
3 Implementácia	32
3.1 Pochopenie cieľa	32
3.2 Pochopenie dát	33
3.2.1 Štruktúra rozhodnutí súdu	33
3.2.2 Textový korpus	33
3.3 Predspracovanie	38
3.3.1 Normalizácia a čistenie dát	38
3.3.2 Segmentácia textu rozhodnutí na logické časti	40
3.4 Modelovanie	42
3.4.1 Porovnanie jazykových modelov pri generovaní rozsudku	42
3.4.2 Dotrénovanie jazykového modelu	50
3.4.3 Klasifikátor na predikciu úspešnosti odvolania	51
4 Vyhodnotenie	58
4.1 Generovanie výroku	58
4.1.1 Kvantitatívne porovnanie	58
4.1.2 Kvalitatívne vyhodnotenie	59
4.2 Klasifikátor na predikciu úspešnosti odvolania	62

4.3 Diskusia	65
5 Záver	68
Zoznam použitej literatúry	69
Zoznam skratiek	80
Slovník termínov	83
Zoznam príloh	84

Zoznam obrázkov

2.1	Schéma hierarchie súdov pred reformou.	4
2.2	Schéma hierarchie súdov po reforme.	5
2.3	Schéma predspracovania textu.	11
2.4	Rôzne techniky embeddingu.	12
2.5	Princíp fungovania umelého neurónu.	14
2.6	Princíp fungovania umelej neurónovej siete.	15
2.7	Vizualizácia aktivačných funkcií.	17
2.8	Základná architektúra CNN.	18
2.9	Schéma architektúry RNN.	19
2.10	Schéma architektúry BiLSTM.	20
2.11	Porovnanie LSTM a GRU.	21
2.12	Schéma mechanizmu pozornosti v transformer architektúre.	22
2.13	Transformer architektúra.	23
2.14	Ukážka konverzácie s programom ELIZA.	25
2.15	BERT oproti GPT architektúre.	26
2.16	Porovnanie veľkostí LLM.	27
2.17	Porovnanie MHA, GQA a MQA.	28
2.18	Porovnanie pôvodného mechanizmu pozornosti, SWA, dilated SWA a kombinácie.	29
2.19	Znázornenie výpočtu návratnosti pre BERTScore.	30
3.1	Ukážka rozhodnutia súdu vo forme pdf.	34
3.2	Počet rozhodnutí podľa konkrétnych typov súdov.	35
3.3	Počet rozhodnutí s konkrétnou dĺžkou tokenov po 99. percentil.	35
3.4	Počet rozhodnutí za konkrétne roky.	36
3.5	Počet rozhodnutí podľa oblasti právnej úpravy.	36
3.6	Počet prvostupňových rozhodnutí podľa povahy rozhodnutia.	37
3.7	Počet rozhodnutí súdov vyšších stupňov podľa povahy rozhodnutia.	37
3.8	Porovnanie dĺžky všetkých a napadnutých rozhodnutí.	38

3.9	Porovnanie počtu tokenov v rozhodnutiach s odôvodnením a bez odôvodnenia.	41
3.10	Porovnanie Perplexity score.	46
3.11	Porovnanie BLEU score.	47
3.12	Porovnanie ROUGE-1 score.	47
3.13	Porovnanie ROUGE-2 score.	48
3.14	Porovnanie ROUGE-L score.	48
3.15	Konečné porovnanie modelov.	50
3.16	Priebeh trénovacej straty počas trénovania modelu Qwen.	52
3.17	Priebeh trénovacej straty počas trénovania modelu Mistral.	52
3.18	Priebeh validačnej straty počas konca trénovania modelov.	52
3.19	Porovnanie dĺžky potvrdených rozhodnutí a zrušených/zmenených rozhodnutí.	53
4.1	Porovnanie Perplexity score.	59
4.2	Porovnanie BLEU score.	59
4.3	Porovnanie ROUGE-1 score.	60
4.4	Porovnanie ROUGE-2 score.	60
4.5	Porovnanie ROUGE-L score.	60

Zoznam tabuliek

3.1	Výsledné štatistiky hodnoty Perplexity pre jednotlivé modely.	44
3.2	Výsledné štatistiky hodnoty BLEU pre jednotlivé modely.	45
3.3	Výsledné štatistiky hodnoty ROUGE-1 pre jednotlivé modely.	45
3.4	Výsledné štatistiky hodnoty ROUGE-2 pre jednotlivé modely.	45
3.5	Výsledné štatistiky hodnoty ROUGE-L pre jednotlivé modely.	46
3.6	Porovnanie technickej konfigurácie modelov.	51
4.1	Matice zámen pre klasifikáciu (Qwen3-8B).	64
4.2	Matice zámen pre klasifikáciu (dotrénovaný Qwen).	64
4.3	Metriky vyhodnotenia pre zero-shot klasifikáciu modelu Qwen.	64
4.4	Metriky vyhodnotenia pre few-shot klasifikáciu modelu Qwen.	64
4.5	Matice zámen pre dotrénovaný binárny klasifikátor.	65
4.6	Metriky vyhodnotenia pre binárne klasifikátory.	65

1 Úvod

Predkladaná diplomová práca sa zaoberá využitím veľkých jazykových modelov pre analýzu rozhodnutí slovenských súdov. Motivácia pre vypracovanie tejto práce a formulácia jej cieľov sú popísané nižšie.

1.1 Motivácia

Dobrá správa vecí verejných si vyžaduje okrem aktívnej účasti občanov, slobody tlače a rovnováhy moci medzi jednotlivými zložkami vlády, aj zodpovednosť, transparentnosť a správnosť legislatívneho rozhodovania. To častokrát od zákonodarcov vyžaduje veľké množstvo času a hlboké porozumenie problémov, ktoré je potrebné riešiť [1]. V súčasnosti je známe, že dĺžka súdneho konania na Slovensku sa často pohybuje v rozmedzí niekoľkých rokov, pričom jednou z príčin je aj nedostatok sudcov [2].

Okrem toho sú sudcovia menovaní doživotne, čo znamená, že nie je možné voči nim vyvodiť zodpovednosť nejakým demokratickým mechanizmom. Zároveň však majú veľmi dôležitú úlohu v demokratickom ústavnom štáte, ako je ochrana základných práv a slobôd, dohľad nad dodržiavaním zákonov a ústavy, či v neposlednom rade stráženie princípov demokracie. Tento fakt taktiež poukazuje na dôležitosť transparentnosti, kvality a verejnej kontroly rozhodovania súdov [3].

Medzi situácie spojené s činnosťou, resp. nečinnosťou sudcu, ktoré narúšajú princíp právnej istoty, patrí aj odklon od ustálenej judikatúry, alebo odklon od predchádzajúceho rozhodnutia vyšších či najvyšších súdnych autorít [4]. Súdne rozhodnutia by teda mali vykazovať vysokú mieru predvídateľnosti a nemali by sa významne odchyľovať od ustálenej judikatúry. Metódy prediktívnej umelej inteligencie častokrát pracujú s historickými dátami, na základe ktorých sa snažia odhadnúť najpravdepodobnejší výstup pri vložení doposiaľ neznámeho vstupu, čo zaručuje konzistentnosť a to v tomto prípade predstavuje dobrý základ pre ich použitie v právnej oblasti.

Využívanie umelej inteligencie a veľkých jazykových modelov je v právnej praxi zatiaľ skôr zriedkavé, no môže priniesť značné výhody. Jednou z kľúčových oblastí je detekcia anomálií v súdnych rozhodnutiach. Automatizovaná analýza rozsiahlych databáz rozhodnutí dokáže identifikovať štatisticky nezvyčajné vzory, napr. neodôvodnené odchýlky v ukladaní trestov pri skutkovo podobných prípadoch, nekonzistentné aplikovanie právnych noriem alebo náhle zmeny v rozhodovacej praxi konkrétneho sudcu. Môže teda vystupovať ako nezávislá vrstva dohľadu, ktorá môže včas upozorniť na potenciálne pochybenia alebo systémové skreslenia.

Takisto by aplikácia týchto nástrojov mohla priniesť aj priame ekonomické úspory. Prediktívny model dokáže identifikovať odchýlky od štandardných zmluvných formulácií alebo ustálenej rozhodovacej praxe. Ak rozhodnutie nie je významne odchýlené, malo by to znamenať aj menšiu pravdepodobnosť jeho (úspešného) napadnutia, a tým pádom znížiť záťaž na súdny systém pri odvolacích konaniach. Rovnako môže byť prediktívny model použitý, napr. na automatické upozornenie nedostatočného odôvodnenia rozhodnutia ešte pred jeho vydaním, čím predchádza zbytočným odvolaniam a s nimi spojeným nákladom na ďalšie súdne konanie.

Táto práca sa na základe vyššie uvedenej motivácie venuje doladeniu jazykového modelu na korpuse súdnych rozhodnutí s cieľom zlepšiť jeho jazykové schopnosti v právnej doméne a zachytiť vzory rozhodovacej praxe. Výsledný model by mal byť následne schopný identifikovať rozhodnutia, ktoré sa od tejto praxe odchyľujú. Súčasťou práce je taktiež tréning klasifikátora na predikciu úspešnosti odvolania. Takýto nástroj môže byť prínosom pre právnikov aj pre účastníkov konania bez právneho zastúpenia, ktorí zvažujú, či podať odvolanie, ktoré pri neúspechu predstavuje dodatočné finančné náklady.

1.2 Formulácia úlohy

V predkladanej práci overujeme nasledovné hypotézy:

H_1 : Výber východiskového jazykového modelu pre následné dotrénovanie na textoch rozhodnutí slovenských súdov má merateľný vplyv na kvalitu jazykovej reprezentácie slovenského právnického textu, pričom sa predpokladá, že ohodnotenie vybraných jazykových modelov pomocou metrík BLEU, ROUGE a Perplexity odhalí signifikantné rozdiely medzi modelmi a umožní identifikovať kandidátov najvhodnejších pre následnú doménovú adaptáciu.

H_2 : Dotrénovaním vybraného jazykového modelu na korpuse rozhodnutí slo-

venských súdov pre generovanie výroku rozhodnutia na základe odôvodnenia dôjde k štatisticky merateľnému zlepšeniu jazykovej reprezentácie právneho textu v slovenčine, čo sa prejaví vyššími hodnotami metrík BLEU a ROUGE a nižšou Perplexity v porovnaní s východiskovým modelom.

H_3 : Dodatočným dotrénovaním jazykového modelu, adaptovaného na slovenské súdne rozhodnutia, na úlohu klasifikácie úspešnosti odvolania dosiahne model úspešnosť aspoň 80%.

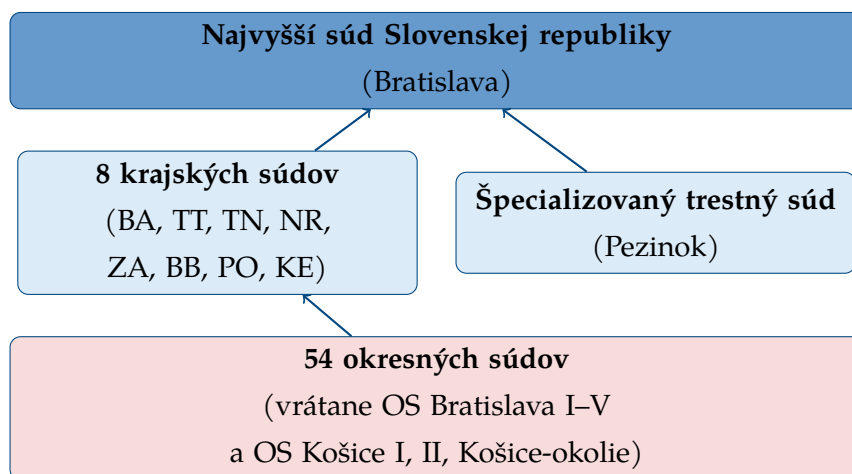
Štruktúra tejto práce je nasledovná. Analýza sa zaoberá štruktúrou súdov v Slovenskej republike, popísaním štúdií venujúcich sa podobnej problematike a súčasným technológiám využívaných v tejto oblasti. Implementácia zahŕňa opis možných aplikácií výstupov práce a ich prínosov, exploratívnu analýzu textového korpusu rozhodnutí slovenských súdov, predspracovanie textových dát, porovnanie existujúcich jazykových modelov pre úlohu generovania výroku rozhodnutia z odôvodnenia, výber modelu, ako aj popis postupu a priebehu tréningu modelov. Vyhodnotenie obsahuje kvantitatívne aj kvalitatívne vyhodnotenie dotrénovaných modelov na testovacích dátach spolu s diskusiou výsledkov a limitáciami aplikácie jazykových modelov v právnej oblasti. Záver zhŕňa celú prácu a načrtná možné rozšírenia práce do budúcnosti. V časti príloh je uvedený odkaz na zdrojový kód vytvorený v rámci práce, resp. systémovú dokumentáciu práce.

2 Analýza

Táto kapitola popisuje teoretické základy pre následnú implementáciu a prehľad štúdií, ktoré sa zaoberali podobnou problematikou v jurisdikciách iných štátov, ale aj na Slovensku.

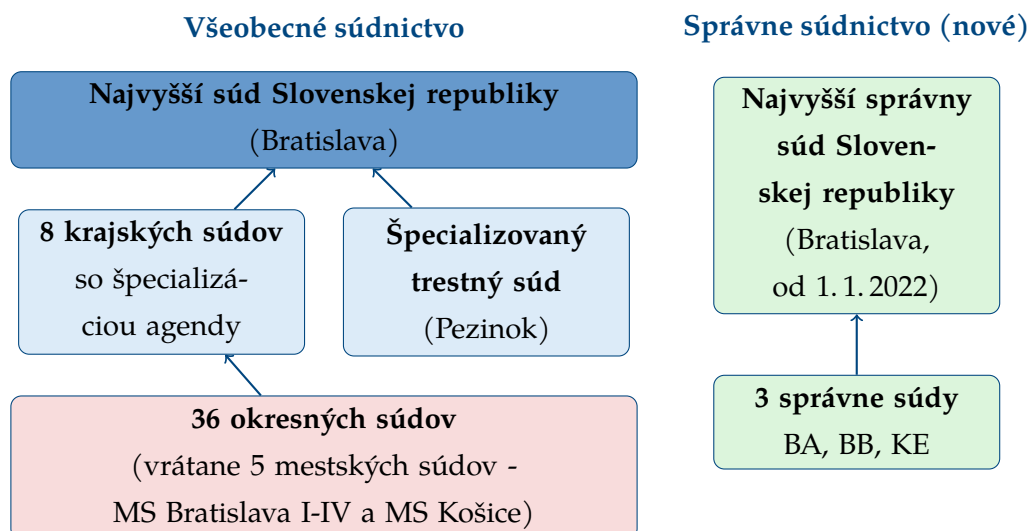
2.1 Štruktúra súdov v SR

Súdna moc je jedna z troch základných štátnych mocí popri zákonodarnej a výkonnej moci. Realizuje sa prostredníctvom rozhodovania súdov. V tejto sekcii je stručne zobrazená štruktúra a hierarchia súdov v SR s porovnaním pred a po reforme súdnej mapy¹, ktorá prebehla v roku 2023. Pre úplnosť je potrebné poznamenať, že na Slovensku existuje aj Ústavný súd Slovenskej republiky so sídlom v Košiciach, ktorý je však nezávislým súdnym orgánom ochrany ústavnosti.



Obrázok 2.1: Schéma hierarchie súdov pred reformou. Zdroj: Vlastné spracovanie.

¹<https://www.justice.gov.sk/sudy-a-rozhodnutia/sudna-mapa/zmeny-v-sudnej-mape/>



Obrázok 2.2: Schéma hierarchie súdov po reforme. Zdroj: Vlastné spracovanie.

2.2 Prehľad súvisiacich štúdií

V posledných rokoch zažívajú veľké jazykové modely ako napr. **GPT**, **LLaMA** alebo Claude svoj vrchol popularity nielen medzi technickými nadšencami, ale aj bežnou populáciou. Autori Ammar a kol. využili takéto modely na predikciu súdneho rozhodnutia [5]. Problémom bolo spracovanie arabského jazyka, v ktorom boli tieto súdne prípady písané. Autori porovnali tri modely a to, LLaMA-7b, GPT-3.5-turbo a JAIS-13b-chat (ten bol trénovaný so špeciálnym zameraním na arabský jazyk), v predikcii rozhodnutia 10 713 súdnych prípadov zameraných na finančné a obchodné spory, ktoré boli získané zo súboru údajov Saudského ministerstva spravodlivosti. Predikcie boli vyhodnotené viacerými metrikami, a to BLEU, ROUGE a GPT score, ako aj ľudským expertom. Suverénne najlepšie hodnotený bol model GPT-3.5-turbo.

V inej publikácii si autori vytvorili vlastné modely zamerané na predikciu rozhodnutí Najvyššieho súdu Filipín pomocou spracovania prirodzeného jazyka a strojového učenia [6]. Dáta pre analýzu boli získané z online dostupných rozhodnutí od roku 1987 do roku 2017, čo spolu predstavovalo 27 492 rozhodnutí a na reprezentáciu textu boli použité metódy spracovania prirodzeného jazyka, najmä model *bag-of-words*, v preklade „vrece slov“, a zoskupenie do *n*-gramov. Takto spracovaný text predstavoval trénovacie dáta pre algoritmy strojového učenia ako lineárna metóda podporných vektorov a klasifikátor náhodného lesa. Najvyššia presnosť týchto modelov sa pohybovala okolo 60%, čo nie je veľmi vysoké číslo, no nakoľko ide o prvú systematickú štúdiu v predpovedaní rozhod-

nutí filipínskeho najvyššieho súdu založenou výlučne na textovom obsahu, je to prijateľný výsledok.

Ako však uviesť takéto predikčné modely do praxe? Autori Almuzaini a Azmi [7] vo svojej publikácii opisujú vytvorenie systému na podporu súdneho rozhodovania, *TaSbeeb* (v preklade „právne zdôvodnenie“), založeného na hlbokom učení, ktorý pozostáva z troch častí a to, anotácia právneho textu, klasifikácia a vyhľadávanie informácií. Takže okrem predikcie rozhodnutia poskytuje aj možnosť sumarizácie právneho textu či klasifikovanie právnej veci (témy) prípadu, čo je veľmi nápomocné pre právnych zástupcov, ale aj bežných občanov, pretože to šetrí čas, zlepšuje pochopenie podobných súdnych prípadov. Pre anotáciu textu bola použitá semi-automatická metóda a podobnosť bola ohodnotená Jaccardovým koeficientom. Predspracovanie textu zahŕňalo odstránenie interpunkčných znamienok, stop-slov, číslic, nearabských a jednopísmenových slov, *stemming*, teda prevod na koreň slova, a algoritmus *Word2Vec*, ktorý umožňuje reprezentovať slová vektormi. Na klasifikáciu boli porovnávané tri modely hlbokého učenia a to **BiLSTM**, **Att-GRU** a BiGRU. Pre zlepšenie klasifikácie autori taktiež tieto modely skombinovali pomocou stacking-u.

Neurónové siete využili aj autori Chaphalkar a kol. [8] v prípadovej štúdii, ktorá sa zaoberá riešením sporov v súvislosti so stavebnými zmluvami v Indii. Spory v stavebníctve často vedú k narušeniu vzťahov medzi stranami, časovým sklzom a zvýšeným nákladom. Autori štúdie tvrdia, že ak by strany vedeli predvídať výsledok sporu, boli by ochotnejšie dohodnúť sa mimosúdne, čím by sa znížili náklady a záťaž na súdny systém. Štúdia sa zameriava na „vnútorné faktory“ (napr. podmienky zmluvy, situáciu na stavbe, dokumenty z arbitráže), ktoré majú vplyv na rozhodnutie v sporoch.

Hlbšiu analýzu faktorov pôsobiacich na rozhodnutie súdov v USA predložili aj autori Conrad a Al-Kofahi [9]. Ich práca zahŕňa automatickú klasifikáciu právnej témy, zhľukovanie podľa nárokov žalobcu aglomeratívnymi, deliacimi a grafovými algoritmami a algoritmom k-Means, ktorý dosiahol najlepšie výsledky, či analýzu vzťahov medzi dĺžkou súdneho konania a výškou nároku, ktorý bol priznaný súdom žalobcovi.

Zatiaľ čo vyššie uvedené štúdie sa zameriavali skôr na civilné spory, kolektív autorov zo Záporožskej národnej technickej univerzity sa venoval trestnému konaniu, konkrétne v prípade krádeže, pretože ide o veľmi častý typ trestného činu na Ukrajine [10]. Podľa čl. 65 Trestného zákona Ukrajiny súd ukladá trest v medziach ustanovených v sankcii osobitnej časti Trestného zákona, pričom zohľadňuje: stupeň závažnosti trestného činu, osobu páchatela, poľahčujúce okol-

nosti. V súlade s tým boli zvolené nasledujúce vstupné premenné pre fuzzy rozhodovací systém: závažnosť činu, osoba páchatela, poľahčujúce okolnosti, miera poľahčujúcich okolností, priťažujúce okolnosti, miera priťažujúcich okolností a neutralita sudcu. Výstupom je počet rokov odňatia slobody páchatela v rozmedzí jedného až šiestich rokov. Autori navrhli dva varianty systému na podporu rozhodovania, založené na algoritmoch Mamdani a Sugeno. Obidva systémy sa v porovnaní s rozsudkom súdu líšili vo výstupnej premennej maximálne o niekoľko mesiacov.

Výskumníci z dvoch ruských univerzít taktiež skúmali rozhodnutia súdu pri deliktach, konkrétne týkajúcich sa nerešpektovania výzvy verejného činiteľa. Ich cieľom bolo zlepšenie digitalizácie administratívneho konania, podpora IT infraštruktúry procesov elektronického riadenia a elektronického štátu prostredníctvom tvorby systémov umelej inteligencie v oblasti práva, ako aj zistiť okolnosti ovplyvňujúce prísnosť trestu [11]. Autori vyvinuli metódy automatického usporiadania textov súdnych rozhodnutí, pričom úlohou algoritmu je automatizovane extrahovať kľúčové entity (ukazovatele) z pološtruktúrovaných dát pre ďalšiu tvorbu prognostických a deskriptívnych modelov na analýzu účinnosti kontrolných a dozorných činností. Pre spracovanie súdnych rozhodnutí a odvolaní bola použitá kombinácia TF-IDF a latentnej sémantickej analýzy a zoskupenie do n-gramov. Jednotlivé fakty, ktoré sa frekventovane vyskytovali v dokumentoch kategorizovali do dvoch tried, a to „skutková okolnosť“ a „procesná skutočnosť“. Autori okrem iného upozornili na to, že zrušené rozhodnutia súdov prvého stupňa spravidla obsahujú podrobnejšiu motivačnú časť než tie, ktorými bol uložený správny trest, avšak LSA ukazuje, že v tomto prípade súdy druhého stupňa venujú väčšiu pozornosť preverovaniu splnenia procesných náležitostí v predchádzajúcich štádiách konania o správnom práve. Na anotáciu súdnych rozhodnutí bola vyvinutá klasifikácia subjektov (typ tokenov), ktorá odráža právne procesy. Takto anotované dáta potom použili na tvorbu modelu rozhodovacieho stromu pre predikciu výšky trestu vo forme väzby alebo pokuty za priestupok, ktorý dosiahol významnú výkonnosť s hodnotou ROC krivky 90%. Výsledky boli napokon validované a interpretované odborníkmi.

Trochu z iného pohľadu analyzovali trestné činy v izraelskej štúdii zameranej na dôveryhodnosť (kredibilitu) obete sexuálneho násillia [12]. K ich motivácii prispela opakovaná kritika verejnosti, ktorá v mnohých krajinách poukazovala na nedostatočné reakcie súdov na sexuálne násillie, útoky a znásillenia. Analyzovaných bolo 855 rozhodnutí súdov z rokov 1990 až 2021 vo veci sexuálneho násillia, pričom veľkou výzvou bolo spracovanie hebrejského jazyka. Anotá-

ciu textu uskutočnilo päť ľudských expertov a na extrakciu textu použili nástroj SPIKE². S takto predspracovanými textovými dátami mali autori za cieľ vytvoriť model, ktorý by klasifikoval výpoveď obete do jednej zo štyroch, resp. deviatich kategórií podľa stupňa dôveryhodnosti, a to od jednoznačnej dôveryhodnosti po nedôveryhodnosť alebo irelevantnosť. Na túto klasifikáciu doladili existujúci hebrejský jazykový model AlephBERT pridaním výstupnej vrstvy špecifickej pre danú úlohu nad predtrénovaný model. Ten dosiahol úspešnosť 77% pri štyroch kategóriách a 64% pri deviatich kategóriách. Tieto výsledky dosahujú úroveň porovnateľnú s ľudským označovaním, čo potvrdzuje aj skóre medzianotátorskej dohody (IAA).

Predikciou rozhodnutia súdu v oblasti sexuálneho násillia sa zaoberá aj thajská štúdia, ktorá sa usiluje pomôcť obetiam sexuálneho násillia prostredníctvom vytvorenia modelu, ktorý by presne klasifikoval trestný čin a úmysel páchatel'a, čím by obeti uľahčili nahlásenie trestného činu, uplatnenie svojich práv a domáhanie sa spravodlivosti [13]. V štúdiu sú porovnávané dva typy modelov, a to predtrénované jazykové modely založené na transformeroch (T-PTLMs), a model Word2Vec spolu s modelom SVM. Predtrénovaný algoritmus Word2Vec bol použitý na transformáciu fragmentov rozhodnutí na vektorové reprezentácie na vytvorenie vektorovej reprezentácie slov. Tieto boli ďalej použité na trénovanie modelu SVM na klasifikáciu fragmentov rozhodnutí Najvyššieho súdu podľa úmyslu páchatel'a trestného činu. Modely založené na transformeroch, konkrétne XLM-RoBERTa-base (XLM-RBASE), XLM-RoBERTa-large (XLM-RLARGE), a WangchanBERTa (WangchanBERBASE) prispôsobené pre thajský jazyk, dosiahli lepšie výsledky pri multitriednej klasifikácii a to F1-skóre rovné takmer 80%. Štúdia taktiež demonštruje potenciál chatbotov s umelou inteligenciou na podporu obetí sexuálneho násillia a prispieva k boju proti všadeprítomnému problému sexuálneho násillia.

Nie je to jediný príspevok v Thajsku zaoberajúci sa predikciou súdnych rozhodnutí. Iní thajskí výskumníci sa zamerali na porovnávanie viacerých techník na extrakciu textu z rozhodnutí, ako Doc2Vec, Word2Vec, Count Vectorizer, and TF-IDF Vectorizer v spojení s metódami strojového učenia, a to SVM, Naivný Bayes, klasifikátor náhodného lesa, rozhodovací strom a logistická regresia [14]. Na vyhodnotenie modelov boli použité metriky presnosť a návratnosť, pri ktorých sa zohľadnilo mikro aj makro priemerovanie, spolu s celkovou úspešnosťou a F1 skóre. Autori sa museli popasovať s náročnejším predspracovaním dát, resp. tokenizáciou slov, nakoľko v thajskom jazyku nie sú medzery medzi slovami, pri-

²<https://spike.apps.allenai.org>

čom využili aj knižnice jazyku Python ako napr. PyThaiNLP.

Doposiaľ popísané výskumy boli vykonané v rozličných častiach sveta, no ako je to na Slovensku? Aj u nás existuje niekoľko štúdií skúmajúcich súdne rozhodnutia pomocou technológií využívajúcich umelú inteligenciu. Výskumníci z Univerzity Pavla Jozefa Šafárika v Košiciach (UPJŠ) si v jednej zo svojich štúdií vytýčili dva ciele, a to vytvoriť klasifikačný model, ktorý by dokázal predikovať verdikt súdu z odôvodnenia, a identifikovať významné pojmy v odôvodneniach rozsudkov, ktoré úzko súvisia s výsledkami rozsudkov v trestnom konaní (nevina alebo vina) [15]. Využili na to 226 500 rozhodnutí slovenských súdov vo veci trestného činu za roky 2016 až 2020 vo formáte **JSON**, z ktorých extrahovali potrebné informácie, a to úvod (názov súdu, mená sudcov a obhajcov), výrok (verdikt súdu), odôvodnenie a poučenie. Viac ako polovica z týchto rozhodnutí však neobsahovala časť odôvodnenia, preto ju autori nakoniec nezahrnuli pri tvorbe modelu. Autori aplikovali už niekoľkokrát spomenuté algoritmy Word2Vec a Doc2Vec, ktoré vynikajú svojou schopnosťou zakódovať kontext daných slov a dokumentov, čo bolo prínosné pre ich druhý cieľ. Na predikciu boli zvolené metódy ako logistická regresia, **SVM**, klasifikátor náhodného lesa a rôzne typy neurónových sietí. Najvyššiu úspešnosť dosiahla kombinácia Word2Vec a **TCN**, a to vyše 99%, avšak ani ostatné metódy nezaostávali a väčšina z nich dosiahla úspešnosť viac ako 95%. Najrelevantnejšie pojmy pre verdikt súdu znázornili v tabuľkách, pričom ich zoradili podľa percentuálneho zastúpenia v dvoch typoch verdiktu. Pre verdikt „vinný“ boli relevantné hlavne pojmy ako: voľba, slobodný, dobrovoľne a súhlasiť, a pre verdikt „nevinný“ napr. obžalovať, svedkyňa, pojednávanie, dôvod a jednoznačne.

Iný článok z UPJŠ bol venovaný analýze rozhodnutí Najvyššieho súdu SR pomocou metód strojového učenia a formálnej pojmovej analýzy na preskúmanie vzťahov medzi atribútmi súdnych rozhodnutí [16]. Dva ďalšie príspevky, taktiež od autorov z UPJŠ, boli orientované na extrakciu kľúčových fráz z rozhodnutí slovenských súdov nekontrolovanými metódami strojového učenia, ako napr. **TF-IDF**, YAKE! a algoritmus váženého PageRanku založeného na grafe [17], ako aj odkazov na právne predpisy pomocou regulárnych výrazov a Levenshteinovej metriky podobnosti [18]. Takáto extrakcia informácií im má pomôcť pri ich dlhodobom celi, vytvorení systému na vyhľadávanie a analýzu súdnych rozhodnutí, ktorý sa bude podobáť kanadskému systému CanLII. Novší článok od autorov z UPJŠ [19] sa venuje automatickému vytvoreniu tezauru právnickej terminológie. V procese jeho konštrukcie sú využité aj jazykové modely SlovakBERT a LLaMa 3.1-7B konkrétne v kroku predikcie, či daný odstavec právnického textu

obsahuje definíciu, ktorú by bolo vhodné následne extrahovať, napr. regulárnymi výrazmi alebo identifikáciou slovných druhov vo vete, a zaradiť do tezauru. Anotáciou a extrakciou právnych pojmov z rozhodnutí súdu najvyššej inštalácie sa zaoberali autori aj v Českej republike v rámci projektu hodnotenia významu súdnych rozhodnutí v českom práve [20].

Žiadna zo štúdií slovenských autorov sa však nevenovala konkrétne predikcii v kontexte právneho poriadku Slovenskej republiky s využitím veľkých jazykových modelov. Predkladaná práca tak vyplňa túto medzeru, a súčasne môže slúžiť ako dobrý základ pre budúci výskum v tejto oblasti.

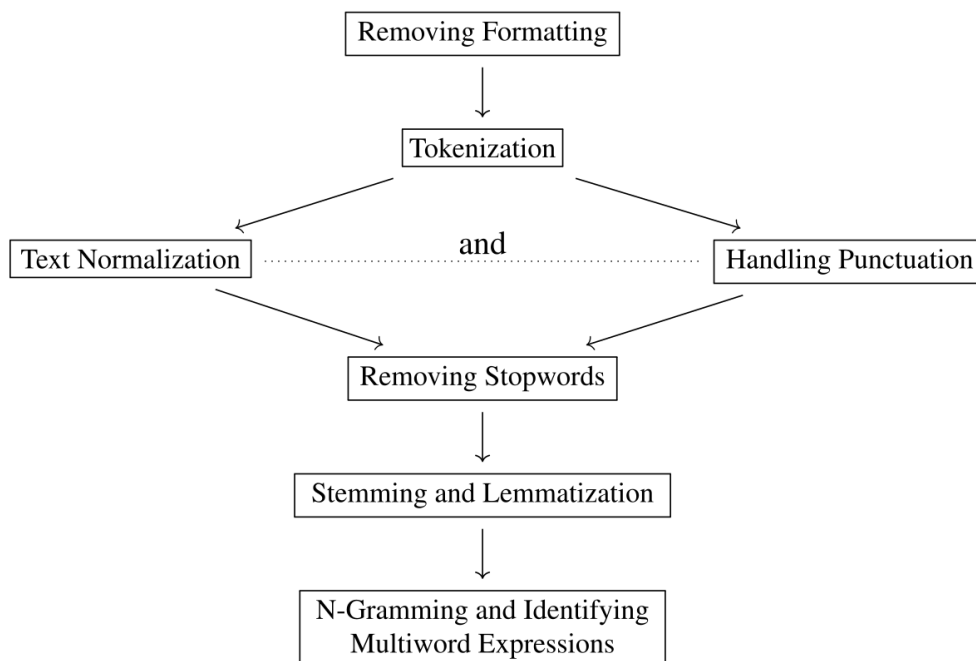
2.3 Metódy a technológie

V tejto podkapitole si priblížime metódy a technológie relevantné pre riešenie úlohy tejto práce.

2.3.1 Spracovanie prirodzeného jazyka

Pri spracovaní a analýze textu, tak ako aj pri rôznych iných typoch dát, je veľmi dôležitým krokom predspracovanie, aby bol text pripravený slúžiť ako vstup pre modely umelej inteligencie. Autori vo viacerých štúdiách preukázali, že predspracovanie textu môže zvýšiť presnosť, a to aj o viac ako 20 percent, napr. pri analýze sentimentu údajov sociálnych médií [21]. Na obr. 2.3 môžeme vidieť schému bežného poradia techník predspracovania textu.

V prípade aplikácie tradičných metód strojového učenia je v prvom rade potrebné odstránenie formátovania, teda prevod na „raw“ text. Ďalším veľmi dôležitým krokom je tokenizácia, ktorá rozdelí text na tzv. tokeny, teda základné lexikálne jednotky. V niektorých prípadoch tokeny znamenajú celé slová, no v slovenčine, a všeobecne v jazykoch s bohatou morfológiou, môže byť vhodné rozdeliť slová ďalej, napríklad na koreň, predpony a prípony. Ďalej nasleduje normalizácia textu, čiže mapovanie výrazov na jednotný tvar, čo je obzvlášť dôležité v textoch s veľkým množstvom neštandardných zápisov, ako sú napríklad príspevky na sociálnych sieťach. Odstránenie interpunkcie umožní zamerať sa na slová v textovom korpuse, zjednodušuje analýzu textu a zlepšuje výkonnosť vyhľadávania informácií. Nemenej podstatným krokom je odstránenie tzv. „stop slov,“ teda neplnovýznamových slov, čo sú v slovenčine spojky, predložky, častice, zámená a pomocné slovesá, čím sa zníži rozmer príznakového priestoru. Tak tiež je vhodné previesť slová na základný tvar (lematizácia) alebo na koreň slova (stemming). Posledným krokom v schéme je zoskupenie do n-gramov a iden-



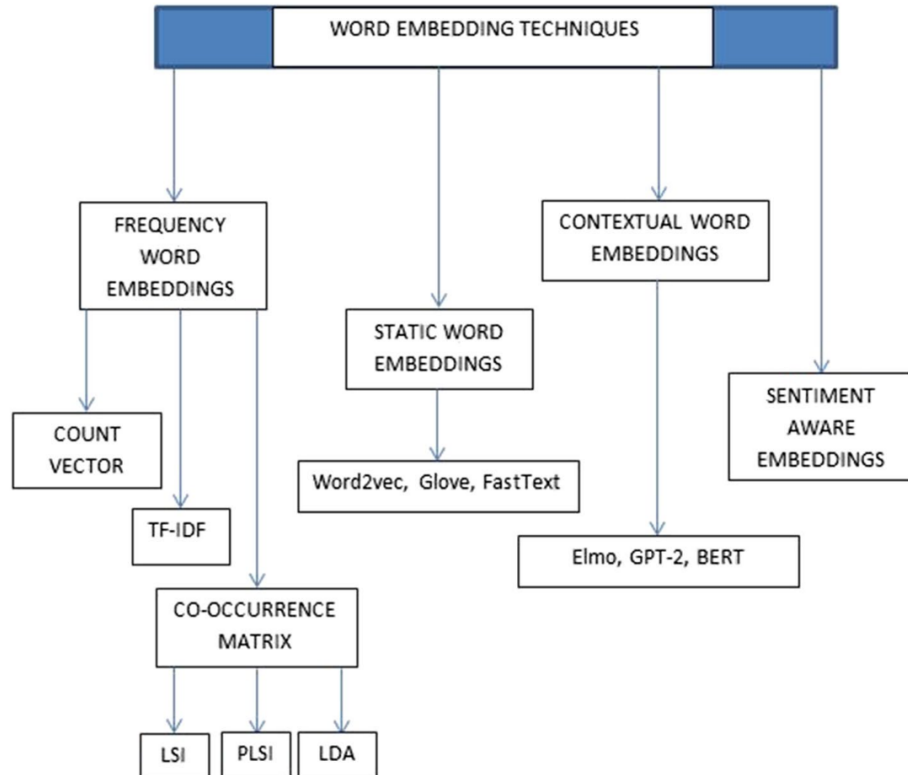
Obrázok 2.3: Schéma predspracovania textu. Zdroj: [22].

tifikácia viacslovných výrazov [22]. Pri tréňovaní veľkých jazykových modelov sa ako vstup používa pôvodný súvislý text, ktorý je tokenizovaný tokenizérom konkrétneho modelu.

Následne je možné pristúpiť k tzv. „embeddingu,” t.j. transformácii textu do numerickej, resp. vektorovej reprezentácie. Ako je znázornené na obr. 2.4, existujú rôzne typy embeddingov, ktoré môžu byť rozdelené do štyroch kategórií, a to: na základe frekvencie výskytu slov; statické; na základe kontextu; na základe sentimentu.

Jedným z najjednoduchších a efektívnych prístupov je „bag-of-words“ (BoW) model, inak nazývaný aj „count vector.“ V reprezentácii BoW je každá veta v dokumente reprezentovaná vektormi slov, kde každá dimenzia zodpovedá frekvencii termínov pre každé jedinečné slovo v danom textovom korpuse. Celkový počet dimenzií sa teda rovná celkovému počtu jedinečných slov [23]. Tento prístup ignoruje poradie slov a považuje každé slovo za nezávislé, keďže reprezentuje slová ako neusporiadanú kolekciu. To znamená, že nedokáže rozpoznať vzťahy medzi slovami vo vete alebo dokumente, a teda nie je schopný zachytiť sémantický význam a kontext slov. To predstavuje obmedzenie v úlohách analýzy sentimentu textu alebo jazykového prekladu. Tento model je však veľmi prospešný napr. pri vyhľadávaní informácií alebo v aplikáciách počítačového videnia [24].

Ďalší prístup patriaci do skupiny embeddingov založených na frekvencii výskytu slov je „Term Frequency and Inverse Document Frequency.“ TF-IDF je šta-



Obrázok 2.4: Rôzne techniky embeddingu. Zdroj: [23].

tistická metóda, ktorá vytvára maticu „term-dokument“, čím vyhodnocuje dôležitosť slova v dokumente vzhľadom na kolekciu dokumentov alebo korpus. Dôležitosť slova narastá priamo úmerne k počtu jeho výskytov v dokumente a zároveň klesá nepriamo úmerne s počtom dokumentov, v ktorých sa slovo vyskytuje. Frekvencia termu TF je definovaná ako:

$$TF_{t,d} = \frac{n_{t,d}}{\text{počet termov v dokumente}} \quad (2.1)$$

kde n je početnosť termu t v dokumente d . Inverzná frekvencia dokumentu IDF je definovaná ako:

$$IDF_t = \log \frac{\text{počet dokumentov}}{\text{počet dokumentov obsahujúcich term 't'}} \quad (2.2)$$

Vynásobením rovníc 2.1 a 2.2 dostaneme celkové **TF-IDF** skóre pre každý term:

$$(TF-IDF)_{t,d} = TF_{t,d} * IDF_t \quad (2.3)$$

Týmto spôsobom **TF-IDF** zvýrazňuje slová, ktoré sú špecifické pre konkrétny dokument, a potláča tie, ktoré sa často opakujú naprieč celým korpusom [24].

Matica „term-dokument“ je však vysokorozmerná a veľmi riedka. Tento problém riešia metódy založené na faktorizácii matic. Ich cieľom je redukovať počet

dimenzií vstupnej matice, a tým komprimovať najdôležitejšie informácie do niekoľkých nízkorozmerných matíc, z ktorých možno vstup približne alebo presne obnoviť. V prvom rade ide o „latentnú sémantickú analýzu“ (**Latent Semantic Analysis**), resp. „latentné sémantické indexovanie“ (**Latent Semantic Indexing**). Aplikáciou „dekompozície singulárnych hodnôt“ (**SVD**) na **TF-IDF** sa vytvorí latentný sémantický priestor, ktorý zachytáva väčšinu variability v texte. V tomto priestore sú jednotlivé dimenzie reprezentované ako lineárne kombinácie pôvodných **TF-IDF** dimenzií. Takýto prístup pomáha riešiť problém synonymie, teda situáciu, kedy existuje viacero spôsobov, akými možno popísať ten istý predmet alebo jav, ako aj problém polysemie, kedy jedno slovo môže mať viacero významov. Výsledkom je efektívna kompresia aj pri veľkých textových korpusoch, ktorá môže prispieť aj k zníženiu šumu. Nevýhodou je nutnosť indexovania celého korpusu nanovo po pridaní alebo upravení dokumentu, preto nie je veľmi vhodný pre dynamicky meniaci sa korpus [23]. Pravdepodobnostné **LSI** predpokladá, že vzhľadom na doposiaľ neznámu tému „p“ sú dokument „d“ a slovo „ w_n “ podmiennečne nezávislé a vznikli nezávisle od seba. Hlavným konceptom tohto štatistického prístupu je vytvorenie pravdepodobnostného modelu [25].

Najnovším prístupom v tejto skupine je „latentná dirichletova alokácia“ (**LDA**), ktorá predstavuje bayesovskú verziu **PLSI**. Pomocou **LDA** možno extrahovať z korpusu dokumentov témy interpretovateľné človekom, kde každú tému charakterizujú slová, ktoré sú s ňou najpravdepodobnejšie spojené [24]. Pri porovnaní použitia **LDA**, **LSI** a **PLSI** v kombinácii s **SVM** pre klasifikáciu, dosahovala **LDA** najvyššie F1-skóre [26]. Aplikácia metódy **LDA** na klasifikáciu právnych dokumentov, konkrétne pracovných zmlúv, volebných petícií, listín a stanov spoločností, taktiež viedla k vysokej presnosti a optimálnemu výkonu [27].

Populárnym a rozšíreným spôsobom embeddingu je aj „Word2Vec“ patriaci do skupiny statických embeddingov. Word2Vec využíva dvojvrstvovú doprednú neurónovú sieť. Ako naznačuje názov, tento model transformuje slovo na vektor, ktorý je hustý a nízkodimenzionálny v porovnaní s veľkosťou slovníka. Vektorové reprezentácie slov, ktoré sa model Word2Vec naučí, nesú sémantický význam. Embeddingy založené na kontexte, ako napr. „BERT“ a „GPT-2“, vychádzajú z predpokladu, že význam slova je určovaný kontextom okolitého textu [24].

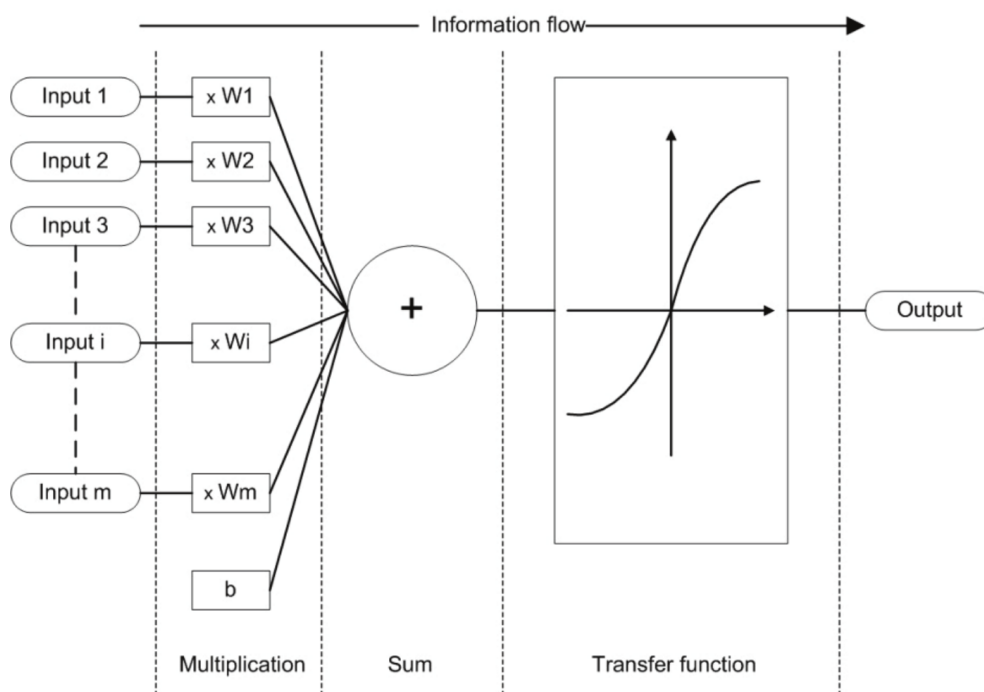
Žiaden z vyššie popísaných prístupov však úplne nerieši sentiment textu, preto sa niektorí autori snažia kombinovať sémantický embedding s tzv. „emocionálnym“ embeddingom založeným na emocionálnom lexikóne. Takto vytvorené hybridné vektorové reprezentácie slov sa ukázali ako efektívne pre klasifikáciu sentimentu a viedli k zvýšeniu úspešnosti klasifikácie [28].

2.3.2 Neurónové siete

Po prevode textu do numerickej reprezentácie naň môžeme aplikovať rôzne algoritmy strojového učenia. Právne texty predstavujú pomerne rozsiahle textové dáta so silnými sekvenčnými závislosťami. Pre spracovanie takýchto veľkých dát je vhodné použiť prístupy založené na hlbokom učení a neurónových sieťach pre porozumenie kontextu, extrakciu vzťahov alebo sumarizáciu. Umelé neurónové siete sa snažia simulovať štruktúru a funkcie biologických neurónových sietí prostredníctvom matematického modelu. Ich základným stavebným prvkom je matematická funkcia, ktorá predstavuje neurón, a má tri jednoduché kroky [29], ktoré môžeme vidieť znázornené na obr. 2.5:

1. násobenie jednotlivých vstupov váhami w_i ;
2. sčítanie týchto výsledkov a biasu b ;
3. aktivácia tohto súčtu pomocou aktivačnej funkcie.

Keď pospájame veľké množstvo takýchto jednoduchých funkcií, vznikne komplexná neurónová sieť zložená z troch typov vrstiev neurónov a to: vstupná vrstva; skrytá vrstva; výstupná vrstva. Obr. 2.6 znázorňuje popísaný princíp.



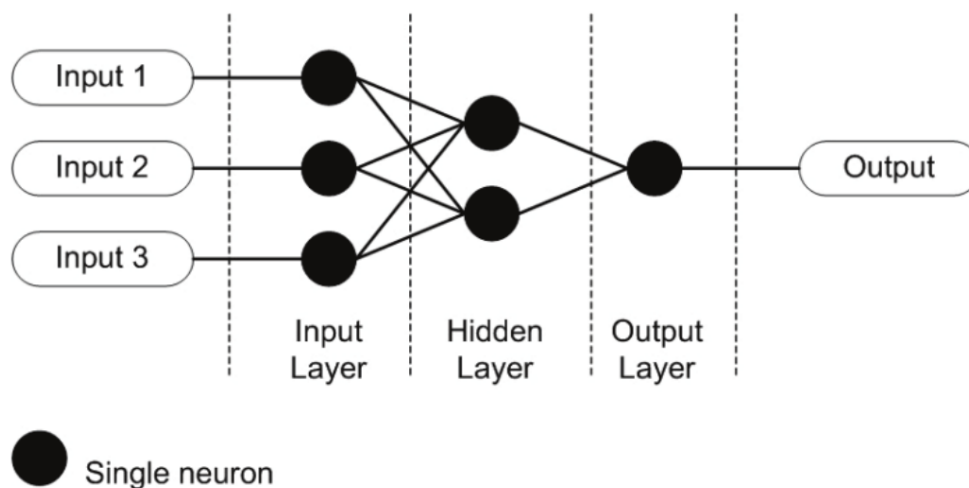
Obrázok 2.5: Princíp fungovania umelého neurónu. Zdroj: [29].

Matematicky je možné vyjadriť to nasledovným spôsobom:

$$y(k) = F\left(\sum_{i=0}^m w_i(k) \cdot x_i(k) + b\right) \quad (2.4)$$

Kde:

- $x_i(k)$ je vstupná hodnota v diskretnom čase k ;
- $w_i(k)$ je váha v diskretnom čase k ;
- b je bias;
- F je aktivačná funkcia;
- $y(k)$ je výstupná hodnota v diskretnom čase k .



Obrázok 2.6: Princíp fungovania umelej neurónovej siete. Zdroj: [29].

Aktivačné funkcie

Presnosť neurónovej siete je závislá od počtu vrstiev, ale takisto aj od typu aktivačnej funkcie použitej v neurónoch. Aktivačné funkcie, inak nazývané aj prahové funkcie (threshold function) alebo prenosové funkcie (transfer function), majú za úlohu transformovať vstupný signál na výstupný signál, ktorý slúži ako vstup pre ďalšiu vrstvu v neurónovej sieti [30]. Teraz si priblížime niektoré typy aktivačných funkcií.

Triviálnou aktivačnou funkciou je tzv. **binárna kroková funkcia** (binary step function). Tá definuje prahovú hodnotu, ktorá rozhoduje o tom, či hodnota lineárnej transformácie musí aktivovať neurón alebo nie. Ak je vstup do aktivačnej

funkcie vyšší ako daný prah, neurón sa aktivuje. V opačnom prípade, t.j. vstup do aktivačnej funkcie je nižší alebo rovný danému prahu, sa neurón deaktivuje, čo znamená, že nie je posunutý ako vstup do ďalšej vrstvy. Táto funkcia má však obmedzenia, ako napr. nemožnosť jej použitia pri multitriednej klasifikácii na výstupnej vrstve alebo fakt, že derivácia tejto funkcie je rovná nule, čo prináša prekážky pri spätnom šírení signálu (back propagation) [30].

Tento problém rieši **lineárna aktivačná funkcia**, ktorá je priamo úmerná vstupnej hodnote. V tomto prípade je derivácia funkcie rovná konštantnej hodnote, čo znamená, že pri spätnom šírení bude faktor aktualizácie (updating factor) rovnaký pre všetky neuróny. Aj keď je táto funkcia ideálna pre interpretovateľnosť a jednoduché úlohy, v neurónových sieťach nepomôže v znížení chyby klasifikácie [30].

Jednou z najpoužívanejších funkcií je **logistická funkcia**, ktorá transformuje hodnoty do intervalu $(0, 1)$. Je definovaná ako: $f(x) = \frac{1}{1+e^{-x}}$. Táto funkcia nie je symetrická okolo nuly, a teda nie všetky výstupné hodnoty budú mať rovnaké znamienko. Podobnou funkciou je **hyperbolická tangenciálna funkcia**, avšak tá je symetrická vzhľadom na počiatok a jej výstupné hodnoty sa pohybujú v intervale $(-1, 1)$. Možno ju vyjadriť ako: $f(x) = \frac{2}{1+e^{-2x}} - 1$.

Funkcia **softmax** predstavuje kombináciu viacerých logistických funkcií. Keďže výstupom z logistickej funkcie je hodnota z intervalu $(0, 1)$, takáto hodnota môže byť použitá ako pravdepodobnosť pre konkrétny dátový bod. To predstavuje dobrý základ pre použitie funkcie softmax pri multitriednej klasifikácii. Matematicky sa vypočíta na základe rovnice:

$$P_i = \frac{e^i}{\sum_j e^j} \quad (2.5)$$

kde P_i označuje pravdepodobnosť i -tej kategórie, e_i označuje príslušnú hodnotu výstupu i -tej kategórie a j označuje celkový počet kategórií [31].

Rectified Linear Unit (ReLU) sa odlišuje tým, že nie všetky neuróny musia byť aktivované zároveň. Funkcia má tvar $f(x) = \max(0, x)$, čo implikuje, že neurón sa aktivuje len vtedy, ak je výstup lineárnej transformácie väčší ako nula. Obmenou ReLU je tzv. **Leaky Rectified Linear Unit (LReLU)**, ktorá je daná predpisom:

$$f(x) = \begin{cases} \alpha x, & \text{pre } x < 0; \quad \alpha > 0 \\ x, & \text{pre } x \geq 0 \end{cases} \quad (2.6)$$

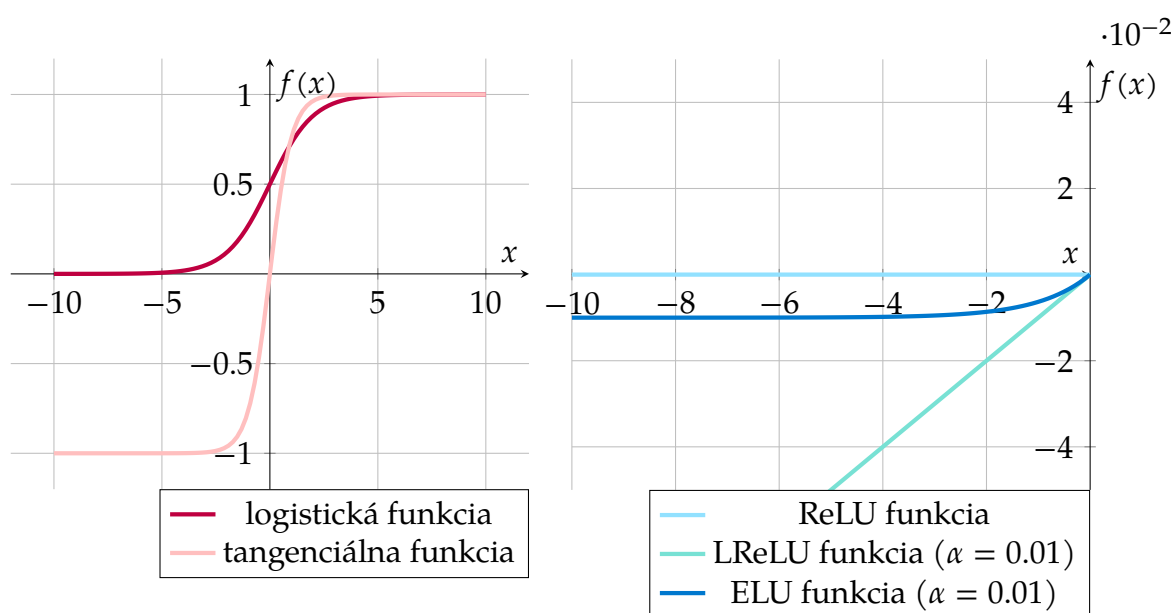
pričom hyperparameter α je veľmi malé číslo (napr. 0.01), čo značí, že namiesto nuly je výstupnou hodnotou veľmi malé číslo závislé od x . **Parametric Rectified**

Linear Unit (PReLU) sa líši od **LReLU** len v tom, že parameter α je aktualizovaný pri spätnom šírení v procese učenia, čím je možné optimalizovať túto hodnotu [32].

Vylepšením aktivačnej funkcie **ReLU** je **Exponential Linear Unit (ELU)**, ktorá znižuje chybu klasifikácie a zrýchľuje proces učenia. Keďže hodnoty neurónov s aktivačnou funkciou **ReLU** sú nezáporné, ich priemerná aktivácia je väčšia ako nula, čo môže v nasledujúcich vrstvách neurónovej siete spôsobovať tzv. „bias shift“, t.j. zmenu strednej hodnoty neurónu v dôsledku aktualizácie váh, ktoré vedú k osciláciám a bránia učeniu [33]. **ELU** má za cieľ riešiť tento problém, a je definovaná ako:

$$f(x) = \begin{cases} x & \text{pre } x > 0 \\ \alpha (e^x - 1) & \text{pre } x \leq 0; \quad \alpha > 0 \end{cases} \quad (2.7)$$

To znamená, že jej negatívne hodnoty posúvajú strednú hodnotu neurónu bližšie k nule. Vizualizácia s porovnaním popísaných funkcií je znázornená na obr. 2.7.



Obrázok 2.7: Vizualizácia aktivačných funkcií. Zdroj: Vlastné spracovanie.

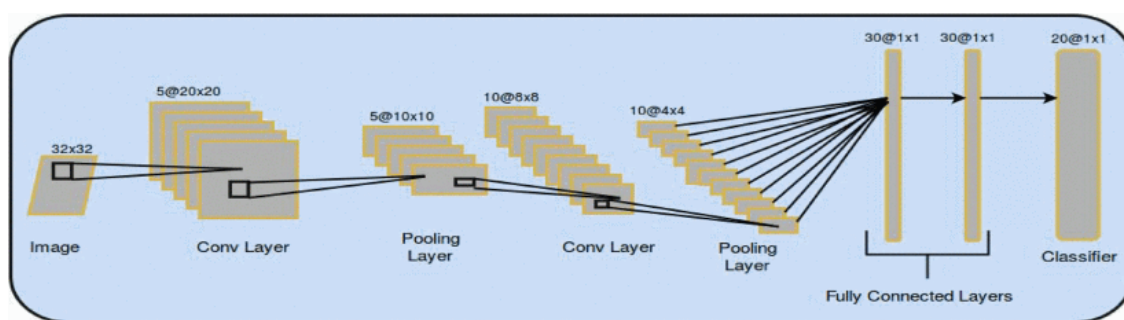
Architektúry neurónových sietí

Teraz sa pozrieme na neurónové siete ako celok. Rozlišujeme dva typy neurónových sietí, a to **dopredné (FNN)** a **rekurentné (RNN)**.

Špecifickým typom doprednej neurónovej siete je **konvolučná neurónová sieť (CNN)**, ktorá nachádza uplatnenie najmä pri úlohách počítačového videnia. **CNN**

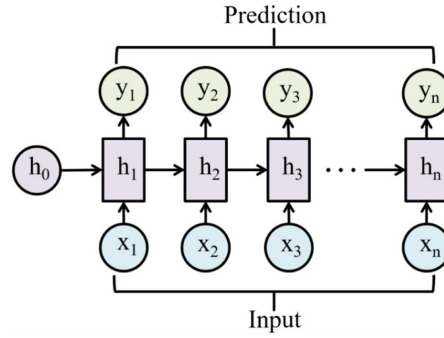
využívajú vrstvy s konvolučnými filterami, inak nazývanými ako kernelové funkcie, ktoré sa aplikujú na lokálne príznaky [34]. Aj keď pôvodne boli CNN určené primárne pre spracovanie obrazu, dnes sú užitočné aj pri spracovaní textu. CNN využíva štyri kľúčové koncepty naplno využívajúc potenciál a výhody prirodzených signálov, a to: lokálne spojenia; zdieľané váhy; združovanie (pooling); používanie mnohých vrstiev. Typická architektúra pozostáva zo série etáp. Niekoľko prvých etáp pozostáva z dvoch typov vrstiev, a to konvolučnej vrstvy a združovacej (pooling) vrstvy. Jednotky v konvolučnej vrstve sú usporiadané do príznakových máp (feature maps), pričom každá jednotka je spojená s lokálnymi výrezmi v mapách príznakov predchádzajúcej vrstvy prostredníctvom súboru váh nazývaného filter (filter bank). Výsledok tohto lokálneho váženého súčtu je potom vstupom pre nelineárnu aktivačnú funkciu, ako napr. už spomínanú ReLU [35].

CNN sa líši od ostatných neurónových sietí tým, že neurón skrytej vrstvy (hidden layer) je pripojený len k podmnožine neurónov v predchádzajúcej vrstve. Toto riedke prepojenie umožňuje sieti učiť sa vlastnosti (príznaky) implicitne. Na obr. 2.8 je zobrazená jej základná architektúra so striedajúcimi sa konvolučnými a pooling vrstvami. Zvýraznené malé políčka predstavujú receptívne oblasti. Spojenia znázorňujú implicitné hierarchické učenie príznakov. Túto architektúru je možné ďalej rozvíjať, pričom medzi populárne typy architektúr patria „LeNet“, „AlexNet“, „GoogleNet“, či „VGGNet“ [36].



Obrázok 2.8: Základná architektúra CNN. Zdroj: [36].

Rekurentné neurónové siete narozdiel od dopredných obsahujú rekurentné spojenia, ktoré umožňujú cirkuláciu informácií v rámci siete, ako možno vidieť na obrázku 2.9. Obojsmerné rekurentné neurónové siete (BiRNN) rozširujú tradičnú architektúru tým, že spracúvajú sekvenciu súčasne smerom vpred aj vzad. Vďaka tomu má model prístup nielen k predchádzajúcim, ale aj k nasledujúcim informáciám, čo zvyšuje jeho presnosť pri úlohách, kde je dôležité porozumieť celému kontextu, minulému aj budúcemu, ako napr. analýza časových radov pri predikcii cien akcií [37].



Obrázok 2.9: Schéma architektúry RNN. Zdroj: [37].

V každom časovom kroku t prijíma RNN vstupný vektor $\mathbf{x}_t$ a aktualizuje svoj skrytý stav $\mathbf{h}_t$ na základe nasledujúcej rovnice:

$$\mathbf{h}_t = \sigma_h(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h), \quad (2.8)$$

kde $\mathbf{W}_{xh}$ je matica váh medzi vstupnou a skrytou vrstvou; $\mathbf{W}_{hh}$ je matica váh rekurentného spojenia; $\mathbf{b}_h$ je vektor biasov a σ_h je aktivačná funkcia, zvyčajne hyperbolický tangens ($\tanh$) alebo ReLU. Výstup v každom časovom kroku t je daný rovnicou:

$$\mathbf{y}_t = \sigma_y(\mathbf{W}_{hy}\mathbf{h}_t + \mathbf{b}_y), \quad (2.9)$$

kde $\mathbf{W}_{hy}$ je matica váh medzi skrytou a výstupnou vrstvou; $\mathbf{b}_y$ je vektor biasov a σ_y je aktivačná funkcia výstupnej vrstvy.

Tu však nastáva problém so zánikom, resp. explodovaním gradientu pri spätnej propagácii. To má za následok nestabilitu a sťaženie tréningu siete, najmä pri učení dlhodobých závislostí. Skrytý stav v čase t môžeme matematicky popísať takto:

$$\mathbf{h}_t = \sigma_h(\mathbf{W}_{xh}\mathbf{x}_t + \mathbf{W}_{hh}\sigma_h(\mathbf{W}_{xh}\mathbf{x}_{t-1} + \mathbf{W}_{hh}\mathbf{h}_{t-2} + \mathbf{b}_h) + \mathbf{b}_h). \quad (2.9)$$

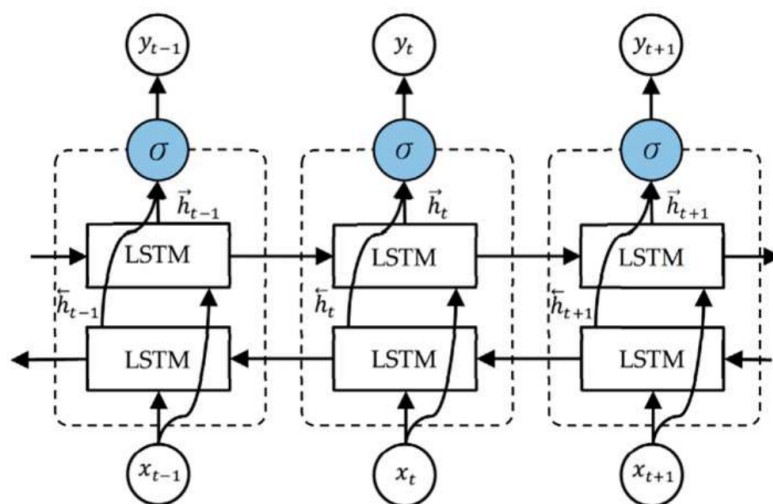
Výpočty gradientov obsahujú členy, ktoré zahŕňajú súčin viacerých Jakobových matic:

$$\frac{\partial \mathbf{h}_t}{\partial \mathbf{h}_{t-n}} = \prod_{k=t-n}^{t-1} \mathbf{J}_k, \quad (2.10)$$

kde $\mathbf{J}_k$ je Jakobova matica skrytého stavu v časovom kroku k . Gradient bude zánikať, ak sú vlastné hodnoty matic $\mathbf{J}_k$ menšie ako 1, a teda súčin týchto matic sa bude s rastúcim n blížiť k nule. Naopak, gradient bude explodovať, t.j. rásť exponenciálne, ak sú vlastné hodnoty väčšie ako 1. To môže spôsobiť, že parametre modelu sa stanú nestabilnými a počas tréningu dôjde k numerickému pretečeniu [37].

Vrstvením viacerých **RNN**, umožníme sieti učiť sa zložitejšie reprezentácie. Takýto prístup je pomenovaný ako „Deep **RNN**“, čiže **hlboké rekurentné neurónové siete**. Tie predstavujú signifikantné vylepšenie v úlohách, akým je preklad a generovanie textu, no v niektorých oblastiach môžu mať problémy s pretrénovaním [38].

Long Short-Term Memory (LSTM) eliminuje problém miznúceho gradientu, pričom na to používa tzv. „brány“ (gates), konkrétne: *input gate* i_t ; *forget gate* f_t ; *output gate* o_t , ktoré rozhodujú o tom, aké množstvo informácií z predchádzajúceho stavu ponechať a koľko zo súčasného stavu posunúť ďalej [39]. Ak spojíme dve vrstvy **LSTM** pomocou vrstvy, ktorá využíva aktivačnú funkciu *softmax*, vznikne **Bidirectional Long Short-Term Memory (BiLSTM)**. Dopredná a spätná vrstva **LSTM** v tejto architektúre bude spracovávať text postupne a umožňuje zachytenie informácií zo začiatku aj z konca [7]. Obr. 2.10 schematicky približuje princíp **BiLSTM**.



Obrázok 2.10: Schéma architektúry **BiLSTM**. Zdroj: [37].

Matematicky je možné tento tok informácií v **LSTM** využitím aktivačnej funkcie *tanh* vyjadriť nasledovným spôsobom:

$$\mathbf{i}_t = \sigma(\mathbf{W}_{xi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_i), \quad (2.11)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{xf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_f), \quad (2.12)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{xo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_o), \quad (2.13)$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{xg}\mathbf{x}_t + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_g), \quad (2.14)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \quad (2.15)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (2.16)$$

kde $\mathbf{g}_t$ je kandidát na novú hodnotu bunky (cell input); $\mathbf{c}_t$ je stav bunky (cell state); $\mathbf{h}_t$ je skrytý stav (hidden state); $\odot$ označuje prvkový (element-wise) súčin.

Podobný princíp v sebe integruje aj novšia architektúra **Gated Recurrent Units** (**GRU**), ktorá vznikla ako alternatíva ku **LSTM**, aby zmiernila jej zložitosť. Brány v **GRU** neobsahujú stav bunky $\mathbf{c}_t$, a sú len dve, a to *reset gate* r_t , ktorá kombinuje nové vstupy s doterajšou pamäťou, a *update gate* z_t , ktorá funguje veľmi podobne ako *input* a *forget gate* pri **LSTM**. Keďže **GRU** nemá *output* vrstvu, má menej tréningových parametrov, čo môže byť výhodou, ako preukázala aj istá štúdia [40], kedy pri porovnaní **GRU** a **LSTM** v robotike mala **GRU** nižšiu chybu predikcie. **Bidirectional Gated Recurrent Units** (**BiGRU**) obdobne ako **BiLSTM**, integruje viaceré vrstvy **GRU** [7]. Obr. 2.11 znázorňuje schému **GRU** oproti **LSTM**. Obdobne ako pri **LSTM** je možné tento tok informácií v **GRU** matematicky vyjadriť nasledovným spôsobom:

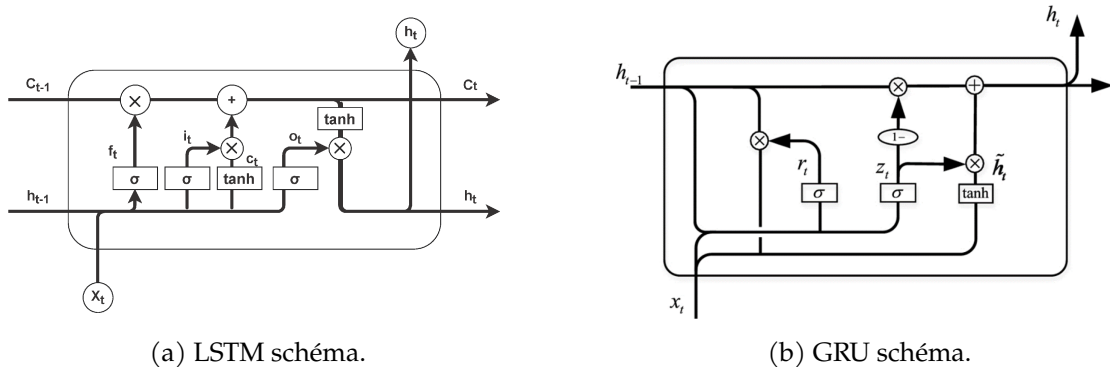
$$\mathbf{r}_t = \sigma(\mathbf{W}_{ir}\mathbf{x}_t + \mathbf{b}_{ir} + \mathbf{W}_{hr}\mathbf{h}_{t-1} + \mathbf{b}_{hr}) \quad (2.17)$$

$$\mathbf{z}_t = \sigma(\mathbf{W}_{iz}\mathbf{x}_t + \mathbf{b}_{iz} + \mathbf{W}_{hz}\mathbf{h}_{t-1} + \mathbf{b}_{hz}) \quad (2.18)$$

$$\mathbf{n}_t = \tanh(\mathbf{W}_{in}\mathbf{x}_t + \mathbf{b}_{in} + r_t \odot (\mathbf{W}_{hn}\mathbf{h}_{t-1} + \mathbf{b}_{hn})) \quad (2.19)$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{n}_t + \mathbf{z}_t \odot \mathbf{h}_{t-1} \quad (2.20)$$

kde $\mathbf{n}_t$ je kandidát na novú hodnotu bunky (cell input).



Obrázok 2.11: Porovnanie **LSTM** a **GRU**. Zdroj: [37]

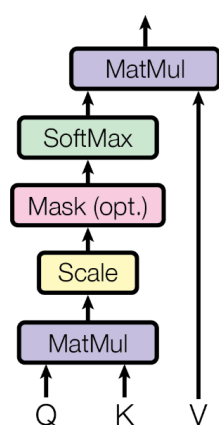
Sekvenčný charakter rekurentných neurónových sietí, ktoré priradením pozícií ku krokom v čase výpočtu generujú postupnosť skrytých stavov $\mathbf{h}_t$ ako funkciu predchádzajúceho skrytého stavu $\mathbf{h}_{t-1}$ a vstupu pre pozíciu t , zabraňuje paralelizácii, čo môže predstavovať problém pri veľkom objeme dát. Toto bolo motíváciou pre vytvorenie **Transformer** architektúry, ktorá predstavuje výrazný posun v oblasti spracovania prirodzeného jazyka. Využíva sa tu „mechanizmus pozornosti“ (self-attention mechanism), ktorý prepája rôzne pozície v rámci jednej

sekvencie s cieľom vypočítať jej reprezentáciu. „Pozornosť“ (attention function) môže byť definovaná ako mapovanie dopytu a množiny dvojíc kľúč-hodnota na výstup v podobe vektorov, pričom sa vyráta ako vážený súčet hodnôt, kde váha priradená každej hodnote sa vypočíta pomocou funkcie kompatibility dopytu s príslušným kľúčom. Každý token je najprv transformovaný troma lineárnymi projekciami, pričom výstupmi sú vektory Query (Q), Key (K) a Value (V), ktoré nejakým spôsobom charakterizujú tento token. Hodnota Query jedného tokenu sa porovnáva s hodnotou Key iného tokenu (skalárnym súčinom), čoho výsledkom po aplikácii softmax funkcie sú váhy pozornosti (attention weights). Tieto váhy vyjadrujú, nakoľko sú iné tokeny relevantné pre daný token. Výsledná hodnota attention mechanizmu teda vyjadruje význam tokenu v kontexte ostatných tokenov, keďže sa vypočíta ako vážená suma vektorov Value všetkých s ním porovnávaných tokenov. „Multi-head attention“, čiže „viacnásobná pozornosť“, potom kombinuje paralelne niekoľko vrstiev pozornosti, čo je znázornené na obr. 2.12. Konkrétne zobrazená „Scaled Dot-Product“ pozornosť je daná predpisom:

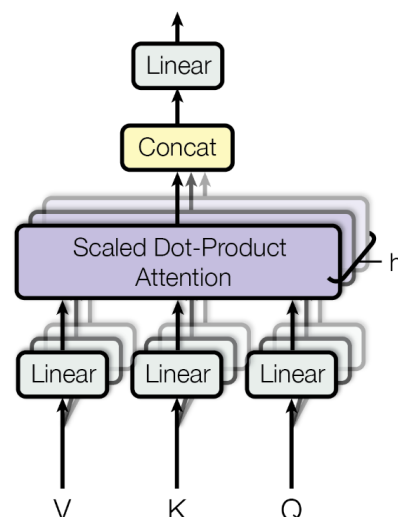
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.21)$$

pričom d_k predstavuje dimenzionalitu kľúčov. Použitie mechanizmu pozornosti má viaceré výhody, ako napr. znížená výpočtová komplexnosť jednotlivých vrstiev, možnosť paralelizácie výpočtov, či lepšia interpretovateľnosť modelov [41].

Scaled Dot-Product Attention



Multi-Head Attention

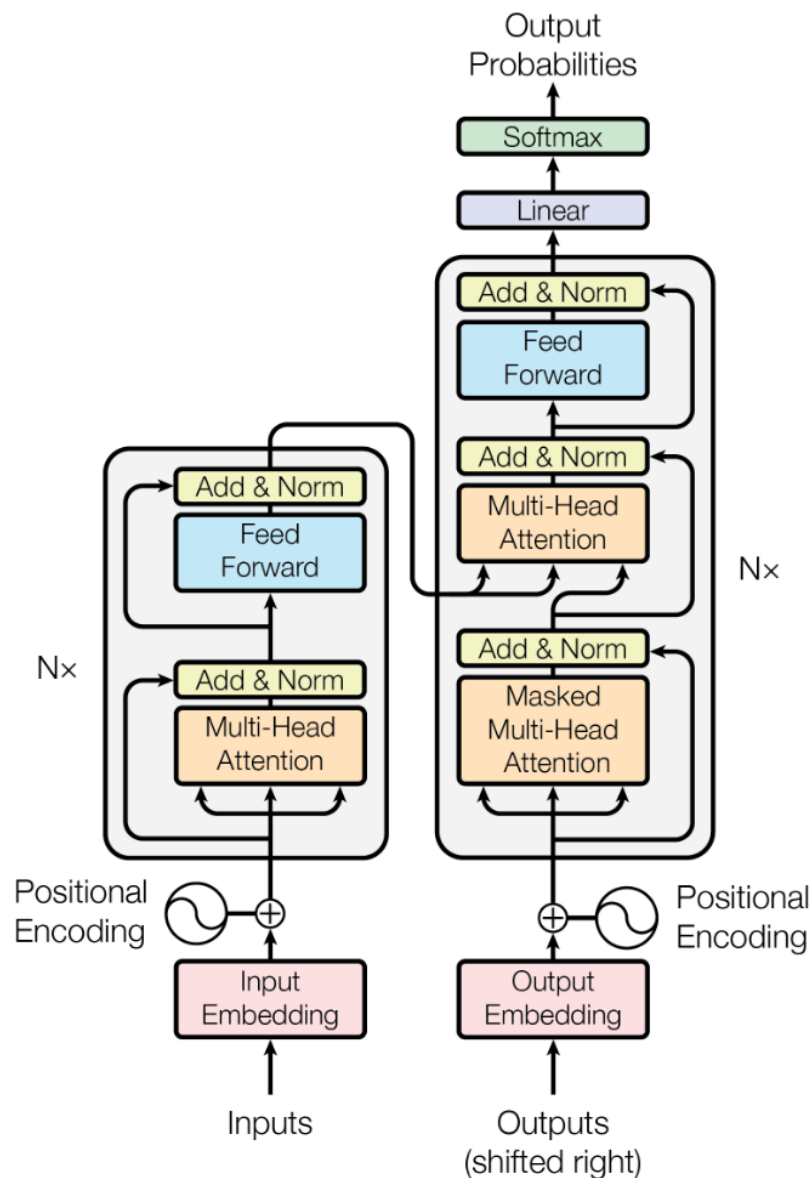


Obrázok 2.12: Schéma mechanizmu pozornosti v transformer architektúre.

Zdroj: [41].

Model transformer pozostáva z *enkóderu* a *dekóderu*. Enkóder sa skladá z N

identických vrstiev, z ktorých každá má dva funkčné komponenty, a to mechanizmus viacnásobnej pozornosti a jednoduchú polohovo plne prepojenú doprednú sieť (position-wise fully connected feed-forward network). Enkóder namapuje symbolickú reprezentáciu vstupnej sekvencie na kontinuálnu. Dekóder sa tiež skladá z N identických vrstiev, avšak každá vrstva má okrem spomínaných dvoch komponentov ešte ďalšiu zložku, ktorá vykonáva mechanizmus viacnásobnej pozornosti nad výstupom enkódera. Dekóder teda z kontinuálnej sekvencie vygeneruje výstupnú sekvenciu symbolov po jednotlivých prvkoch [41]. Na obr. 2.13 je zobrazená schéma architektúry transformer.



Obrázok 2.13: Transformer architektúra. Zdroj: [41].

2.3.3 Veľké jazykové modely

Najnovším trendom v oblasti umelej inteligencie sú veľké jazykové modely (LLM), postavené na princípe neurónových sietí, ktoré umožňujú inštruovať počítačový program, resp. model pomocou prirodzeného jazyka tak ako pri komunikácii s bežným človekom. V právnej oblasti predstavuje použitie takýchto modelov veľkú výhodu, nakoľko rôzne právne otázky sa v demokratickej spoločnosti dotýkajú každého občana, t.j. aj človeka bez pokročilejších technických znalostí. V posledných rokoch boli publikované stovky štúdií a jazykových modelov založených na transformeroch. V tejto kapitole popíšeme začiatky jazykových modelov, konkrétne modely využívané v súčasnosti a vybrané metriky na ich porovnanie a vyhodnotenie.

Začiatky

Už v 50. rokoch minulého storočia si britský matematik Alan Mathison Turing vo svojom článku kládol otázku, či dokážu stroje myslieť, a navrhol test na určenie, či počítač môže vykazovať inteligentné správanie. V teste človek (posudzovateľ) komunikuje pomocou textových správ s dvoma subjektmi, a to človekom a strojom, bez toho, aby vedel, kto je kto. Stroj prejde Turingovým testom, ak posudzovateľ nedokáže spoľahlivo určiť, ktorý zo subjektov je človek a ktorý je stroj. Turing predpokladal, že o 50 rokov počítače prejdú týmto testom. Odvtedy sa mnoho výskumníkov snaží vytvárať počítačové programy, ktoré by umožnili ľuďom komunikovať so strojmi pomocou prirodzeného jazyka. Prvým takýmto programom sa v roku 1966 stala ELIZA, ktorú vytvoril profesor na MIT Joseph Weizenbaum. Pomenoval ho po postave z divadelnej hry Pygmalion, Elize Dolittleovej, ktorú v hre profesor Henry Higgins učí rozprávať spisovnou angličtinou s akcentom britskej vyššej triedy. ELIZA predstavovala akéhosi psychoterapeuta, ktorý mal za úlohu doviesť pacienta k riešeniu jeho problémov vytváraním otázok na základe pacientových vyjadrení [42]. Takýto prístup napodobňoval metódu „client-centered counseling“, čiže poradenstva zameraného na klienta, resp. nepriamu metódu, navrhnutú Carlom Rogersom v 40. rokoch minulého storočia [43]. Ukážku konverzácie s programom ELIZA môžeme vidieť na obr. 2.14. V nadväznosti na to ďalší odborníci taktiež predstavili podobné modely ako PARRY, JABBERWACKY, A.L.I.C.E. či SmarterChild. Jedným z novodobých takýchto programov je napr. aj hlasový asistent od spoločnosti Apple, *Siri*, alebo *Alexa* od spoločnosti Amazon, ktoré sa využívajú dodnes.

Patient: Men are all alike.
Computer: In what way?
Patient: They're always bugging us about something or the other.
Computer: Can you think of a specific example?
Patient: Well, my boyfriend made me come here.
Computer: Your boyfriend made you come here?
Patient: He says I'm depressed much of the time.
Computer: I am sorry to hear you are depressed.
Patient: It is true I am unhappy.
Computer: Do you think coming here will help you not to be unhappy?
Patient: I need some help, that seems certain.
Computer: What would mean to you if you got some help?
Patient: Perhaps I could learn to get along with my mother.
Computer: Tell me more about your family.
Patient: My mother takes care of me.
Computer: Who else in your family takes care of you?
Patient: My father.
Computer: Your father?

————— Conversation continues —————

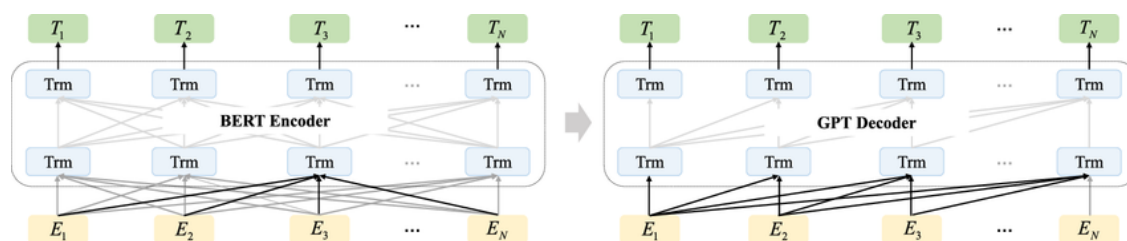
Obrázok 2.14: Ukážka konverzácie s programom ELIZA. Zdroj: [42].

Súčasnosc

Mnoho ľudí však považuje za prvý jazykový model **ChatGPT** (**Generative Pre-trained Transformers**) od spoločnosti OpenAI predstavený v novembri 2022, keďže tento model spopularizoval umelú inteligenciu medzi širokou verejnosťou a predstavuje istý prelom v tejto oblasti [42]. Tento model taktiež úspešne zvládol spomínaný Turingov test. Pri tréovaní ChatGPT bol najskôr využitý obrovský dataset rôznorodých textov z verejne dostupných zdrojov, ako knihy, články, fóra, webstránky a pod. [44]. Na doladenie bol potom aplikovaný algoritmus **Reinforcement Learning from Human Feedback (RLHF)**, na základe spätnej väzby od ľudí [45]. Pri tréovaní ChatGPT pre riadenie sa pokynmi (prompts) boli využité pokyny napísané používateľmi staršej verzie InstructGPT od OpenAI, rôznych typov ako napr. odpovedanie na otázky, dialóg, generovanie, extrakciu, sumariáciu a ďalšie [46]. ChatGPT generuje sekvencie slov na základe pravdepodobnosti, čo znamená, že niekedy si môže protirečiť, či „halucinovať“ [45]. Avšak je nesporné, že tento nástroj je veľmi nápomocný pri poskytovaní informácií a postrehov, oprave chýb v kóde, či generovaní bezpečnostných testov pre nájdenie zraniteľností v kóde [47]. Jeho verzia GPT-4 taktiež dosahuje veľmi dobrú úspešnosť nielen v angličtine, ale aj v jazykoch s nízkym počtom textových zdrojov, čo

bolo otestované benchmarkom **MMLU** [45], kedy sa testuje ako presne dokáže model odpovedať na otázky z rôznych vedných oblastí, keď dostane tri príklady v konkrétnom jazyku [46].

Ďalším veľmi populárnym jazykovým modelom je **BERT** (**Bidirectional Encoder Representations from Transformers**) od spoločnosti Google. **BERT** je predtrénovaný pomocou maskovaného jazykového modelu (**Masked Language Model** (**MLM**)), čo znamená, že úlohou je doplniť náhodne vybrané „zamaskované“ slová len pomocou kontextu. Taktiež je predtrénovaný aj na predikciu nasledujúcej vety. Textové zdroje použité na tréning tohto modelu tvorilo 16 GB nekomprimovaného textu, konkrétne korpus BookCorpus (800 miliónov slov) a anglická verzia Wikipédie, čo predstavuje približne 2,5 miliardy slov [44]. Jeho základnú štruktúru tvorí viacvrstvový obojsmerný transformer. Na segmentáciu slov využíva **BERT** metódu WordPiece, pričom vstup pre model sa skladá z troch častí: embedding slova; embedding segmentu; embedding pozície. Výhodou oproti **GPT** je extrakcia príznakov vykonávaná pomocou obojsmerného transformera [48]. **GPT** narozdiel od **BERT** používa transformerový dekódér, ktorý pracuje zľava doprava (left-to-right), keďže pri generovaní textu predpovedá najpravdepodobnejšie nasledujúce slovo v sekvencii [49]. Názorné porovnanie architektúr **BERT** a **GPT** môžeme vidieť na obr. 2.15.

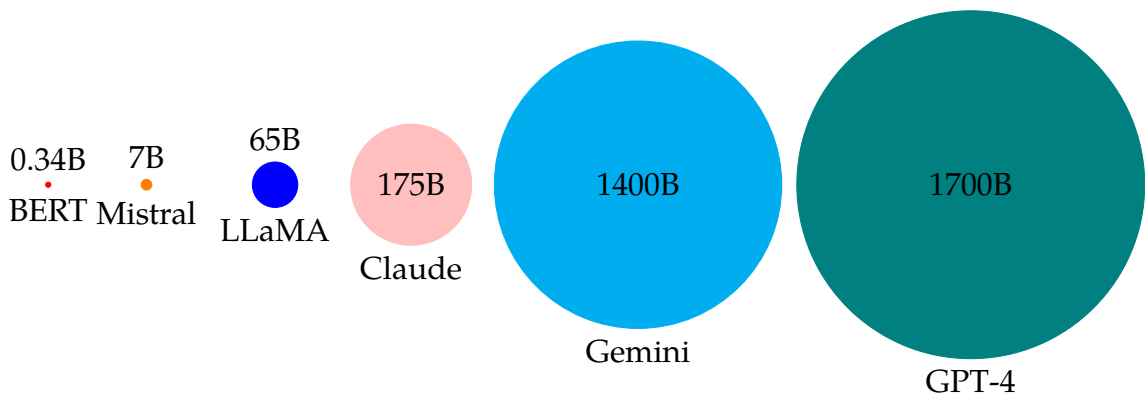


Obrázok 2.15: **BERT** oproti **GPT** architektúre. Zdroj: [48].

BERT je tzv. „base“ model primárne natrénovaný pre anglický jazyk, pričom ho tvorí približne 110 až 340 miliónov parametrov v závislosti od verzie, a je **Open-source**. To znamená, že sa tu naskytá priestor pre „transfer learning“, ktorý pomocou prenosu informácií z jednej oblasti zlepšuje vedomosti zo súvisiacej oblasti [50]. V tomto kontexte možno využiť sémantické znalosti nadobudnuté v jednom jazyku a následne model doladiť pomocou textových dát z iného jazyka [44]. Takto vznikli modely ako CamemBERT pre francúzsky jazyk [51], BERTje pre holandský jazyk [52], ALBERTo pre taliansky jazyk [53], alebo aj SlovakBERT pre slovenský jazyk [54] a mnoho ďalších. V právnej oblasti to však nemusí predstavovať až takú výhodu, keďže právne texty ako zákony a nariadenia sú viazané na konkrétny právny poriadok a nie sú prenositeľné medzi jurisdikciami, naroz-

diel od iných typov textov, ktorých obsah je ekvivalentný v rôznych jazykoch.

Avšak ani v tejto oblasti sa nedá vyhnúť istým úskaliam. Dopusiaľ zvyšovanie veľkosti jazykového modelu jasne viedlo k zlepšeniu výsledkov v úlohách jazykového spracovania, no to si vyžaduje extrémne vysoké výpočtové nároky a veľmi kvalitné dáta [55]. Pre porovnanie sú na obr. 2.16 znázornené veľkosti vybraných LLM na základe počtu ich parametrov (v miliardách).³



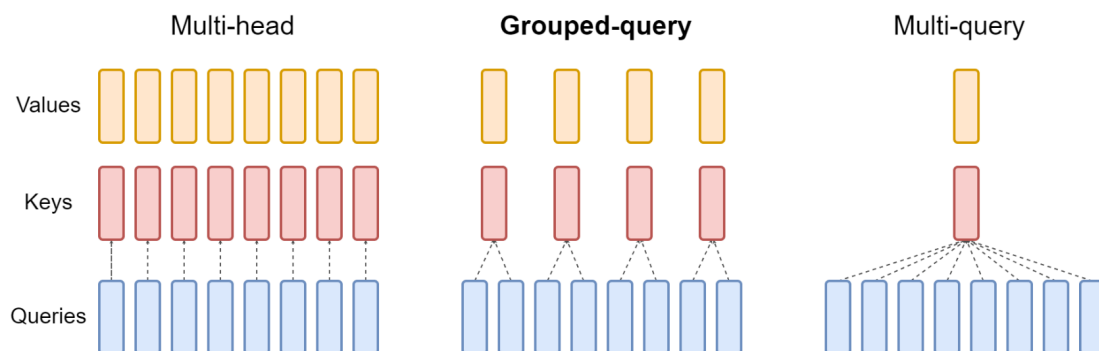
Obrázok 2.16: Porovnanie veľkostí LLM. Zdroj: vlastné spracovanie.

Aj preto sa dnes mnoho výskumníkov snaží o tzv. „distillation“, kedy sa usilujeme o zmenšenie modelu pri zachovaní jeho výkonu. Tento koncept predstavuje transformáciu implicitných znalostí modelov do explicitnej symbolickej formy, čo predstavuje aj zlepšenie interpretovateľnosti a aplikovateľnosti [56].

Výskumníci z francúzskej spoločnosti Mistral AI sa taktiež zaoberali týmto problémom, pričom sa snažili o dôkladné navrhnutie a vytvorenie vybalansovaného modelu, ktorý by dosahoval vysoký výkon pri zachovaní efektívnosti inferencie (proces, kedy sa niektoré vstupné údaje, napríklad obrázok, odošlú cez sieť raz, v doprednom priechode, aby sa vypočítal požadovaný výsledok) [57]. Takto vznikol model **Mistral** so siedmimi miliardami parametrov, ktorý prekonal úspešnosť modelov **LLaMA** s 13 a 34 miliardami parametrov [58]. Narozdiel od vyššie spomenutých modelov, nie je známe, aké dáta boli použité na tréning modelu Mistral. Jeho architektúra využíva mechanizmus pozornosti so skupinovými dotazmi (**Grouped-Query Attention (GQA)**) [59], ktorý vychádza z mechanizmu viacnásobnej pozornosti (**MHA**) (popísaný v kapitole 2.3.2), kedy každý dopyt má vlastný kľúč aj hodnotu. **GQA** však v prvom rade rozdelí dopyty do G skupín, z ktorých každá skupina má jeden vlastný kľúč a hodnotu. Tie sa vytvárajú spriemerovaním (mean-pooling) pôvodných kľúčov a hodnôt

³Zdroje údajov: [47], huggingface.co/bert, huggingface.co/mistral, ai.meta.com/llama, the-decoder.com/gpt, anthropic.com/claude

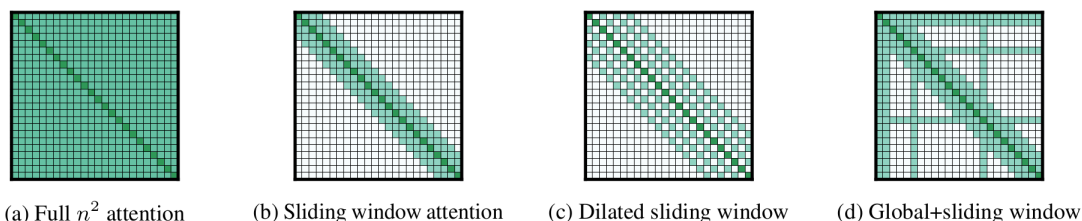
v rámci danej skupiny. Ak $G = 1$, teda nastane zoskupenie do jednej skupiny, potom sa to nazýva **Multi-Query Attention (MQA)**. Transformácia do G skupín spôsobí zrýchlenie modelu oproti **MHA**, ale zároveň ho tiež príliš nezjednoduší ako pri **MQA**, čo predstavuje dobrý kompromis pre veľké modely [59]. Na obr. 2.17 je názorne vyobrazené porovnanie týchto prístupov.



Obrázok 2.17: Porovnanie **MHA**, **GQA** a **MQA**. Zdroj: [59].

Ďalej architektúra modelu Mistral zahŕňa aj tzv. **Sliding-Window Attention (SWA)**, čiže mechanizmus pozornosti s posuvným oknom. Keďže výpočtová komplexnosť pri použití mechanizmu pozornosti narastá kvadraticky voči dĺžke vstupnej sekvencie, predstavuje to problém pri spracovaní dlhých sekvencií. Ak však integrujeme do mechanizmu posuvné okno s nemennou veľkosťou okolo každého tokenu, a naskladáme na seba viacero takýchto vrstiev, výpočtová komplexnosť bude narastať lineárne voči dĺžke sekvencie. Zároveň majú vrchné vrstvy prístup ku všetkým pozíciám vstupu a sú schopné vytvárať reprezentácie, ktoré zahŕňajú informácie z celého vstupu, čo vedie k širokému receptívnemu poľu [60]. Podobný princíp v sebe zahŕňajú aj konvolučné neurónové siete, ktoré boli popísané v kapitole 2.3.2. Spomínané posuvné okná môžu byť ešte ďalej „dilatované“. Ak okná v niektorých vrstvách dilatujeme, umožňuje to týmto vrstvám zamerať sa na širší kontext a ostatným vrstvám sústrediť sa na užší (lokálny) kontext [60]. Na obr. 2.18 je názorne zobrazené porovnanie mechanizmov pozornosti s posuvnými oknami.

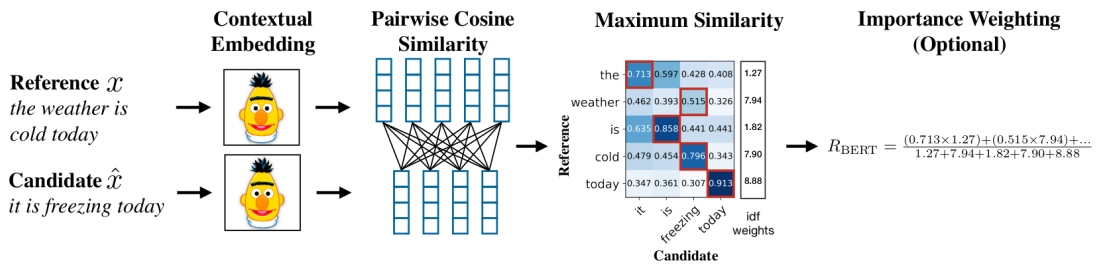
Integrácia popísaných mechanizmov v modeli Mistral významne znižuje potrebný objem pamäte počas procesu dekodovania a zvyšuje rýchlosť **inferencie**, resp. spracovania výstupov, čo umožňuje vyššiu priepustnosť, a to predstavuje zásadnú výhodu pri praktickom nasadení v reálnom čase [58]. Model Mistral bol doladený aj na slovenských dátach, čo preukázateľne viedlo k vyššej kvalite vygenerovaného textu v slovenčine [61].



Obrázok 2.18: Porovnanie pôvodného mechanizmu pozornosti, **SWA**, dilated **SWA** a kombinácie. Zdroj: [60].

Metriky pre vyhodnotenie jazykových modelov

Aby sme vedeli porovnať kvalitu výstupov jazykových modelov pri konkrétnej úlohe, môžeme na to využiť rozličné prístupy od evaluácie ľudským expertom po rôzne typy metrík. Natural Language Generation (**NLG**) metriky porovnávajú vygenerovaný text s referenčným textom, ktorý napísal človek. Takýmto typom metriky je napr. **BLEU** metrika, ktorá sa často používa pri vyhodnotení kvality prekladu pomocou jazykového modelu. Pri tejto metrike sa spočítajú zhody n -gramov kandidáta (prekladu konkrétneho jazykového modelu) s n -gramami referenčného prekladu, pričom tieto n -gramy nemusia byť na rovnakej pozícii, a platí, že kvalita kandidáta je tým vyššia, čím je vyššia táto zhoda. Takisto sa z tohto dá odvodiť tzv. „modifikovaná presnosť n -gramov“ (modified n -gram precision), kedy sa počet zhodných n -gramov vydolí počtom všetkých n -gramov [62]. Takto definovaná metrika však neberie do úvahy lexikálnu a štruktúrnú rozmanitosť, ktorá je kritická pre zachovanie pôvodného významu [63]. Metrika **METEOR** sa odlišuje od **BLEU** zameraním na návratnosť, ktorá sa preukázala ako zásadnejšia pri dosiahovaní vysokej úrovne korelácie s ľudskými úsudkami [64]. Na podobnom princípe fungujú aj metriky **ROUGE** pri vyhodnocovaní kvality sumarizácie textu [65] (pričom môže porovnávať aj zhodu dvojíc po sebe idúcich slov (ROUGE-2) alebo zhodu najdlhšej spoločnej podsekvencie (ROUGE-L)), alebo **CIDEr** pri porovnávaní kvality vygenerovaného popisu obrázka [66]. Slabú stránku **BLEU** metriky sa snaží zmierniť metrika **BERTScore**, ktorá sa vypočíta na základe kosínusovej podobnosti tokenov reprezentovaných kontextovými embeddingmi kandidáta a referenčného textu. Výpočet môže zahŕňať aj váženie pomocou **IDF**, ktoré zvýrazňuje dôležité (menej časté) slová [67]. Metrika **BERTScore** lepšie zachytáva semantický význam textu a je menej citlivá na drobné chyby, pokiaľ je kandidát štylisticky alebo lexikálne podobný referencii [68]. Názorná ukážka výpočtu návratnosti pre **BERTScore**, kedy sa porovnáva každý token referencie s tokenom kandidáta, je zobrazená na obr. 2.19.



Obrázok 2.19: Znáozornenie výpočtu návratnosti pre BERTScore. Zdroj: [67].

Dôležitou metrikou je aj Perplexity. Táto metrika neporovnáva generovaný text s referenčným textom ako vyššie popísané metriky. Perplexity vyjadruje inverznú hodnotu (geometrického) priemeru pravdepodobností priradených modelom jednotlivým slovám v texte [69]. Postup výpočtu hodnoty Perplexity je nasledovný. Nech $X = (x_1, x_2, \dots, x_T)$ je vstupná textová sekvencia tokenov a model predikuje pravdepodobnosti $P_\theta(x_t | x_1, \dots, x_{t-1})$ pre každý token x_t . Najprv sa vypočíta logaritmus pravdepodobnosti cieľového tokenu:

$$\log_prob(x_t) = \log P_\theta(x_t | x_1, \dots, x_{t-1})$$

Pre ignorovanie padding tokenov sa používa maska $mask_t \in \{0, 1\}$:

$$\text{masked_log_prob}(x_t) = \log_prob(x_t) \cdot mask_t$$

Nech L je počet nenulových tokenov (t.j. $\sum_{t=1}^T mask_t$), potom negatívny logaritmus pravdepodobnosti (negative log-likelihood) pre jednu sekvenciu je vyjadrený ako:

$$\text{NLL}(X) = -\frac{1}{L} \sum_{t=1}^T \text{masked_log_prob}(x_t)$$

Perplexita sa definuje ako hodnota exponenciálnej funkcie záporného logaritmu pravdepodobnosti:

$$\text{Perplexity}(X) = \exp(\text{NLL}(X)) = \exp\left(-\frac{1}{L} \sum_{t=1}^T \log_prob(x_t) \cdot mask_t\right)$$

Nech $W = (w_1, w_2, \dots, w_N)$ je sekvencia slov. Celková perplexita sekvencie je potom:

$$\text{Perplexity}(W) = \sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i)}}$$

kde $P(w_i)$ je pravdepodobnosť slova w_i predikovaná modelom. Vyjadruje to teda istú mieru neurčitosti modelu pri generovaní textu. Nižšie hodnoty Perplexity teda znamenajú, že model lepšie predikuje aké slovo má v kontexte nasledovať

[70]. Táto metrika je veľmi užitočná, avšak nie je aplikovateľná pri modeloch, ktoré nemajú normalizované pravdepodobnosti a takisto nie je vhodná pre porovnávanie modelov s rôznymi slovníkmi [69].

3 Implementácia

Implementačná časť práce uvedená v tejto kapitole je štruktúrovaná podľa metodológie **CRISP-DM**, pričom jednotlivé časti zodpovedajú jej postupným krokom.

3.1 Pochopenie cieľa

Pri každom výskume je potrebné si v prvom rade stanoviť cieľ, resp. hypotézu, pričom tie boli stanovené na konci kap. 2. V tejto časti popíšeme cieľ z hľadiska aplikácie v praxi. Cieľom tejto práce je vybrať jazykový model, ktorý je schopný adekvátne reprezentovať slovenské právne texty, a následne tento model dotrénovať na doménovo špecifických dátach v slovenskom jazyku. Takýto model je možné následne použiť na detekciu anomálií, keďže súdne rozhodnutia by mali vykazovať vysokú mieru predvídateľnosti a nemali by sa významne odchyľovať od ustálenej judikatúry, ako bolo spomenuté v úvode. Automatizovaná analýza rozsiahlych databáz rozhodnutí by dokázala identifikovať štatisticky nezvyčajné vzory, napr. neodôvodnené odchýlky v ukladaní trestov pri skutkovo podobných prípadoch, nekonzistentné aplikovanie právnych noriem alebo náhle zmeny v rozhodovacej praxi konkrétneho sudcu. Môže teda vystupovať ako nezávislá vrstva dohľadu, ktorá môže včas upozorniť na potenciálne pochybenia alebo systémové skreslenia.

Ďalej má táto práca za cieľ vytvoriť predikčný model na predikciu zrušenia, resp. potvrdenia rozhodnutia v odvolacom konaní. Praktické nasadenie takéhoto modelu môže priniesť aj merateľné finančné úspory. Model schopný predvídať rozhodnutia vie odhaliť, keď sa konkrétne rozhodnutie výrazne líši od zaužívaanej rozhodovacej praxe. Rozhodnutia, ktoré sú s ňou v súlade, sú zároveň menej náchylné na úspešné odvolanie, čím by bolo možné predísť nákladom spojeným s následným súdnym konaním, a takisto to môže odbremeniť súdy, ako bolo spomenuté aj v úvode.

3.2 Pochopenie dát

Táto sekcia zahŕňa krátku ukážku rozhodnutia súdu a popis textového korpusu súdnych rozhodnutí.

3.2.1 Štruktúra rozhodnutí súdu

Ako môžeme vidieť na obr. 3.1, štandardne majú rozhodnutia slovenských súdov predpísanú štruktúru, pričom obsahujú formálny úvod, v ktorom je definovaný súd, spisová značka, sudca a ECLI. Nasleduje nadpis „ROZHODNUTIE“ a po slove „r o z h o d o l :“ nasleduje rozsudok, resp. výrok. Hlavná a najdlhšia časť „O d ô v o d n e n i e :“ je členená do viacerých častí, ktoré zahŕňajú skutkový stav, sumarizovaný návrh žalobcu, príp. vyjadrenie žalovaného a priebeh súdneho konania. Záver obsahuje poučenie.

3.2.2 Textový korpus

V spolupráci s UPJŠ nám boli poskytnuté rozhodnutia vo formáte JSON, kde každé rozhodnutie je samostatný objekt s 27 atribútmi obsahujúcimi metadáta, ako je typ a názov súdu, kraj, oblasť právnej úpravy alebo meno sudcu, a atribút obsahujúci text rozhodnutia. Dáta obsahujú rozhodnutia okresných a krajských súdov, Najvyššieho súdu Slovenskej republiky a Špecializovaného trestného súdu, pričom rozdelenie počtu rozhodnutí podľa typov súdov môžeme vidieť na Obr. 3.2. Pri implementácii nás bude zaujímať najmä text rozhodnutia (dokument_fulltext). Tento textový korpus rozhodnutí súdov Slovenskej republiky obsahuje celkovo 4 104 483 dokumentov. Celkovo obsahuje 17 691 729 172 tokenov. Z hľadiska lexikálnej rozmanitosti má korpus 47 671 unikátnych tokenov (slovník), čo odráža špecifickú terminológiu právneho jazyka a relatívne nízku variabilitu v porovnaní s bežnými jazykovými korpusmi. Priemerná dĺžka dokumentu v korpuse je 4 310 tokenov, pričom medián dĺžky dokumentu je 2 553 tokenov. Rozdiel medzi priemerom a mediánom naznačuje prítomnosť výrazne dlhších dokumentov, ktoré ťahajú priemer smerom nahor, zatiaľ čo väčšina dokumentov má skôr strednú dĺžku okolo dvoch až troch tisíc tokenov. Detailný histogram môžeme vidieť na Obr. 3.3. Časové rozdelenie dokumentov v korpuse pokrýva obdobie prevažne od roku 2011 do roku 2023, ako môžeme vidieť na Obr. 3.4. Korpus obsahuje aj ojedinelé rozsudky z rokov 1965, 1987 a 1995, ktoré boli z histogramu vyradené. Histogram preto zachytáva obdobie rokov 1997 až 2023. Typy súdov v týchto rozhodnutiach zodpovedajú sústave platnej pred

Súd: Okresný súd Košice okolie
 Spisová značka: 13Csp/318/2021
 Identifikačné číslo súdneho spisu: 6120441219
 Dátum vydania rozhodnutia: 17. 12. 2021
 Meno a priezvisko sudcu, VSÚ: JUDr. Danica Hovančáková
 ECLI: ECLI:SK:OSKE3:2021:6120441219.5

Rozhodnutie

Okresný súd Košice - okolie sudkyňou JUDr. Danicou Hovančákovou v spore žalobcu: Inkasáreň s.r.o., IČO: 35 777 974, so sídlom Pribinova 25, Bratislava - Staré mesto, zast. Mgr. Zuzanou Bírovou, advokátkou so sídlom Americká 2, Bratislava - Staré Mesto proti žalovanému: V. S., D.. X.XX.XXXX, N. Z. XXX, o zaplatenie 70,70 eura s príslušenstvom takto

rozhodol:

I. Žalovaný je povinný zaplatiť žalobcovi sumu vo výške 70,70 eura s úrokom z omeškania vo výške 5 % ročne z dlžnej sumy od 31.7.2020 do zaplatenia a s nákladmi spojenými s uplatnením pohľadávky v sume 3,- eurá, všetko v lehote troch dní odo dňa právoplatnosti tohto rozsudku.

II. Žalobcovi priznáva náhradu trov konania voči žalovanému v rozsahu 100 %, o výške ktorých bude rozhodnuté samostatným uznesením po právoplatnosti tohto rozsudku.

odôvodnenie:

1. Žalobca žalobou doručitou Okresnému súdu Banská Bystrica dňa 24.11.2020 žiadal, aby súd v upomínamom konaní vydal platobný rozkaz, ktorým zaviaže žalovaného zaplatiť žalobcovi sumu 70,70 eura s úrokom z omeškania 5 % ročne zo sumy 70,70 eura od 31.7.2020 do zaplatenia, náklady spojené s uplatnením pohľadávky 3,- eurá a náhradu trov konania. Podanú žalobu odôvodnil tým, že žalovaný sa dňa 30.7.2020 prepravoval mestskou hromadnou dopravou v Bratislave a pri kontrole nepreukázal platný cestovný lístok. Podľa Prepravného poriadku integrovaného dopravného systému v Bratislavskom kraji vyhláseného a zverejneného právnym predchodcom žalobcu (ďalej len „Prepravný poriadok“) a v zmysle zákona č. 514/2009 Z.z. o doprave na dráhach a č. 56/2012 Z.z. o cestnej doprave žalovaný ako cestujúci uzavrel dohodu s dopravcom momentom nastúpenia do dopravného prostriedku dopravcu a na výzvu je ako cestujúci povinný predložiť platný cestovný lístok. Cestujúci, ktorý sa na vyzvanie podľa bodu A.14.3 prepravného poriadku nemôže preukázať platným cestovným lístkom, je povinný zaplatiť dopravcovi okrem cestovného aj sankčnú úhradu. Žalovaný neuhradil cestovné 0,70 eura a sankčnú úhradu 70,- eur. Pohľadávka bola postúpená na žalobcu zmluvou o postúpení pohľadávok 3.4.2019, čo bolo žalovanému oznámené.

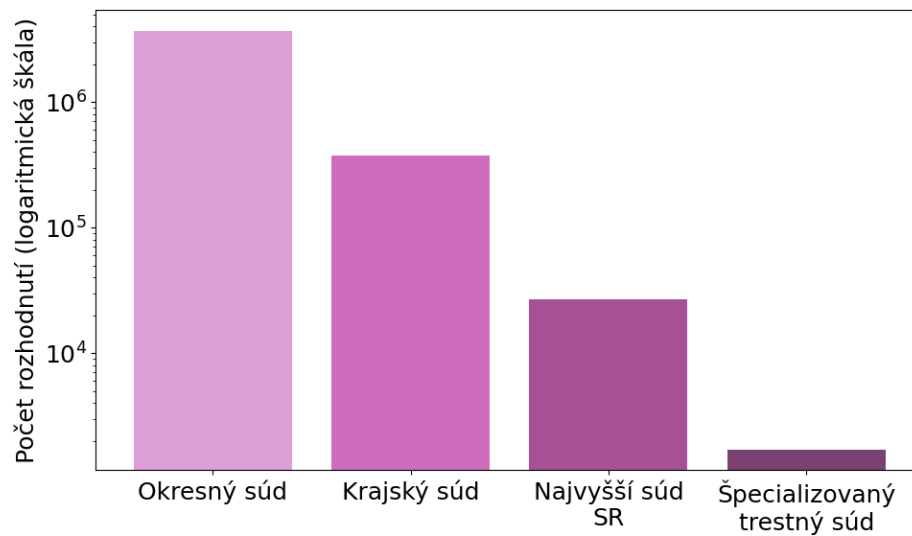
2. Platobný rozkaz Okresného súdu Banská Bystrica zo dňa 14.12.2020 sp.zn. 24Up/2153/2020 nebolo možné žalovanému doručiť do vlastných rúk a žalobca navrhol pokračovať v konaní, upomínací súd preto dňa 4.5.2021 postúpil vec na ďalšie konanie tunajšiemu súdu ako vecne a miestne príslušnému.

3. Súd na návrh žalobcu rozhodol vo veci platobným rozkazom zo dňa 26.5.2021 č.k. 13Csp/318/2021-32, ktorým žalobe v celom rozsahu vyhovel. Platobný rozkaz spolu so žalobou nebolo možné doručiť žalovanému do vlastných rúk, preto ho súd uznesením zo dňa 7.10.2021, č.k. 13Csp/318/2021-41 zrušil v plnom rozsahu.

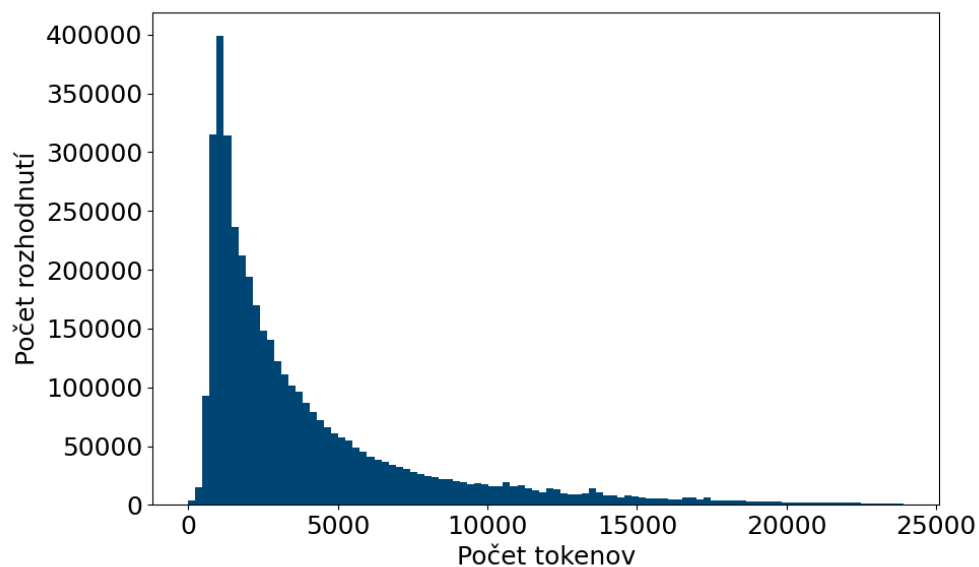
Obrázok 3.1: Ukážka rozhodnutia súdu vo forme pdf. Zdroj:

<https://www.justice.gov.sk/sudy-a-rozhodnutia/sudy/rozhodnutia/868e98b9-5f04-439f-be05-d66c994c5db9:fe08907a-242a-44ae-b512-20a37bd95529/>

reformou súdnej mapy z roku 2023, ako bolo popísané v kap. 2.1.



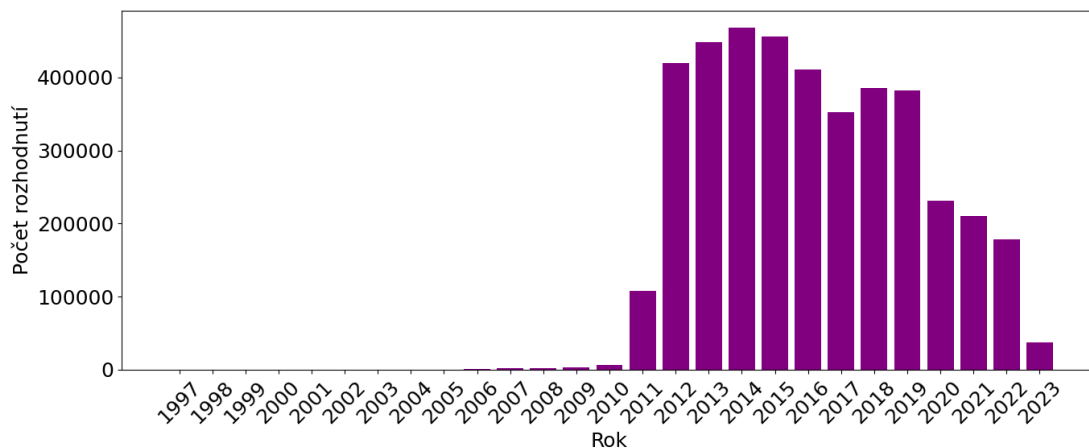
Obrázok 3.2: Počet rozhodnutí podľa konkrétnych typov súdov.



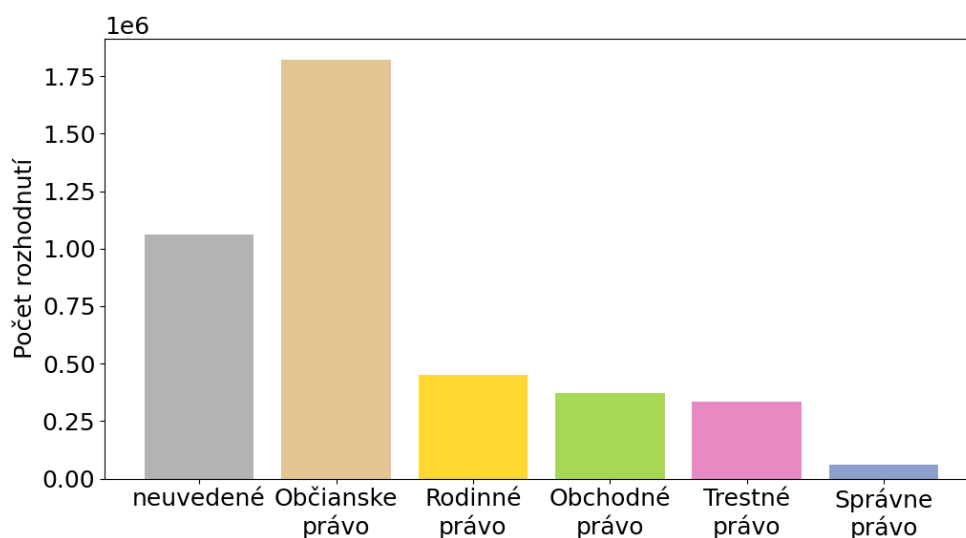
Obrázok 3.3: Počet rozhodnutí s konkrétnou dĺžkou tokenov po 99.percentil.

Čo sa týka oblasti právnej úpravy, korpus obsahuje rozhodnutia týkajúce sa obchodného, trestného, občianskeho, správneho a rodinného práva, pričom najviac je rozhodnutí týkajúcich sa občianskeho práva. Rozdelenie hodnôt môžeme vidieť na Obr. 3.5.

Metadáta obsahujú aj údaj o povahe rozhodnutia. Počty rozhodnutí podľa povahy rozhodnutia, ktoré boli preskúvané v odvolacom konaní (8.7% rozhodnutí), môžeme vidieť na Obr. 3.6 a 3.7, pričom najväčšia časť rozhodnutí bola po-



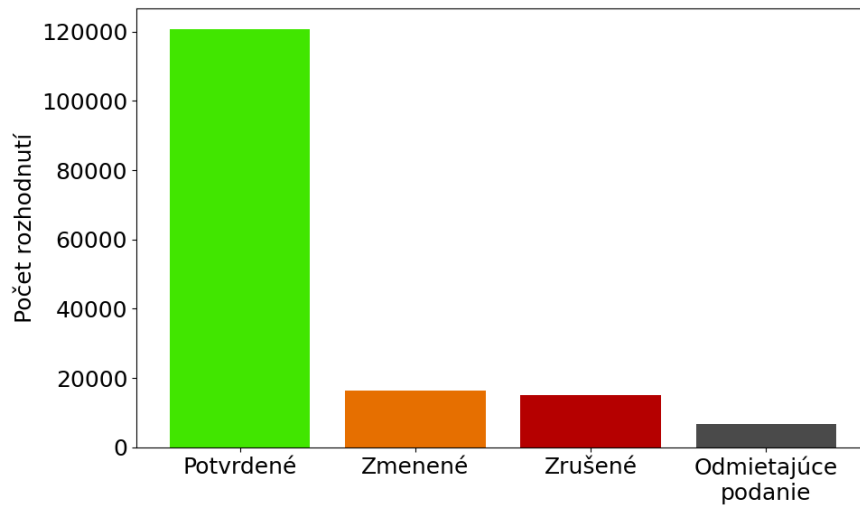
Obrázok 3.4: Počet rozhodnutí za konkrétne roky.



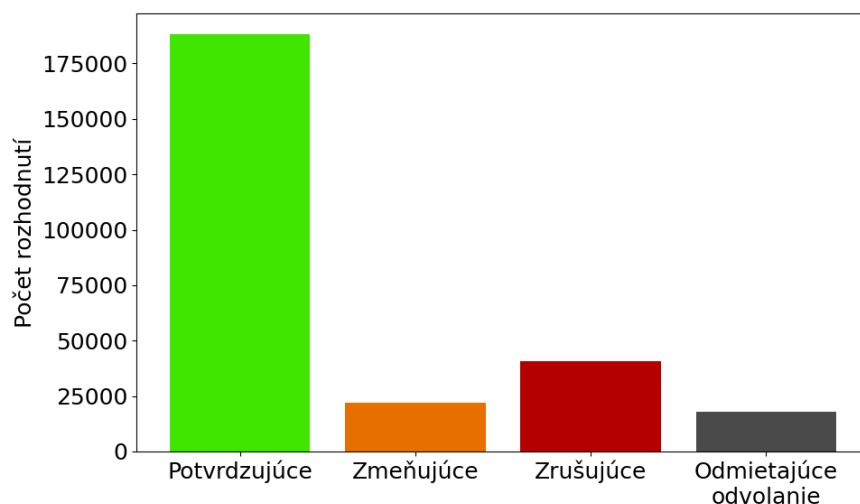
Obrázok 3.5: Počet rozhodnutí podľa oblasti právnej úpravy.

tvrdená, resp. potvrdzujúca (okolo 75%). Okrem zobrazených označení (Zrušené, Zrušujúce, Zmenené, Zmeňujúce, Potvrdené, Potvrdzujúce, Odmietajúce podanie, Odmietajúce odvolanie) boli ešte rozhodnutia označené aj ako „Iná povaha rozhodnutia“, „Zastavujúce odvolacie konanie“, „Pripúšťajúce späťvzatie návrhu, zrušujúce rozsudok a zastavujúce konanie“ (všetky tri spolu tvoria necelých 0,7%) a „Prvostupňové nenapadnuté opravnými prostriedkami“, pričom tých bolo najviac (3 180 705, čo predstavuje približne 77% všetkých rozhodnutí). Po hlbšom skúmaní bolo zistené, že takto označené rozhodnutia môžu predstavovať rozhodnutia, proti ktorým nebolo podané odvolanie, ale aj rozhodnutia prvostupňového súdu, ktoré boli vykonané po vrátení súdu vyššej inštancie na základe odvolania. Približne polpercento rozhodnutí bolo neoznačených. Taktiež

na základe hlbšej analýzy sa ukázalo, že v tomto prípade mohlo ísť aj o zrušené rozhodnutie.

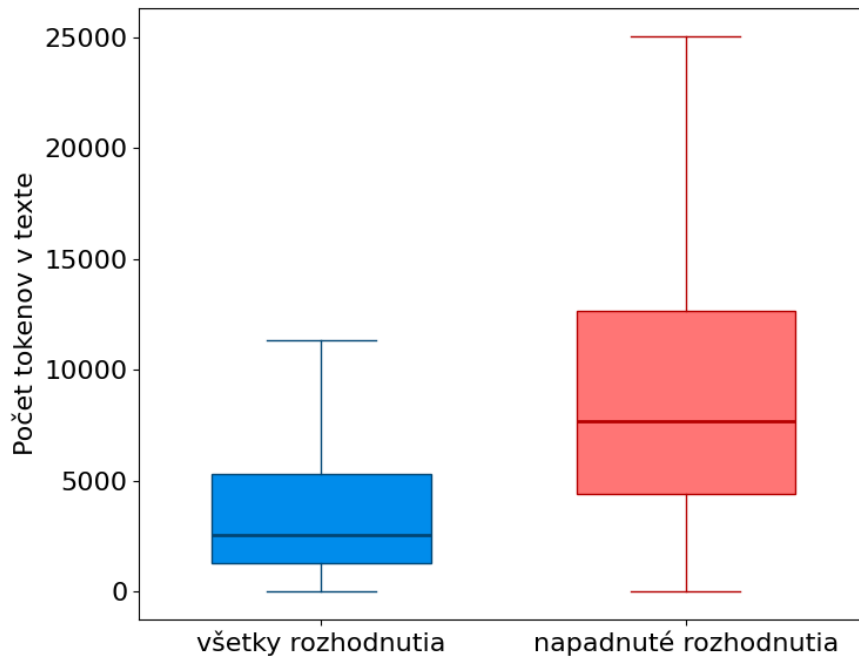


Obrázok 3.6: Počet prvostupňových rozhodnutí podľa povahy rozhodnutia.



Obrázok 3.7: Počet rozhodnutí súdov vyšších stupňov podľa povahy rozhodnutia.

Ak sa pozrieme bližšie na napadnuté rozhodnutia (zrušené, zmenené a potvrdené), z vizualizácie dĺžky spojenia odôvodnenia a výroku na Obr. 3.8 je zrejmé, že tieto rozhodnutia sú značne rozsiahlejšie než ostatné, ich distribúcia hodnôt sa líši a typická dĺžka je približne trikrát väčšia. Možno sa domnievať, že ide o komplexnejšie prípady, ktoré si prirodzene vyžadujú rozsiahlejšie odôvodnenie. Väčší rozsah týchto rozhodnutí teda pravdepodobne odráža vyššiu mieru právnej zložitosti daných sporov, ako aj procesnú povinnosť súdu podrobne sa vyrovnáť s argumentáciou účastníkov konania.



Obrázok 3.8: Porovnanie dĺžky všetkých a napadnutých rozhodnutí.

3.3 Predspracovanie

V tejto časti je popísaná predpríprava textového korpusu obsahujúca normalizáciu, čistenie a rozdelenie textu rozhodnutia na jednotlivé časti.

3.3.1 Normalizácia a čistenie dát

Vstupné textové dáta rozhodnutí slovenských súdov obsahujú viaceré formálne a typografické artefakty vzniknuté pri digitalizácii a exporte dokumentov (napr. nadpisové hlavičky, nadbytočné medzery, špeciálne znaky či rozdelené slová). Pred ich využitím v úlohách spracovania prirodzeného jazyka bolo preto nevyhnutné vykonať systematické čistenie a normalizáciu textu.

Na tento účel bol implementovaný skript v jazyku Python, ktorý spracúva vstupný súbor vo formáte **JSON** postupne (streamovo), aby bola zachovaná nízka pamäťová náročnosť aj pri veľkom objeme dát. Na iteratívne načítanie objektov z **JSON** poľa je použitá knižnica `ijson`, ktorá umožňuje spracovanie jednotlivých rozhodnutí bez potreby načítať celý súbor do operačnej pamäte, nakoľko naša dátová množina má veľkosť vyše 45 GB.

Hlavným cieľom predspracovania je extrakcia a vyčistenie plného textu rozhodnutia, ktorý je v dátach uložený v atribúte `dokument_fulltext`. Tento text je po spracovaní uložený do nového atribútu `text`, pričom pôvodný neupravený

obsah je odstránený, aby sa predišlo redundancii.

V rámci čistenia textu sa realizujú nasledujúce kroky:

- **Odstránenie hlavičky dokumentu** – texty rozhodnutí často obsahujú formálne úvodné časti (napr. identifikátory ECLI, názvy súdu a pod.), ktoré nemajú význam pre jazykovú analýzu a sú zároveň znova uvedené v metadátoch. Text je preto orezaný od prvého výskytu identifikátora ECLI, čím sa zachováva iba samotné odôvodnenie a výroková časť rozhodnutia.
- **Normalizácia zalomení riadkov** – Unicode znak U+2028 (line separator) sa v dátach používa nejednotne. Izolované výskyty sú nahradené medzerou, zatiaľ čo viacnásobné výskyty sú zredukované na jeden znak, čím sa zachováva štruktúra odsekov bez fragmentácie viet.
- **Redukcia nadbytočných medzier** – všetky viacnásobné medzery (okrem oddeľovačov odsekov) sú nahradené jednou medzerou, čím sa zvyšuje konzistentnosť textu.
- **Zlučovanie umelo rozdelených slov** – v rozhodnutiach sa často vyskytujú slová, ktoré sú rozdelené na jednotlivé písmená oddelené medzerami (napr. „o d ô v o d n e n i e“). Pomocou regulárnych výrazov sú tieto sekvencie identifikované a rekonštruované do pôvodnej podoby slov. Na identifikáciu takýchto sekvencií bol použitý regulárny výraz s podporou Unicode tried znakov: `(?<=.\s)(\p{L}[\s.:]){3,}`. Tento regulárny výraz vyhľadáva sekvencie, ktoré pozostávajú z opakovaní vzoru písmena nasledovaného medzerou alebo znakom „.“ alebo „:“. Tento vzor sa musí vyskytnúť aspoň trikrát za sebou, aby sa predišlo nesprávnemu zlučovaniu krátkych fragmentov. Pred samotnou sekvenciou je použitý pozitívny lookbehind, ktorý zabezpečuje, že zlučovanie sa vykoná len vtedy, ak sekvenciu predchádza znak a medzera, čím sa znižuje riziko zlúčenia aj s predchádzajúcim slovom. Po nájdení takejto sekvencie je použitá funkcia, ktorá odstráni medzery medzi písmenami a tým obnoví pôvodné slovo.

Celý proces čistenia je zapuzdrený v samostatnej funkcii, čo zvyšuje čitateľnosť kódu a umožňuje jeho jednoduché opätovné použitie alebo rozšírenie v ďalších experimentoch. Výstupom skriptu je nový **JSON** súbor obsahujúci pôvodné metadáta rozhodnutí spolu s normalizovaným textom pripraveným na ďalšie kroky spracovania, ako je tokenizácia, extrakcia znakov alebo jemné doladovanie jazykového modelu. Takto realizované predspracovanie významne prispieva

k zníženiu šumu v dátach a zvyšuje kvalitu vstupu pre následné modelovanie, resp. tréovanie jazykového modelu.

3.3.2 Segmentácia textu rozhodnutí na logické časti

Po prieskumovaní a normalizácii textov rozhodnutí je ďalším krokom ich rozdelenie na logické časti. Rozhodnutia súdov SR majú typickú štruktúru, ktorá zvyčajne zahŕňa úvodnú časť, samotné rozhodnutie (výrok), odôvodnenie a záverečnú informáciu o právnom poučení. Pre účely analýzy a následného modelovania je výhodné tieto časti separovať, pretože jednotlivé segmenty sa líšia štylisticky aj obsahovo a môžu mať odlišnú informatívnu hodnotu pre model. Takisto je nutné oddeliť odôvodnenie od výroku rozhodnutia pre vyhodnotenie metrick jazykového modelu založených na prekrytí, ako napr. BLEU a ROUGE.

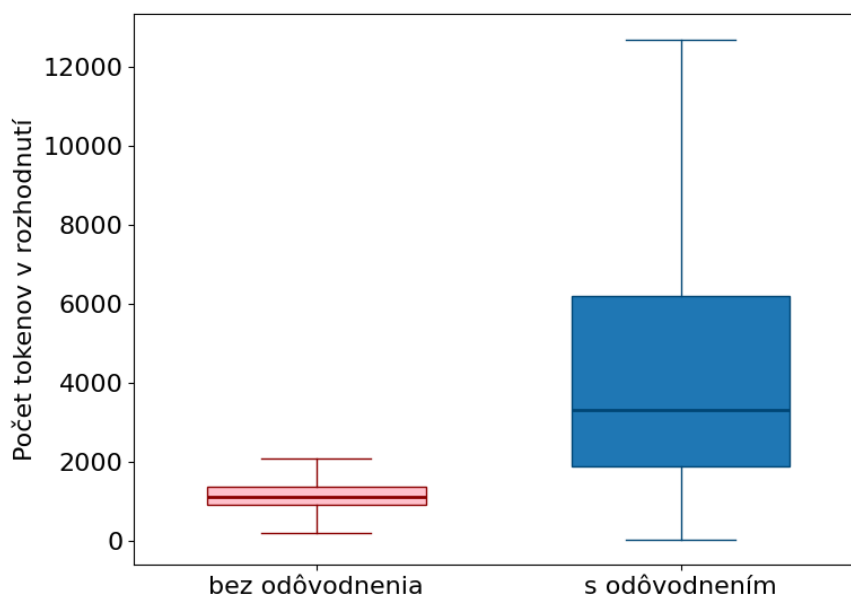
V tomto kroku bol implementovaný skript, ktorý pre každý dokument vykonáva automatické rozdelenie textu na tri základné časti:

- decision – časť textu pred odôvodnením, ktorá obsahuje identifikáciu rozhodnutia, právne východiská a samotný výrok. Neskôr pre tréovanie jazykového modelu a konečné vyhodnotenie bola táto časť ešte ďalej rozdelená na dve časti, a to:
 - introduction (úvod) - vynechaný pri tréovaní modelu;
 - statement (výrok) - použitý na tréovanie modelu.
- reasoning (odôvodnenie) – hlavná argumentačná časť, v ktorej sú uvedené právne dôvody a analýza prípadu.
- judicial_notice (poučenie) – záverečná časť, ktorá informuje o možnosti odvolania alebo ďalších právnych krokoch.

Rozdelenie je založené na identifikácii v texte často používaných sekvenčných markerov. Skript vyhľadáva výraz „odôvodnenie“ (bez ohľadu na veľkosť písmen) a považuje ho za začiatok argumentačnej časti. Ako koniec odôvodnenia je použitý marker „poučenie“, ktorý je v rozhodnutiach typický pre záverečné informácie. Takýto prístup je v praxi bežný pri segmentácii právnych dokumentov, kde sú textové časti často oddelené slovnými signálmi (tzv. section headings). Keďže pri prvotnej normalizácii textu sme previedli umelo rozdelené slová „o d ô v o d n e n i e“ a „p o u č e n i e“ na bežný tvar, môžeme na identifikáciu častí odôvodnenia a poučenia použiť regulárny výraz: `odôvodnenie.*(?=[Pp]oučenie)`, ktorý funguje nasledovne:

- odôvodnenie – vyhladá slovný marker začiatku odôvodnenia.
- .* – zachytí ľubovoľný počet znakov (vrátane medzier a nových riadkov), čím umožní zachytiť celý obsah odôvodnenia až po koniec sekcie.
- (?=[Pp]oučenie) – tzv. positive lookahead, ktorý zabezpečí, že text skončí tesne pred slovom „Poučenie“ alebo „poučenie“, bez toho aby sa tento marker zaradil do zachyteného textu. Zároveň je použitá voľba `re.IGNORECASE`, takže je vyhľadávanie nezávislé od veľkosti písmen.

Ak je v texte identifikovaná sekvencia „odôvodnenie ... poučenie“, skript vykoná: extrakciu úvodnej časti (všetko pred slovom „odôvodnenie“); extrakciu odôvodnenia (od slova „odôvodnenie“ až po „poučenie“); extrakciu poučenia (všetko za „poučenie“). V prípade, že sa v dokumente marker „odôvodnenie“ nenájde (napr. v dôsledku netypického formátovania alebo chýb v texte), skript uloží celý obsah do atribútu `decision` a ponechá prázdne hodnoty pre odôvodnenie a poučenie. Identifikátory takýchto dokumentov sú zaznamenané do samostatného súboru, z čoho bolo zistené, že približne pri pätine rozhodnutí týmto skriptom nebolo odôvodnenie identifikované. Ako však naznačuje porovnanie na Obr. 3.9, tieto rozhodnutia zrejme odôvodnenie skutočne ani neobsahujú, keďže rozdelenia sa výrazne líšia. Medián je podstatne nižší v porovnaní so skupinou rozhodnutí s odôvodnením. Zároveň je interkvartilové rozpätie užšie, čo naznačuje menšiu variabilitu dĺžky dokumentov.



Obrázok 3.9: Porovnanie počtu tokenov v rozhodnutiach s odôvodnením a bez odôvodnenia.

Výstupom skriptu je nový **JSON** súbor, kde každý záznam obsahuje už segmentované časti textu. Takto pripravené dáta sú vhodné na samostatné modelovanie, napr. na analýzu štylistiky odôvodnení, klasifikáciu podľa typu rozhodnutia alebo extrakciu právnych argumentov.

3.4 Modelovanie

Pre analýzu rozhodnutí súdov Slovenskej republiky budú využité veľké jazykové modely, ktorých princípy fungovania sú popísané v kapitole 2. Pri spracovaní právnych textov je však potrebné zohľadniť špecifiká právneho jazyka, ktoré sa odlišujú od bežnej hovorovej alebo publicistickej slovenčiny. Rozhodnutia súdov sú charakteristické presnou terminológiou, štruktúrovanými formuláciami, opakovanými právnymi frázami a typickými formálnymi konštrukciami, ktoré v bežných korpusoch môžu byť podreprezentované.

Z tohto dôvodu sa ukazuje ako vhodné a žiaduce dotrénovanie existujúceho predtrénovaného jazykového modelu na špecifickom korpuse rozhodnutí. Tento prístup umožňuje modelu adaptovať sa na právny štýl, zlepšiť schopnosť rozpoznávať a generovať právne relevantné konštrukcie a zvýšiť kvalitu reprezentácií textu pre následné úlohy, ako sú klasifikácia, extrakcia informácií alebo predikcia výsledku rozhodnutia. Dotrénovanie (fine-tuning) zároveň pomáha zmierniť problém doménovej adaptácie, keď model predtrénovaný na všeobecnom texte nemusí dostatočne zachytávať špecifické významy právnych termínov a kontextu.

3.4.1 Porovnanie jazykových modelov pri generovaní rozsudku

Aby sme vedeli vybrať model vhodný na dotrénovanie, ohodnotili sme vybrané modely ktoré už boli trénované aj na slovenských textoch, metrikami Perplexity, BLEU, ROUGE-1, ROUGE-2 a ROUGE-L, pričom ich význam bol detailnejšie popísaný v kap. 2.3.3. Porovnávali sme teda 10 modelov dostupných na platforme huggingface.co, a to konkrétne:

- Qwen/Qwen3-8B;
- google/gemma-7b-it⁴;

⁴Model google/gemma-7b-it nakoniec nebol zahrnutý do porovnania, keďže pri jeho evaluácii dochádzalo po spracovaní niekoľkých rozsudkov k opakovaným chybám typu *OutOfMemoryError*.

- deepseek-ai/deepseek-llm-7b-chat;
- meta-llama/Llama-3.1-8B-Instruct;
- mistralai/Mistral-7B-Instruct-v0.3;
- slovak-nlp/mistral-sk-7b;
- openGPT-X/Teuken-7B-instruct-v0.6;
- utter-project/EuroLLM-9B-Instruct;
- BSC-LT/salamandra-7b-instruct;
- LLaMAX/LLaMAX3-8B-Alpaca.

Porovnanie bolo vykonané na vzorke 3 303 rozhodnutí⁵, čo predstavuje približne 0,1% z rozhodnutí obsahujúcich odôvodnenie. Pre porovnanie schopností veľkých jazykových modelov generovať rozhodnutia na základe odôvodnenia bol implementovaný experimentálny skript v jazyku Python, ktorý postupne načítava viacero predtrénovaných modelov, generuje text a následne vyhodnocuje kvalitu generovaného výstupu pomocou viacerých metrík. Hlavným cieľom je zistiť, ktorý model najlepšie dokáže transformovať odôvodnenie na výsledné rozhodnutie.

Skript v prvom kroku nastaví a načíta model. Pre každý model zo zoznamu sa načíta predtrénovaný model typu Causal LM spolu s príslušným tokenizérom. Modely sú ukladané do lokálneho cache adresára, aby sa znížila potreba opakovaného sťahovania. Pre každý model je tiež určený torch_dtype (float16 alebo bfloat16, čo predstavuje uloženie čísla v 16bitoch), teda dátový typ, v ktorom sú uložené váhy a výpočty modelu, čo optimalizuje využitie GPU pamäte a zrýchľuje výpočty. Potom nasleduje príprava vstupných dát, pričom ako vstup slúži predspracovaný JSON súbor, v ktorom je text dokumentu rozdelený na časti reasoning (odôvodnenie) a decision (výrok). Pre každý záznam sa vytvorí prompt, ktorý modelu zadáva úlohu: na základe odôvodnenia vygenerovať rozhodnutie. Bol k tomu využitý jednoduchý system prompt „You are a slovak judge. Generate judicial decision from reasoning.“ Pre niektoré modely sa prompt konštruuje v chat formáte (s rolami system a user), pre iné modely je použitý špecifický prompt formát podľa ich požiadaviek (napr. Qwen, Teuken, Salamandra, LLaMAX). Modely sú obmedzené na generovanie maximálne 500 nových tokenov.

⁵Konečný počet je u modelov deepseek-llm-7b-chat, Qwen3-8B a Teuken-7B-instruct-v0.6 menší o jeden rozsudok, keďže pri jednom neobvykle dlhom rozhodnutí (vyše 50000 tokenov), dochádzalo k *OutOfMemoryError*.

Texty sú následne spracované v dávkach (batchoch) veľkosti `batch_size = 1`. Po vygenerovaní rozhodnutia modelom sa vyhodnotí kvalita výstupu pomocou spomínaných metrík. Pre každý batch a každý model sa priebežne zapisujú hodnoty BLEU, ROUGE a Perplexity do samostatných súborov. Experiment bol realizovaný paralelne na štyroch grafických kartách NVIDIA RTX A6000 s kapacitou 48 GB, pričom na každej GPU bežal samostatný skript pre iný model. V dôsledku obmedzenej kapacity GPU (pamäť a výpočtový výkon) bolo nutné zvoliť batch size 1, aby bol beh skriptov stabilný a aby sa minimalizovalo riziko vyčerpania pamäte. Po dokončení experimentu sa výsledné hodnoty agregujú a ukladajú do **JSON** súborov pre následnú analýzu a vizualizáciu.

Výsledkom je porovnateľný súbor metrík pre viacero modelov, ktorý umožňuje vyhodnotiť, ktorý model je najvhodnejší pre úlohu generovania rozhodnutí na základe odôvodnenia. Tento experiment poskytuje kvantitatívne základy pre výber modelu do ďalšej časti práce. Výsledky sme vizualizovali pomocou box-plotov pre jednotlivé modely, ako možno vidieť na Obr. 3.10 až 3.14 a detailných tabuľkách Tab. 3.1 až Tab. 3.5 obsahujúcich deskriptívne štatistiky: priemer; medián; minimum; maximum; štandardná odchýlka. Je nutné poznamenať, že pred výpočtom štatistických hodnôt boli z dát odstránené hodnoty nekonečna pre metriku Perplexity, aby bola zabezpečená robustnosť výpočtov, a to konkrétne jedna hodnota pri modeli Mistral-7B-Instruct-v0.3 a 59 hodnôt pri modeli LLaMAX3-8B-Alpaca.

Tabuľka 3.1: Výsledné štatistiky hodnoty Perplexity pre jednotlivé modely.

Model	Mean	Median	Min	Max	Std
deepseek-llm-7b-chat	6,992	6,531	2,141	43,750	2,259
Mistral-7B-Instruct-v0.3	5,051	4,742	1,756	17,422	1,437
mistral-sk-7b	2,208	2,158	1,377	5,379	0,282
Qwen3-8B	4,973	4,652	1,745	21,469	1,505
EuroLLM-9B-Instruct	38,984	4,250	2,219	$33 * 10^3$	632,26
salamandra-7b-instruct	3,289	3,045	1,372	11,578	1,178
Teuken-7B-instruct-v0.6	$33 * 10^3$	2,141	1,164	$45 * 10^6$	10^6
Llama-3.1-8B-Instruct	4,864	4,570	1,758	16,422	1,330
LLaMAX3-8B-Alpaca	5,095	4,211	1,982	68,188	4,631

Hodnoty Perplexity pre väčšinu skúmaných modelov sa mimo odľahlých hodnôt pohybujú v rozmedzí 1 až 10. Modely EuroLLM-9B-Instruct a Teuken-7B-instruct-v0.6 ukazujú vyššie hodnoty Perplexity a omnoho vyššiu štandardnú

Tabuľka 3.2: Výsledné štatistiky hodnoty BLEU pre jednotlivé modely.

Model	Mean	Median	Min	Max	Std
deepseek-ai/deepseek-llm-7b-chat	0,012	0,000	0,000	0,472	0,033
mistralai/Mistral-7B-Instruct-v0.3	0,056	0,037	0,000	0,538	0,068
slovak-nlp/mistral-sk-7b	0,017	0,000	0,000	0,408	0,041
Qwen/Qwen3-8B	0,051	0,035	0,000	0,481	0,061
utter-project/EuroLLM-9B-Instruct	0,026	0,000	0,000	0,585	0,049
BSC-LT/salamandra-7b-instruct	0,017	0,000	0,000	0,374	0,033
openGPT-X/Teuken-7B-instruct-v0.6	0,017	0,000	0,000	0,515	0,043
meta-llama/Llama-3.1-8B-Instruct	0,052	0,037	0,000	0,590	0,062
LLaMAX/LLaMAX3-8B-Alpaca	0,050	0,029	0,000	0,545	0,069

Tabuľka 3.3: Výsledné štatistiky hodnoty ROUGE-1 pre jednotlivé modely.

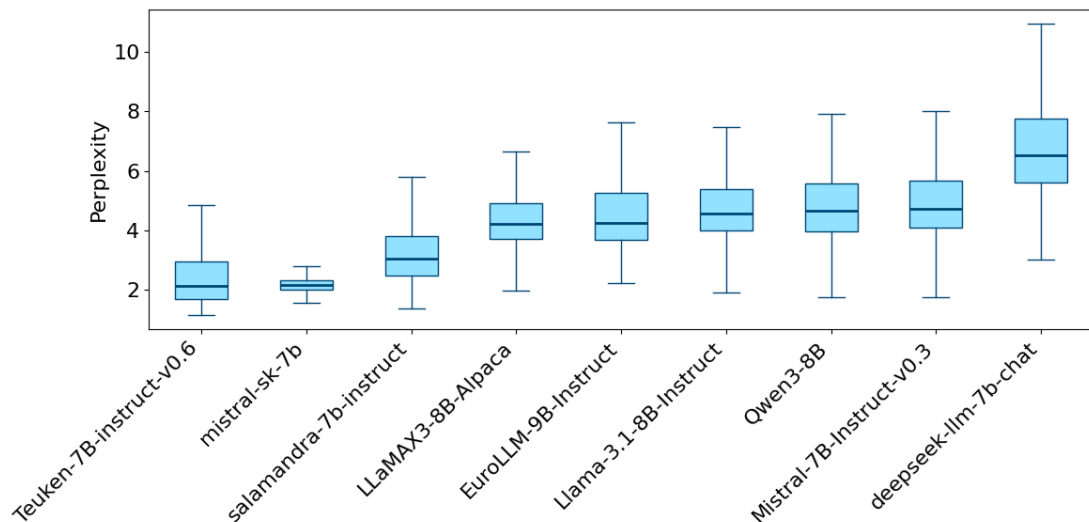
Model	Mean	Median	Min	Max	Std
deepseek-ai/deepseek-llm-7b-chat	0,156	0,146	0,000	0,604	0,114
mistralai/Mistral-7B-Instruct-v0.3	0,288	0,283	0,005	0,735	0,093
slovak-nlp/mistral-sk-7b	0,186	0,182	0,000	0,657	0,091
Qwen/Qwen3-8B	0,294	0,291	0,022	0,756	0,087
utter-project/EuroLLM-9B-Instruct	0,199	0,202	0,000	0,766	0,123
BSC-LT/salamandra-7b-instruct	0,172	0,167	0,000	0,560	0,088
openGPT-X/Teuken-7B-instruct-v0.6	0,099	0,045	0,000	0,706	0,121
meta-llama/Llama-3.1-8B-Instruct	0,287	0,280	0,000	0,692	0,084
LLaMAX/LLaMAX3-8B-Alpaca	0,251	0,257	0,000	0,779	0,117

Tabuľka 3.4: Výsledné štatistiky hodnoty ROUGE-2 pre jednotlivé modely.

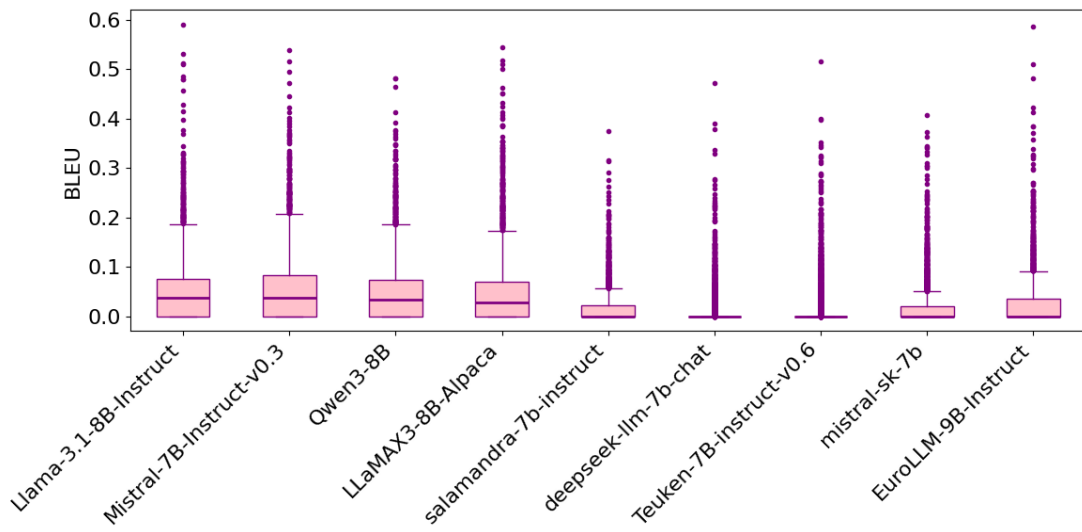
Model	Mean	Median	Min	Max	Std
deepseek-ai/deepseek-llm-7b-chat	0,040	0,018	0,000	0,446	0,053
mistralai/Mistral-7B-Instruct-v0.3	0,105	0,089	0,000	0,634	0,077
slovak-nlp/mistral-sk-7b	0,044	0,027	0,000	0,557	0,053
Qwen/Qwen3-8B	0,107	0,094	0,000	0,665	0,072
utter-project/EuroLLM-9B-Instruct	0,062	0,038	0,000	0,730	0,072
BSC-LT/salamandra-7b-instruct	0,046	0,035	0,000	0,423	0,045
openGPT-X/Teuken-7B-instruct-v0.6	0,033	0,000	0,000	0,572	0,061
meta-llama/Llama-3.1-8B-Instruct	0,107	0,094	0,000	0,603	0,071
LLaMAX/LLaMAX3-8B-Alpaca	0,099	0,085	0,000	0,757	0,082

Tabuľka 3.5: Výsledné štatistiky hodnoty ROUGE-L pre jednotlivé modely.

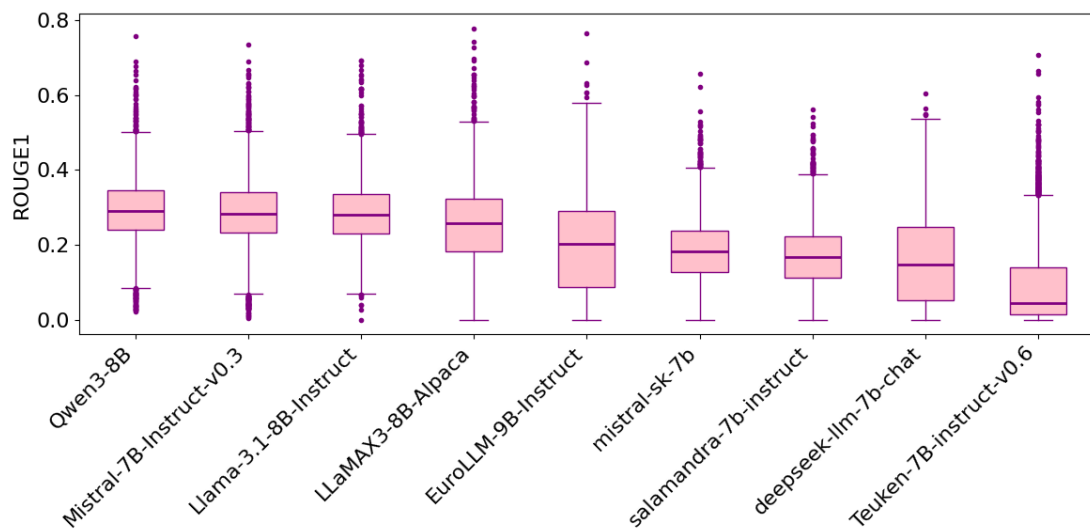
Model	Mean	Median	Min	Max	Std
deepseek-ai/deepseek-llm-7b-chat	0,095	0,091	0,000	0,464	0,065
mistralai/Mistral-7B-Instruct-v0.3	0,167	0,155	0,005	0,687	0,068
slovak-nlp/mistral-sk-7b	0,117	0,116	0,000	0,560	0,053
Qwen/Qwen3-8B	0,172	0,161	0,017	0,703	0,063
utter-project/EuroLLM-9B-Instruct	0,118	0,113	0,000	0,727	0,076
BSC-LT/salamandra-7b-instruct	0,094	0,089	0,000	0,474	0,050
openGPT-X/Teuken-7B-instruct-v0.6	0,064	0,037	0,000	0,599	0,074
meta-llama/Llama-3.1-8B-Instruct	0,172	0,161	0,000	0,630	0,061
LLaMAX/LLaMAX3-8B-Alpaca	0,156	0,149	0,000	0,761	0,079



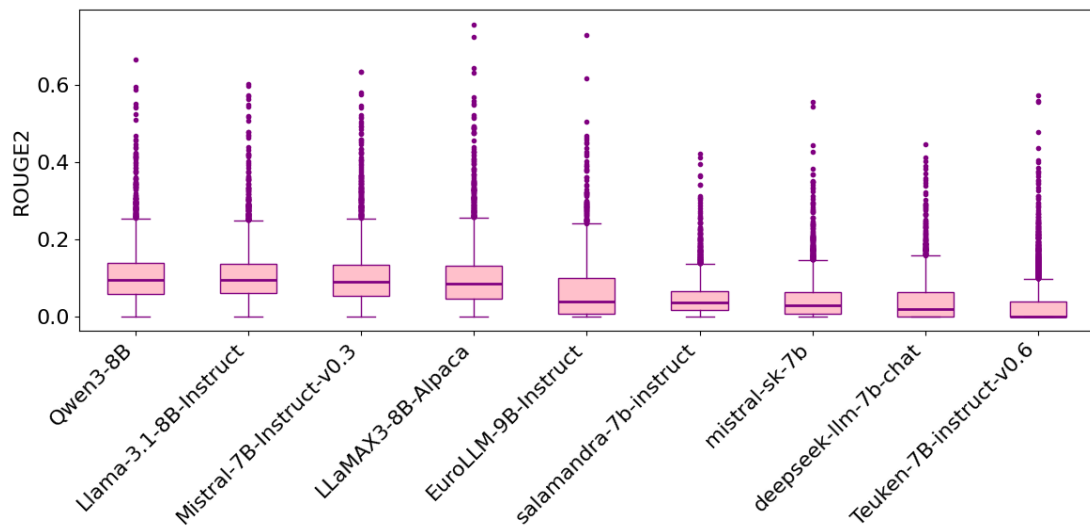
Obrázok 3.10: Porovnanie Perplexity score.



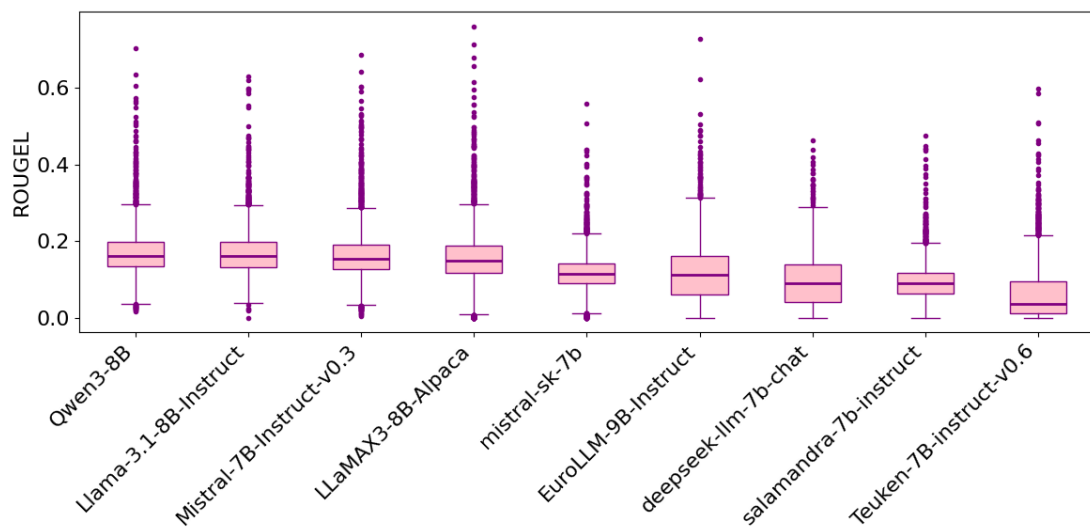
Obrázok 3.11: Porovnanie BLEU score.



Obrázok 3.12: Porovnanie ROUGE-1 score.



Obrázok 3.13: Porovnanie ROUGE-2 score.

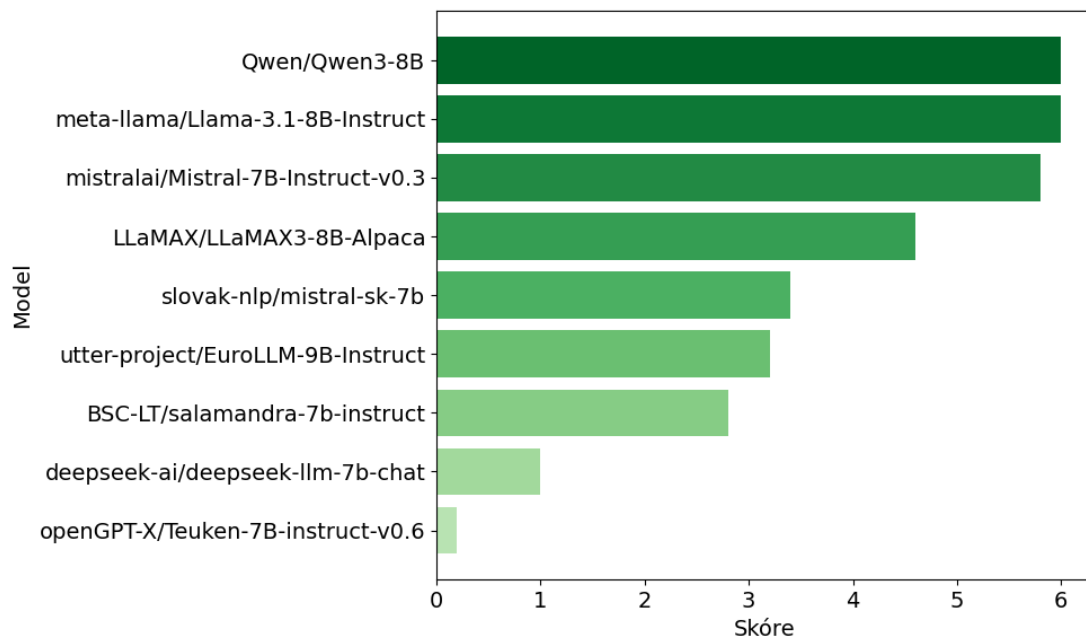


Obrázok 3.14: Porovnanie ROUGE-L score.

odchýlku, čo signalizuje menšiu stabilitu modelov a väčšiu mieru neurčitosti pri predikovaní nasledovných tokenov. Hodnoty metriky BLEU sa pre všetky modely veľmi nelíšia, priemerné hodnoty sa nachádzajú všetky pod hranicou 0,06 a maximá okolo hodnoty 0,6 dosahujú len odľahlé hodnoty. Najvyššie priemerné hodnoty ROUGE-1 dosiahli modely Qwen3-8B (0,294), Mistral-7B-Instruct-v0.3 (0,288) a Llama-3.1-8B-Instruct (0,287), čo naznačuje ich stabilne vyššiu zhodu generovaných textov s referenčnými odpoveďami. Naopak, model openGPT-X/Teuken-7B-instruct-v0.6 dosiahol najnižšie priemerné aj mediánové skóre, pričom zároveň vykazuje vysokú variabilitu výsledkov. Rozdiely medzi priemerom a mediánom sú pri väčšine modelov relatívne malé, čo naznačuje vyvážené rozdelenie hodnôt. Najvyššie priemerné hodnoty ROUGE-2 takisto dosiahli modely Qwen3-8B a Llama-3.1-8B-Instruct (zhodne 0,107), tesne nasledované modelom Mistral-7B-Instruct-v0.3 (0,105). Väčšina modelov vykazuje veľmi nízke mediánové hodnoty a nulové minimá, čo poukazuje na značnú náročnosť metriky ROUGE-2 a na časté prípady, v ktorých sa dvojice slov v generovaných výstupoch nezhodujú s referenčnými textami. Obdobne aj pri metrike ROUGE-L najvyššie priemerné hodnoty dosiahli modely Qwen3-8B a Llama-3.1-8B-Instruct (zhodne 0,172), nasledované modelom Mistral-7B-Instruct-v0.3 (0,167). Tieto modely tak najlepšie zachovávajú globálnu sekvenčnú štruktúru textu v porovnaní s referenčnými odpoveďami.

Na základe uvedených hodnôt pre každú metriku bol pre každý pár modelov vykonaný jednostranný z-test rozdielu stredných hodnôt s nerovnakými rozptylmi. Hladina štatistickej významnosti bola stanovená na 0,05. Ak bola na základe p-hodnoty zistená štatisticky signifikantne vyššia hodnota metriky (resp. štatisticky signifikantne nižšia hodnota v prípade metriky Perplexity), bol danému modelu priradený jeden bod. Na konci boli sčítané body pre model vydelené počtom metrík (5). Stĺpcový graf na Obr. 3.15 zobrazuje takéto agregované skóre modelov, ktoré možno interpretovať ako priemerný počet modelov, nad ktorými daný model dosiahol lepší výsledok v rámci jednej metriky. Hodnota skóre 6 teda znamená, že model v priemere prekonal šesť ostatných modelov v rámci jednej metriky.

Ako môžeme vidieť, najvyššie sa umiestnili modely Llama-3.1-8B-Instruct a Qwen3-8B (zhodne 6,0) a tesne za nimi je na treťom mieste model Mistral-7B-Instruct-v0.3 (5,8). Na štvrtom mieste sa umiestnil model LLaMAX3-8B-Alpaca s hodnotením 4,6, avšak ako bolo spomenuté, takmer v 60 rozhodnutiach bola hodnota metriky Perplexity nekonečno, ktoré tu neboli zohľadnené. Nekonečné hodnoty metriky Perplexity predstavujú riziko, keďže poukazujú na obmedze-



Obrázok 3.15: Konečné porovnanie modelov.

nú schopnosť modelu pracovať s doménovo špecifickou slovnou zásobou, ktorá je typická pre právne texty a rozhodnutia súdov. Na piatom, šiestom a siedmom mieste s hodnotením okolo 3 sú modely mistral-sk-7b (3,4), EuroLLM-9B-Instruct (3,2) a salamandra-7b-instruct (2,8), ktoré predstavujú menej vhodnú alternatívu. Najnižšie sa umiestnili model deepseek-llm-7b-chat s hodnotou 1,0, a model Teuken-7B-instruct-v0.6. s hodnotou len 0,2 kvôli extrémne vysokým hodnotám variability metriky Perplexity.

3.4.2 Dotrénovanie jazykového modelu

Na základe vyhodnotenia boli vybrané dva modely, ktoré boli dotrénované na vybranej časti textového korpusu, a to Qwen/Qwen3-8B s najvyšším hodnotením, a slovak-nlp/mistral-sk-7b ako model primárne natrénovaný na slovenských textoch s najnižšími hodnotami Perplexity. Tieto modely patria medzi dekóderové transformerové architektúry podobné modelom typu GPT. Ich technická konfigurácia je zobrazená v Tab. 3.6.

Doladenie prebiehalo na už spomínanej grafickej karte NVIDIA RTX A6000 pomocou metódy LoRA (Low-Rank Adaptation) s parametrom rank nastaveným na hodnotu 16 a obmedzením učenia na váhy Q, V a Gate, čo predstavuje učenie približne 0,12% pôvodných parametrov modelu. Z pôvodného textového korpusu boli vybrané na prvotné doladenie rozhodnutia, kde kombinácia odôvodnenia a výroku nepresahuje po tokenizácii 2 048 tokenov, čiže približne

Tabuľka 3.6: Porovnanie technickej konfigurácie modelov.

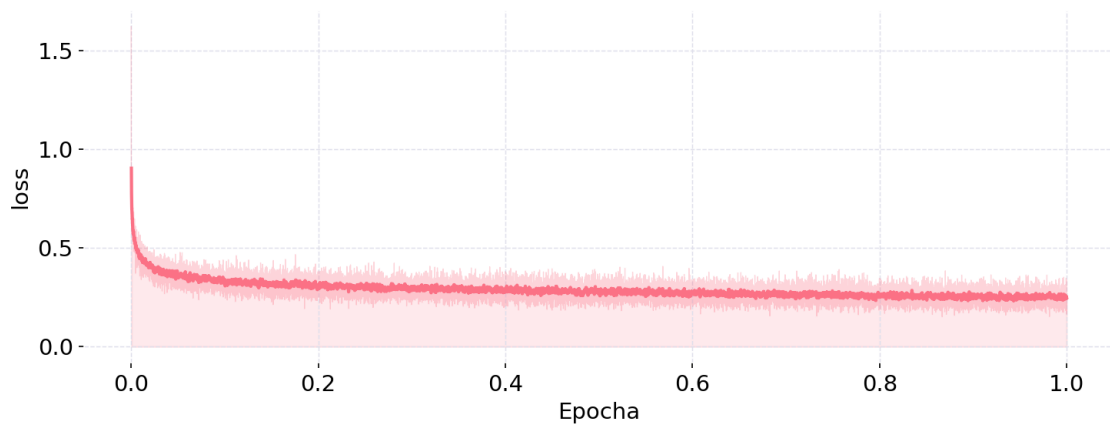
Parameter	Qwen3-8B	mistral-sk-7B
Počet parametrov	8,2B	7,3B
Počet vrstiev	36	32
Počet attention hláv	32 Q + 8 KV	32 Q + 8 KV
Window size	4 096	4 096
FFN dimenzia (hidden_dim)	11 008	14 336
Positional encoding	RoPE	RoPE
Kontextové okno	32 768 tokenov	8 192 tokenov
Normalizácia	RMSNorm	RMSNorm

40% textov, resp. 1,37 miliónov textov. Takto bolo možné nastaviť maximálnu dĺžku vstupnej sekvencie na hodnotu 2 048 bez nutnosti orezávania častí rozhodnutia, čo výrazne zrýchlilo učenie. Následne boli transformované tieto dáta do formátu konverzačnej šablóny (chat template). Každý záznam je štruktúrovaný ako trojica správ: systémová inštrukcia definujúca rolu modelu ako slovenského sudcu, používateľský vstup obsahujúci odôvodnenie a asistentova odpoveď obsahujúca výrok rozhodnutia ukončený špeciálnym tokenom konca sekvencie (eos_token). Následne je aplikovaná tokenizačná chat šablóna konkrétneho modelu. Takisto bola použitá akcelerácia pomocou „FlashAttention“ pre efektívnejšiu pozornosťnú mechaniku. Veľkosť batch-u bola z dôvodu obmedzenej kapacity pamäte grafickej karty nastavená na hodnotu 1, pričom parameter GRADIENT_ACCUMULATION_STEPS bol nastavený na hodnotu 4, čím sa efektívna veľkosť batch-u zvýšila na hodnotu 4.

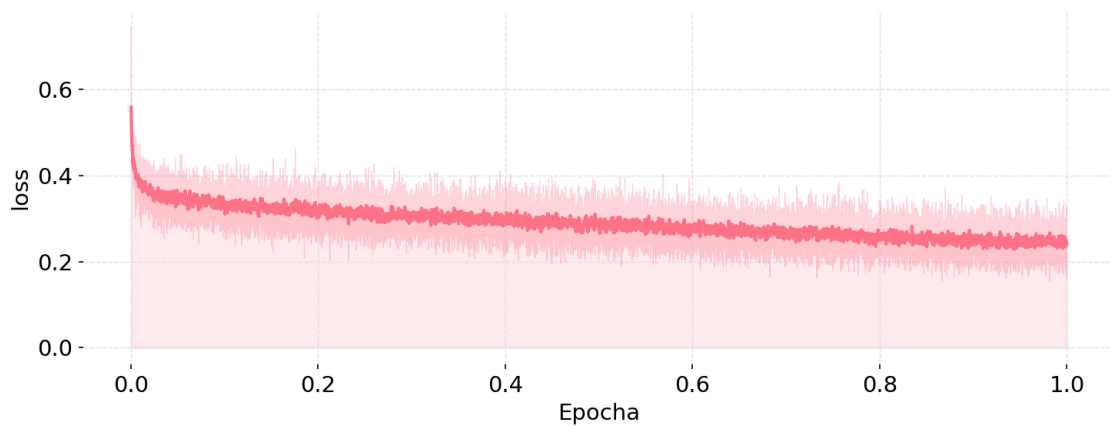
Dotrénovanie modelov trvalo približne 9 dní pre slovak-nlp/mistral-sk-7b pri rýchlosti 1,745 (train_samples_per_second) a 10 dní pre Qwen/Qwen3-8B pri rýchlosti 1,443 (train_samples_per_second). Na Obr. 3.16 až 3.18 môžeme vidieť vizualizovaný priebeh tréovania.

3.4.3 Klasifikátor na predikciu úspešnosti odvolania

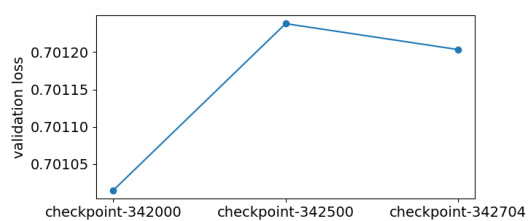
Ďalej sa zameriame na predikciu úspešnosti odvolania. Aby sme natrénovali model, ktorý určí, či je pravdepodobnejšie, že v odvolacom konaní bude rozsudok zrušený, resp. aspoň zmenený, alebo potvrdený súdom vyššej inštancie, musíme najprv pripraviť trénovacie dáta, ktoré obsahujú rozhodnutia, ktoré boli napadnuté. Z nášho textového korpusu je to pri niektorých rozhodnutiach jednoznačné na základe metadát, konkrétne povahy rozhodnutia, a to pri 11 378 roz-



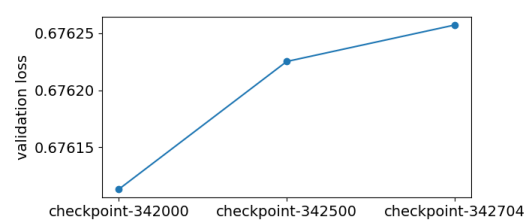
Obrázok 3.16: Pribeh trénovacej straty počas trénovania modelu Qwen.



Obrázok 3.17: Pribeh trénovacej straty počas trénovania modelu Mistral.



(a) Qwen

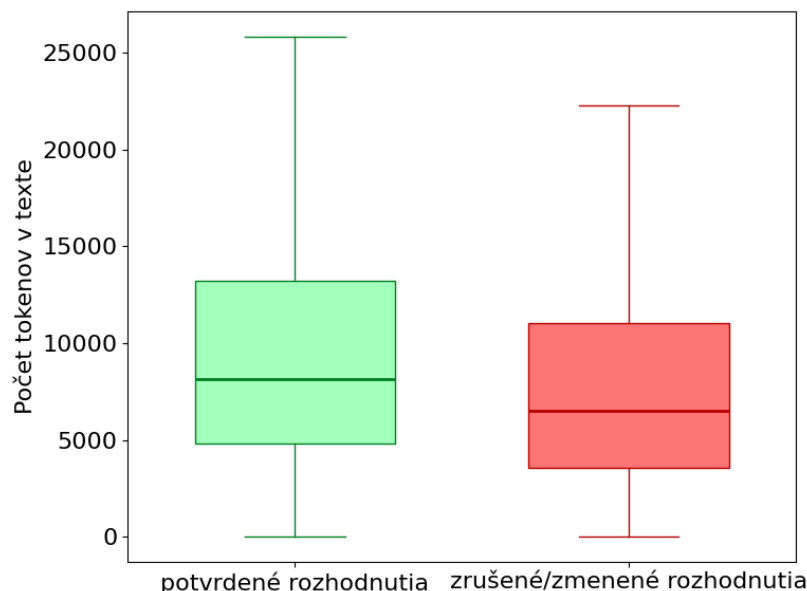


(b) Mistral-sk

Obrázok 3.18: Pribeh validačnej straty počas konca trénovania modelov.

hodnutiach s označením „Zrušené“, 13 600 rozhodnutiach s označením „Zmenené“ a 101 187 rozhodnutiach s označením „Potvrdené“. Ďalších 13 742 zrušených, resp. zmenených rozhodnutí sme identifikovali spárovaním s rozhodnutím súdu vyššej inštancie prostredníctvom rovnakého identifikačného čísla spisu a označenia povahy rozhodnutia ako „Zrušujúce“, resp. „Zmeňujúce“. Na základe porovnania dĺžok textov rozhodnutí napadnutých odvolaním, ktorú môžeme vidieť na Obr. 3.19 sú v našej analýze výrazne kratšie rozhodnutia, ktoré boli zrušené alebo zmenené ako tie, ktoré boli potvrdené. Tento rozdiel korešponduje s povahou odvolacieho konania. Súd vyššej inštancie môže vrátiť vec na nové konanie výhradne z dôvodu nedostatočného odôvodnenia prvostupňového rozsudku, keďže účastníci konania majú právo na dostatočné odôvodnenie súdneho rozhodnutia. Takéto vracanie vecí je z ekonomického a procesného hľadiska zbytočné a nákladné, pretože výsledok rozhodnutia zostáva fakticky nezmenený. Prvostupňový súd tak musí vec prejednať a vydať nový rozsudok s lepším odôvodnením, ktoré však vedie k identickému výroku.

Z tejto množiny bola vybratá vzorka rozhodnutí s dĺžkou najviac 3 000 tokenov, čo predstavuje 20 132 rozhodnutí, kde pomer medzi zrušenými, resp. zmenenými a potvrdenými rozhodnutiami je približne 1:2, teda zrušené (zmenené) rozhodnutia predstavujú minoritnú triedu. Testovacia množina obsahuje 2 014 rozhodnutí so zachovaním pomeru týchto tried.



Obrázok 3.19: Porovnanie dĺžky potvrdených rozhodnutí a zrušených/zmenených rozhodnutí.

Zero-shot a Few-shot prompting

Najprv priamo pristúpime k otestovaniu našich dvoch dotrénovaných modelov pomocou pokročilých techník prompt engineeringu. Konkrétne sme aplikovali zero-shot a few-shot prompting prístupy na rôzne varianty promptov s cieľom porovnať ich vplyv na kvalitu generovaných výstupov. Zero-shot prompting nám umožnil testovať schopnosť modelov vykonávať úlohy bez predchádzajúcich príkladov v prompte, čo nám poskytne lepší obraz o vedomostiach modelov získaných počas tréningu. Prvotne sme odladili prompt pre model Qwen na niekoľkých príkladoch, pričom prvý prompt vyzeral takto:

```
""""
You are a slovak legal expert.
Decide whether an appeal will be successful
and the court decision will be overturned.
Answer YES or NO.\n
""""
```

Vyladenie spočívalo najmä vo vylepšení formátu, akým model vygeneroval odpoveď, a v pochopení, čo znamená YES a NO, keďže YES pre zrušené rozhodnutie môže pôsobiť tak trochu kontradiktórne. Ďalej boli teda odskúšané takéto prompty:

```
""""
You are a slovak legal expert.
Decide whether an appeal will be successful
and the court decision will be overturned.
Reply with only YES or NO – no explanation.\n
""""

""""
You are a slovak legal expert.
Decide whether an appeal for certain court decision
will be successful and the court decision
will be overturned or changed.
Reply with only YES or NO – no explanation.\n
""""
```

Nakoniec bola časť „Reply with only YES or NO — no explanation.“ pridaná až za samotný príklad. Maximálna dĺžka generovanej sekvencie bola nastavená na 20 tokenov, keďže aj napriek inštrukcii niekedy model vygeneroval odpoveď

vo forme ako napr.: „Napadnuté uznesenie okresného súdu potvrdzuje“ alebo „Mení uznesenie súdu 1.stupňa“, čo je v zásade tiež relevantné pre našu úlohu, takže to bolo zohľadnené aj vo vyhodnotení. Pri vyhodnotení bolo uznané za správne namiesto kľúčového slova YES aj „zrušuje“, „mení“ alebo „vracia“, a namiesto kľúčového slova NO aj „potvrdzuje“ či „zamieta“.

Pre model mistral-sk bol finálny prompt v slovenčine a vyzeral takto:

"""

Si slovenský právny expert špecializujúci sa na odvolacie konanie. Rozhodni, či bude odvolanie úspešné a či bude rozhodnutie súdu zrušené.

ÁNO znamená, že odvolanie bude úspešné

a pôvodný rozsudok bude zrušený alebo zmenený.

NIE znamená, že pôvodný rozsudok bude potvrdený.

"""

Pričom za konkrétny príklad bola pridaná inštrukcia:

"""

\nOdpoved:

\n\nOdpovedz iba ÁNO alebo NIE.

Žiadne vysvetlenie, žiadne odôvodnenie.

"""

Avšak výsledný formát v drvivej väčšine vygenerovaných odpovedí vôbec nebol taký, ako sme ho definovali v prompte a odpovede boli úplne irelevantné, a to aj po vyskúšaní oficiálneho formátu few-shot promptu v dokumentácii modelu mistral⁶, ktorý bol takýto:

"""

You are a slovak legal expert. Your task is to detect whether an appeal for certain court decision will be successful and the court decision will be overturned or changed.\n"

Select the result from the following list:

- Court decision will be overturned or changed: "YES"
- Court decision will be confirmed: "NO"

Response Format

⁶https://docs.mistral.ai/capabilities/completion/prompting_capabilities

Your answer must follow this format:

```
{"will_be_overturned": <result>}
""
```

Preto sme pri tomto modeli ďalej nepokračovali s technikou few-shot prompting a nebudeme túto časť zahŕňať do vyhodnotenia v ďalšej kapitole.

Pre few-shot prompting pri modeli Qwen bol ponechaný vyladený prompt a pridané dva príklady rozhodnutí, jedno potvrdené a jedno zrušené. Teda k promptu sa pridalo:

```
prompt += "Here are few examples:\n"
```

```
for obj in samples:
    text = obj.get("text","")
    label = obj.get("label","")
    label = "YES" if label else "NO"
    prompt += "\nCourt Decision: "+ text
            + "\nWill be overturned: " + label + "\n\n"
```

Generovaný text a skutočná trieda boli spolu zapisované ako jeden objekt pre každý príklad do JSON súboru pre následné vyhodnotenie.

Dotrénovanie pre klasifikáciu sekvencie

Pre klasifikáciu sme použili taktiež dotrénované modely, na ktoré sme pridali ďalšiu vrstvu (klasifikačnú hlavu), ktorá transformuje výstup z poslednej skrytej vrstvy modelu (hidden state) na požadovaný počet tried. Klasifikátor je binárny, teda má dve výstupné triedy, a to 0, ak rozhodnutie bude potvrdené a 1, ak bude zrušené alebo zmenené. Model bol nakonfigurovaný na používanie dátového typu bfloat16 a implementácie Flash Attention 2 na optimalizáciu výpočtovej efektívnosti a pamäťových nárokov pri práci na GPU. Taktiež ako pri prvotnom doladovaní modelu sme využili techniku LoRA (Low-Rank Adaptation) s rovnakým nastavením rank na hodnotu 16, ktorá modifikuje iba projekčné vrstvy dotazov a hodnôt (q_proj a v_proj) s dropout hodnotou 0,05, čo umožňuje efektívne doladenie modelu pri minimálnom počte trénovateľných parametrov. Pre trénovanie bolo použitých 80% zo spomínanej vzorky súdnych rozhodnutí kratších ako 3 000 tokenov, a zvyšných 20% sa rozdelilo rovným dielom medzi validačnú a testovaciu množinu. Všetky rozdelenia prebehli stratifikovaným spôsobom, čo zabezpečuje zachovanie pôvodného pomeru tried vo všetkých podmnožinách. Trénovací proces prebiehal s veľkosťou batch-u 1, pričom parame-

ter `GRADIENT_ACCUMULATION_STEPS` bol nastavený na hodnotu 8, čo efektívne zodpovedá dávkovej veľkosti 8, s tromi epochami tréovania, rýchlosťou učenia $2e-5$ a optimizérom AdamW.

4 Vyhodnotenie

Táto sekcia sa venuje záverečnému vyhodnoteniu práce v rámci implementačnej časti a diskusii výsledkov.

4.1 Generovanie výroku

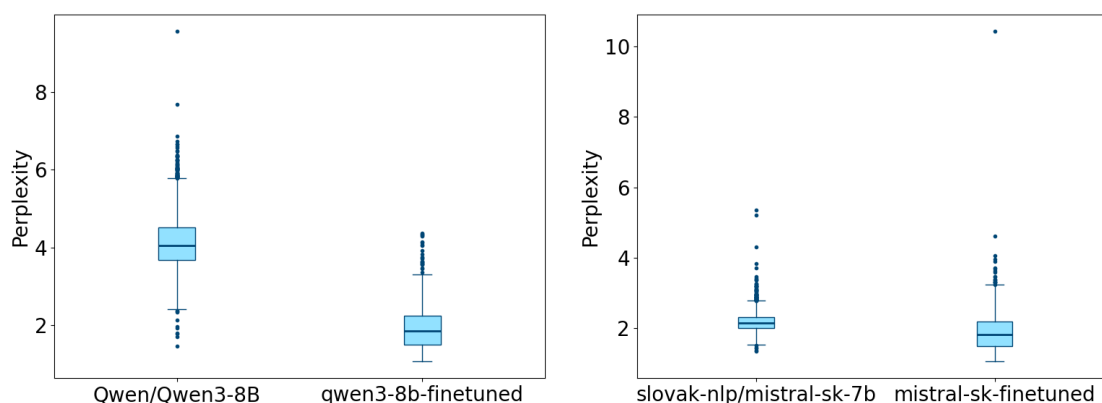
Pri úlohe generovania výroku z odôvodnenia sa pozrieme na zlepšenie jazykovej reprezentácie po finetuningu, kvantitatívne pomocou metrík. Rozhodnutia, ktoré dosahovali nízke hodnoty metrík, si bližšie popíšeme.

4.1.1 Kvantitatívne porovnanie

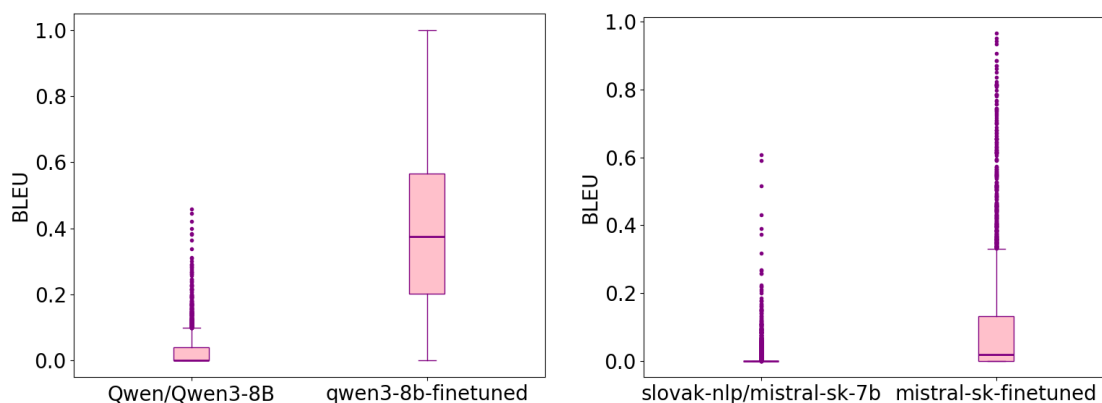
Dotrénované modely boli vyhodnotené pomocou metrík BLEU, ROUGE a Perplexity na validačnej množine približne 3300 textov dlhších ako 2048 tokenov náhodne vybraných z korpusu, aby sme potvrdili zlepšenie schopnosti modelu generovať formulácie právneho charakteru. Generované texty pôvodného aj dotrénovaného modelu boli porovnávané oproti výroku (keďže pri tréningu bola použitá len časť výroku bez úvodnej časti rozsudku). Ako môžeme vidieť na nasledujúcich boxplotoch (Obr. 4.1 až 4.5), zlepšenie je výraznejšie u modelu Qwen než u modelu mistral-sk.

Na Obr. 4.1 vidíme porovnanie metriky Perplexity vo forme boxplotu, medián sa znížil u modelu Qwen približne o polovicu, pričom variabilita sa veľmi nezmenila. U modelu mistral-sk sa hodnoty Perplexity líšia len nepatrne, nakoľko tento model už bol dotrénovaný na slovenských textoch. Na Obr. 4.2 je zobrazená distribúcia hodnôt metriky BLEU, pričom u modelu Qwen sa medián zvýšil z 0,0 na približne 0,4 a variabilita je vyššia, dotrénovaný model dosahuje maximum 1, čo je perfektný výsledok znamenajúci, že prekrytie s referenčným textom je 100%, pri pôvodnom modeli bolo maximum okolo 50%, aj to len ako odľahlá hodnota. U dotrénovaného modelu mistral-sk až také zvýšenie hodnôt BLEU nebolo zaznamenané, medián sa zvýšil len nepatrne, ale zvýšila sa však variabilita, pričom hodnoty presahujúce 0,4 predstavujú výlučne odľahlé pozorovania. Obr.

4.3 až 4.5 zobrazujú porovnanie hodnôt metriky ROUGE, takisto ako pri BLEU tu môžeme vidieť výrazné zlepšenie mediánových hodnôt (2 až 3-násobné), ale aj zvýšenie variability, a 100%-nú maximálnu hodnotu vo všetkých prípadoch u modelu Qwen. U modelu mistral-sk znova ako aj pri metrike BLEU došlo len k nepatrnému zvýšeniu hodnôt a výraznému zvýšeniu variability, pričom hodnoty presahujúce 0,8 predstavujú výlučne odľahlé pozorovania.



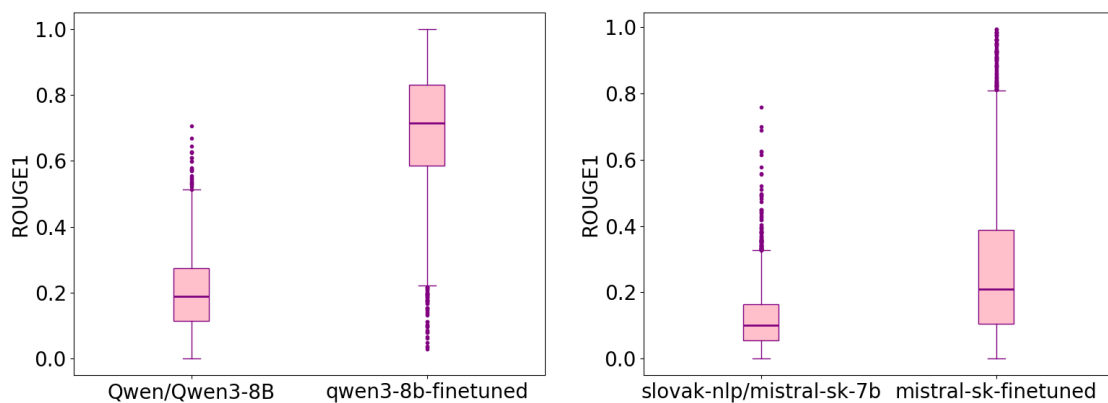
Obrázok 4.1: Porovnanie Perplexity score.



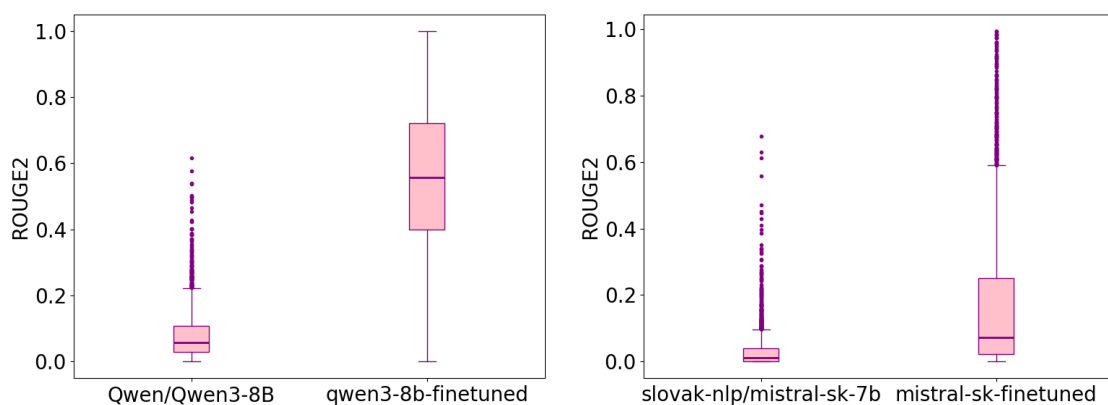
Obrázok 4.2: Porovnanie BLEU score.

4.1.2 Kvalitatívne vyhodnotenie

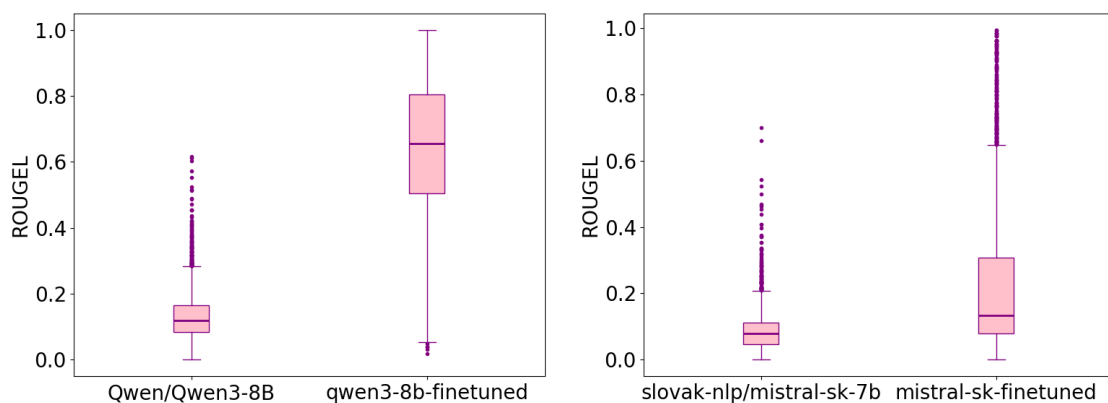
Pri kvalitatívnom vyhodnotení sa pozrieme na vybrané vygenerované rozsudky a porovnáme ich s reálnymi rozsudkami. V prvom rade sa bližšie pozrieme na rozsudky, ktoré mali najmenšie prekrytie s referenčným textom na základe ROUGE-L metriky, teda dosahovali hodnoty nižšie ako 0,1. Dokopy ich bolo 22, pričom išlo o 19 rozhodnutí okresných súdov a 3 rozhodnutia krajských súdov. Podľa povahy rozhodnutia ide prevažne o prvostupňové rozhodnutia nenapadnuté



Obrázok 4.3: Porovnanie ROUGE-1 score.



Obrázok 4.4: Porovnanie ROUGE-2 score.



Obrázok 4.5: Porovnanie ROUGE-L score.

oprávnymi prostriedkami (15), tri potvrdené, jedno zmenené, jedno zrušené rozhodnutie, jedno rozhodnutie zastavujúce odvolacie konanie a jedno rozhodnutie s neuvedenou povahou. V deviatich prípadoch ide o občianske právo, čo korešponduje s tým, že je to najfrekventovanejšia oblasť právnej úpravy, no v siedmich prípadoch ide o trestné právo, ktoré je v celom korpuse oveľa menej frekventované. Rozsudky v trestnom práve sú špecifické oproti iným tým, že sa vo výroku detailne popisujú skutky, ktoré obžalovaný podľa súdu spáchal, resp. nespáchal. Mali by byť teda dlhšie a teda ak máme natrénovaný model pre generovanie výroku, resp. hlavne verdiktu, tento môže byť totožný s referenčným výrokom, aj keď nepopisuje skutky detailne.

Keď sa pozrieme na texty rozsudkov, ide o dlhšie rozsudky týkajúce sa exekučného konania, kde sú vo výroku detailne popísané dlžné sumy a dátumy splatnosti, alebo o uzavretia zmieru pri rozvoze, kde sa popisuje celé vysporiadanie bezpodielového spoluvlastníctva manželov. Ďalej ide o spomínané trestné konania, neodkladné opatrenie a zahľadenie odsúdenia. Čo sa týka samotného verdiktu súdu (hlavnej časti výroku), tá sa v deviatich rozhodnutiach napriek celkovému nízkemu pokrytiu zhoduje s reálnym rozhodnutím. Môžeme to badať na nasledovných príkladoch:

Generovaný text:

Žalobu vo zvyšku sa zamietá. Žalovaný je povinný nahradiť žalobcovi trovy konania vo výške 70,30 € na účet právneho zástupcu žalobcu do troch dní od právoplatnosti rozsudku.

Referenčný text:

I. Žalovaný je žalobcovi povinný zaplatiť sumu 305,80 € spolu s - 8,75 % úrokom z omeškania ročne zo sumy 13,90 € od 03. 11. 2012 do zaplatenia - 8,75 % úrokom z omeškania ročne zo sumy 13,90 € od 21. 11. 2012 do zaplatenia, - 8,75 % úrokom z omeškania ročne zo sumy 13,90 € od 21. 12. 2012 do zaplatenia, - 8,75 % úrokom z omeškania ročne zo sumy 13,90 € od 21. 01. 2013 do zaplatenia, - 8,75 % úrokom z omeškania ročne zo sumy 13,90 € od 21. 02. 2013 do zaplatenia,

....

všetko do troch dní od právoplatnosti rozsudku.II. **Vo zvyšku uplatneného nároku sa žaloba zamietá.** III. Žalovaný je žalobcovi povinný nahradiť trovy konania v sume 70,30 € do troch dní od právoplatnosti rozsudku.

Generovaný text:

E. **schvaľuje zmier** v nasledujúcom znení: E. spoločný majetok strán sporu zaniknutý v dôsledku rozvodu manželstva strán sporu dňa XX.XX.XXXX v konaní o rozvod manželstva vedenom na E. súde Z. pod sp. zn. XXC/XXX/ tak, že do výlučného vlastníctva žalobcu patria nehnuteľnosti zapísané na LV č. XXXX, XXXX, XXXX, XXXX, XXXX, XXXX, ...

Referenčný text:

Súd schvaľuje súdny zmier nasledovného znenia:

I. Z vecí patriacich do bezpodielového spoluvlastníctva strán sporu do výlučného vlastníctva žalobcu pripadnú:

a) byt číslo XX nachádzajúci sa na X.p. bytového domu na ulici H. 3, vchod: H. 3, súpisné číslo XXXX, vybudovaný na parcele registra "C" parcelné č. XXXX - Q. plochy a nádvoria o výmere XXX m², parcele registra "C" parcelné č. XXXX - Q. plochy a nádvoria o výmere XXX m² a parcele registra "C" parcelné č. XXXX - Q. plochy a nádvoria o výmere XXX m², ktorý je zapísaný na liste vlastníctva číslo XXXX pre katastrálne územie D., obec BA - m.č. D., okres R. V, list vlastníctva je evidovaný O. úradom R. - katastrálnym odborom;

b) spoluvlastnícky podiel vo výške XXXX/XXXXXX na spoločných častiach a spoločných zariadeniach bytového domu ...

...

4.2 Klasifikátor na predikciu úspešnosti odvolania

V tejto sekcii sú porovnané výsledky zero-shot prompting, few-shot prompting a natrénovaného klasifikátora. Vyhodnotenie bolo realizované na tej istej testovacej množine 2014 rozhodnutí. Pre každý uložený záznam zo zero-shot a few-shot prompting v súbore vo formáte JSON bola vyhodnotená zhoda modelu so skutočnou triedou, teda za zhodu sa považovalo YES a 1, a NO a 0, prípadne kľúčové slová spomenuté v implementačnej časti („zrušuje“, „mení“ alebo „vracia“, „potvrďuje“ či „zamietá“). Na základe toho sme si pre komplexné vyhodnotenie kvality klasifikácie zobrazili maticu zámen (confusion matrix), ktorá vizualizuje rozdelenie správne a nesprávne klasifikovaných vzoriek pre obe triedy. V matici sa nachádza aj hodnota -1, ktorá znamená, že v odpovedi modelu sa nenachádzala žiadna z tried. Ďalej sme klasifikáciu vyhodnotili pomocou metrick presnosť (precision), návratnosť (recall) a F1-skóre a podpory pre každú triedu.

du samostatne, ako aj ich vážené a makro priemery, čo poskytuje detailný obraz o výkonnosti modelu pri binárnej klasifikácii súdnych rozhodnutí. Ich hodnoty môžeme vidieť v Tab. 4.1 až 4.4.

Klasifikáciu zero-shot a few-shot prompting metódou sme porovnali pre „base“ model Qwen aj náš dotrénovaný model Qwen. Pri base modeli sa výsledné metriky u few-shot klasifikácie veľmi nelíšili od zero-shot klasifikácie, celková úspešnosť bola okolo 0,67, až na nízku návratnosť a F1-skóre, ktoré sa mierne zvýšili pri few-shot metóde pre triedu 0 (potvrdené), ale znížili sa pre triedu 1 (zrušené) na hodnoty 0,15 a 0,25. Náš dotrénovaný model pri zero-shot prompting klasifikácii síce dosiahol horšie výsledky ako base model, ale išlo hlavne o to, že v 70% prípadov mal nesprávne naformátovaný výstup, t.j. v 70% príkladov model neurčil žiadnu z tried. Pri few-shot klasifikácii sa toto výrazne zredukovalo a zlepšilo. Model v tomto prípade neurčil žiadnu z tried iba v necelých 8% príkladov. Celková presnosť vzrástla na 60%, čo predstavuje markantný nárast oproti zero-shot variantu. U triedy 0 došlo k podstatnému zlepšeniu návratnosti na hodnotu 0,84, čo znamená, že model správne zachytil väčšinu potvrdených prípadov, avšak na úkor zníženia hodnoty presnosti na 0,66, pričom F1-skóre dosiahlo hodnotu 0,74. Trieda 1 (zrušené) zostala však problematická, s návratnosťou iba 0,21 a F1-skóre 0,31, čo naznačuje, že model mal naďalej výrazné ťažkosti so správnou identifikáciou zrušených prípadov. Vážený priemer F1-skóre dosiahol hodnotu 0,57, čo predstavuje výrazné zlepšenie oproti zero-shot variante. Celkovo možno konštatovať, že pridanie ukázkových príkladov (few-shot) malo pre dotrénovaný model pozitívny efekt. Napriek tomu model vykazuje obmedzenú schopnosť správne detekovať zrušené a zmenené rozhodnutia, čo limituje jeho praktickú použiteľnosť pri reálnej klasifikácii. Ďalej sme teda použili náš dotrénovaný model a preučili ho na úlohu klasifikácie.

Naproti tomu dotrénovaním binárneho klasifikátora na našom modeli Qwen sa podstatne zlepšila hlavne klasifikácia zrušených a zmenených rozhodnutí, keďže presnosť, návratnosť aj F1-skóre dosahujú hodnoty vyššie ako 70%, ale aj klasifikácia potvrdených rozhodnutí, kde hodnoty metrík presahujú 80%. Ako bolo spomenuté v implementačnej časti, zero-shot a few-shot klasifikácia u nášho modelu mistral-sk nepriniesla relevantné výsledky, no pre porovnanie sme binárny klasifikátor dotrénovali aj na modeli mistral-sk. Výsledné hodnoty metrík sú takmer totožné s hodnotami u modelu Qwen. Môžeme ich vidieť v Tab. 4.5 a 4.6.

Tabuľka 4.1: Matice zámen pre klasifikáciu (Qwen3-8B).

(a) zero-shot			(b) few-shot		
true\pred	0	1	true\pred	0	1
0	1150	92	0	1208	34
1	579	193	1	658	114

Tabuľka 4.2: Matice zámen pre klasifikáciu (dotrénovaný Qwen).

(a) zero-shot				(b) few-shot			
true\pred	-1	0	1	true\pred	-1	0	1
-1	0	0	0	-1	0	0	0
0	890	210	142	0	94	1045	103
1	521	73	178	1	64	548	160

Tabuľka 4.3: Metriky vyhodnotenia pre zero-shot klasifikáciu modelu Qwen.

	Presnosť		Návratnosť		F1-skóre	
	Base	Fine-tuned	Base	Fine-tuned	Base	Fine-tuned
0 (potvrdené)	0.67	0.74	0.93	0.17	0.77	0.28
1 (zrušené)	0.68	0.56	0.25	0.23	0.37	0.33
Makro priemer	0.67	0.43	0.59	0.13	0.57	0.20
Vážený priemer	0.67	0.67	0.67	0.19	0.62	0.29
Úspešnosť	0.67	0.19	–	–	–	–

Tabuľka 4.4: Metriky vyhodnotenia pre few-shot klasifikáciu modelu Qwen.

	Presnosť		Návratnosť		F1-skóre	
	Base	Fine-tuned	Base	Fine-tuned	Base	Fine-tuned
0 (potvrdené)	0.65	0.66	0.97	0.84	0.78	0.74
1 (zrušené)	0.77	0.61	0.15	0.21	0.25	0.31
Makro priemer	0.71	0.42	0.56	0.35	0.51	0.35
Vážený priemer	0.69	0.64	0.66	0.60	0.57	0.57
Úspešnosť	0.66	0.60	–	–	–	–

Tabuľka 4.5: Matice zámen pre dotrénovaný binárny klasifikátor.

(a) qwen			(b) mistral-sk		
true\pred	0	1	true\pred	0	1
0	1058	184	0	1052	190
1	217	555	1	196	576

Tabuľka 4.6: Metriky vyhodnotenia pre binárne klasifikátory.

	Presnosť		Návratnosť		F1-skóre	
	<i>mistral-sk</i>	<i>Qwen</i>	<i>mistral-sk</i>	<i>Qwen</i>	<i>mistral-sk</i>	<i>Qwen</i>
0 (potvrdené)	0.84	0.83	0.85	0.85	0.84	0.84
1 (zrušené)	0.75	0.75	0.75	0.72	0.75	0.73
Makro priemer	0.80	0.79	0.80	0.79	0.80	0.79
Vážený priemer	0.81	0.80	0.81	0.80	0.81	0.80
Úspešnosť	0.81	0.80	–	–	–	–

4.3 Diskusia

Všetky tri hypotézy stanovené na konci kap. 2 boli v rámci experimentálneho overovania potvrdené. Prvá hypotéza predpokladala, že medzi vybranými jazykovými modelmi existujú merateľné rozdiely v kvalite reprezentácie textu rozhodnutí slovenských súdov. Hodnotenie deviatich modelov pomocou metrík BLEU, ROUGE a Perplexity preukázalo štatisticky merateľné rozdiely medzi jazykovými modelmi, čo umožnilo identifikovať a vybrať dvoch najvhodnejších kandidátov pre následnú doménovú adaptáciu. Druhá hypotéza predpokladala, že dotrénovanie vybraných modelov na korpuse slovenských súdnych rozhodnutí povedie k merateľnému zlepšeniu jazykovej reprezentácie. Toto zlepšenie bolo potvrdené nárastom hodnôt metrík BLEU a ROUGE a poklesom Perplexity v porovnaní so základnými modelmi, čo svedčí o úspešnej doménovej adaptácii. Tretia hypotéza predpokladala, že model s doménovou adaptáciou dosiahne pri predikcii úspešnosti odvolania úspešnosť aspoň 80%. Tu boli preverené tri prístupy (zero-shot prompting, few-shot prompting a pridanie vrstvy pre klasifikáciu sekvencie), pričom táto hypotéza je platná len pre tretí prístup, ako bolo popísané vyššie.

Samozrejme, aj napriek dosiahnutým pozitívnym výsledkom je potrebné po-

ukázať na určité limitácie. Jazykový model bol trénovaný na textovom korpuse rôznorodých rozhodnutí zo všetkých právnych oblastí a stupňov súdov. Ako sa ukázalo aj pri kvalitatívnom vyhodnotení, model mal problémy so špecifickými rozhodnutiami v rodinnom práve a trestnom práve, a preto by bolo dobré zamerať sa na špecifický typ právnej oblasti pri trénovaní na konkrétnu úlohu.

Odôvodnenie síce poskytuje dobrý základ a sumarizovaný obraz o skutočnostiach v konaní, keďže popisuje skutkový stav a priebeh konania. Avšak je to stále text napísaný človekom (sudcom) a nie konkrétny dôkazový materiál predkladaný jednotlivými stranami sporu. To znamená, že pokiaľ niektoré skutočnosti, resp. referencie na dôkazy, sudca nezahrnul do odôvodnenia, pre model to znamená, že neexistujú, a teda to môže mať dopad na výsledný vygenerovaný verdikt. Takisto môže odôvodnenie obsahovať náznaky výsledného verdiktu. Pri použití veľkého jazykového modelu to však až tak nevadí, keďže taký model sa učí rozloženie pravdepodobností nad celou sekvenciou tokenov a zachytáva štruktúru a jazyk textu ako celku, pričom verdikt je len ďalší token v sekvencii. Najideálnejšiu situáciu by predstavovalo priame použitie dôkazových materiálov a podaní pri trénovaní modelu na rozhodovanie verdiktu súdu, avšak takéto materiály sú väčšinou v rôznych formátoch a častokrát veľmi obsiahle, pričom kontext jazykových modelov je limitovaný, resp. väčší kontext si vyžaduje aj väčšie výpočtové zdroje, čo predstavuje aj vyššie náklady na ich prevádzku. Navyše materiály pravdepodobne obsahujú citlivé údaje, ktoré podliehajú ochrane, čo si vyžaduje špecifický prístup s ohľadom na bezpečnosť a systémový prístup k takýmto zdrojom.

Ďalším bodom je špecifickosť právnej oblasti. Veľa textov je aj po určitej dobe stále platných a pravdivých, to však nemusí platiť pre zákony a nariadenia, ktoré sú kontinuálne upravované a menené legislatívnou činnosťou štátneho orgánu povereného výkonom zákonodarnej moci. Keďže korpus obsahuje rozhodnutia za časové obdobie niekoľkých rokov, môže obsahovať aj kontradiktórne rozhodnutia, kedy napr. v minulosti istý skutok bol trestným činom, ale v priebehu rokov došlo k zmene v legislatíve, a teda v súčasnosti tomu už tak nie je a vice versa. Aj preto teraz získava v právnej oblasti popularitu prístup **RAG**, kedy jazykový model generuje odpoveď na základe bázy znalostí, ktorú mu používateľ poskytne, resp. definuje. Za takúto bázu znalostí sa môže považovať aj súčasné znenie platnej legislatívy.

V súvislosti s úspešnosťou odvolania v tejto práci nebolo skúmané, ktorá strana odvolanie podala a v čí prospech bolo rozhodnutie zmenené. V právnej praxi je však zaužívané a vysoko pravdepodobné, že odvolanie podáva neúspešná

strana sporu, pričom sa uplatňuje princíp zákazu „**Reformatio in peius**“, podľa ktorého odvolacie konanie nesmie vyústiť do stavu horšieho, ako bol stav odvolateľa pred podaním odvolania [71]. Výsledok konania sa teda môže pohybovať len v rozsahu od zachovania pôvodného stavu po jeho zlepšenie.

5 Záver

Táto diplomová práca sa zaoberala využitím veľkých jazykových modelov pre analýzu rozhodnutí súdov SR, konkrétne predikciu výroku a predikciu úspešnosti odvolania. Kapitola analýzy priblížila súčasné prístupy v oblasti umelej inteligencie v právnej oblasti a doterajšie štúdie vykonané na rozhodnutiach slovenských súdov. Časť implementácie zahŕňala popis korpusu rozhodnutí, ich predspracovanie a rozdelenie na logické časti, porovnanie výkonnosti jazykových modelov pre predikciu rozsudku a proces trénovania samotných modelov. Vyhodnocovania časť kvantitatívne aj kvalitatívne popisuje otestovanie natrénovaných modelov na testovacej množine. Všetky body zadávacieho listu teda boli splnené a vytvorený zdrojový kód spolu s popisom jeho častí je verejne dostupný, ako je uvedené v časti príloh.

Dotrénovaním jazykových modelov na korpuse súdnych rozhodnutí sa podarilo zvýšiť ich jazykovú reprezentáciu v právnej doméne a zachytiť charakteristické vzory rozhodovacej praxe slovenských súdov. Takéto modely môžu v budúcnosti plniť úlohu nástroja občianskej kontroly nad súdnou mocou a zároveň slúžiť ako dostupná pomôcka pre bežného človeka pri orientácii v právnom prostredí.

Praktická aplikácia klasifikátora na predikciu úspešnosti odvolania môže byť obzvlášť prínosná pri rozhodovaní, či odvolanie vôbec podať. Každé odvolacie konanie totiž predstavuje dodatočné náklady nielen pre samotného účastníka v podobe súdneho poplatku a právneho zastúpenia, ale aj pre štát v podobe výdavkov spojených s ďalším súdnym konaním. Nasadenie takéhoto nástroja by preto mohlo prispieť k efektívnejšiemu využívaniu súdnych kapacít a zníženiu celkových nákladov.

Možným rozšírením do budúcnosti je spojenie modelov s metódou RAG a pripojením súčasnej platnej legislatívy ako bázy znalostí, alebo aj vytvorenie aplikácie pre koncového používateľa na jednoduchý a intuitívny prístup k použitiu modelov.

Zoznam použitej literatúry

1. TONN, Bruce; STIEFEL, Dorian. The future of governance and the use of advanced information technologies. *Futures*. 2012, roč. 44, č. 9, s. 812–822. ISSN 0016-3287. Dostupné z DOI: <https://doi.org/10.1016/j.futures.2012.07.004>. Futures of Governance.
2. TASR; RTVS. *Problémy súdnej mapy: Bratislavský správny súd má nedostatok sudcov, podobne aj iné súdy*. 2023-07. Dostupné tiež z: <https://spravy.stvr.sk/2023/07/problemy-sudnej-mapy-bratislavsky-spravny-sud-ma-nedostatok-sudcov-podobne-aj-ine-sudy/>. Aktualizované: 7. 7. 2023 19:42:00.
3. WILFLING, Peter. Prístup verejnosti k súdnym rozhodnutiam na Slovensku. In: *Presvedčivosť a transparentnosť rozhodovania súdov*. Bratislava: VIA IURIS – Centrum pre práva občana, 2010, s. 14–15. Dostupné tiež z: <https://viaiuris.sk/wp-content/uploads/2017/08/pilk-2009.pdf>.
4. MRVA, Michal. Sudca a právna istota. In: GÁBRIŠ, Tomáš (ed.). *Sudcovské rozhodovanie: Záruky a prekážky spravodlivého procesu*. Bratislava: Wolters Kluwer, 2018, s. 113. Dostupné tiež z: https://www.flaw.uniba.sk/fileadmin/praf/Pracoviska/Katedry/KTPSV/obrazky/Zbornik_prispevkov_Sudcovske_rozhodovanie.pdf.
5. AMMAR, Adel; KOUBAA, Anis; BENJDIRA, Bilel; NACAR, Omer; SIBAE, Serry. Prediction of Arabic Legal Rulings Using Large Language Models. *Electronics*. 2024, roč. 13, č. 4. ISSN 2079-9292. Dostupné z DOI: [10.3390/electronics13040764](https://doi.org/10.3390/electronics13040764).
6. VIRTUCIO, Michael Benedict L.; ABOROT, Jeffrey A.; ABONITA, John Kevin C.; AVIÑANTE, Roxanne S.; COPINO, Rother Jay B.; NEVERIDA, Michelle P.; OSIANA, Vanesa O.; PERAMO, Elmer C.; SYJUCO, Joanna G.; TAN, Glenn Brian A. Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning. In: *Proceedings of the 2018 42nd Annual Computer Software and Applications Conference (COM-*

- PSAC). 2018, zv. 02, s. 130–135. Dostupné z DOI: [10.1109/COMPSAC.2018.10348](https://doi.org/10.1109/COMPSAC.2018.10348).
7. ALMUZAINI, Huda A.; AZMI, Aqil M. TaSbeeb: A judicial decision support system based on deep learning framework. *Journal of King Saud University - Computer and Information Sciences*. 2023, roč. 35, č. 8, s. 101695. ISSN 1319-1578. Dostupné z DOI: [10.1016/j.jksuci.2023.101695](https://doi.org/10.1016/j.jksuci.2023.101695).
8. CHAPHALKAR, N.B.; IYER, K.C.; PATIL, Smita K. Prediction of outcome of construction dispute claims using multilayer perceptron neural network model. *International Journal of Project Management*. 2015, roč. 33, č. 8, s. 1827–1835. ISSN 0263-7863. Dostupné z DOI: [10.1016/j.ijproman.2015.09.002](https://doi.org/10.1016/j.ijproman.2015.09.002).
9. CONRAD, Jack G.; AL-KOFAHI, Khalid. Scenario analytics: analyzing jury verdicts to evaluate legal case outcomes. In: *Proceedings of the 16th Edition of the International Conference on Artificial Intelligence and Law*. London, United Kingdom: Association for Computing Machinery, 2017, s. 29–37. ICAIL '17. ISBN 9781450348911. Dostupné z DOI: [10.1145/3086512.3086516](https://doi.org/10.1145/3086512.3086516).
10. BAKUROVA, ANNA; PASICHNYK, MARIIA; TERESCHENKO, ELINA; FILEI, YURII. Production model for administration of judicial decisions in the case of theft. *SHS Web Conf.* 2019, roč. 65, s. 04012. Dostupné z DOI: [10.1051/shsconf/20196504012](https://doi.org/10.1051/shsconf/20196504012).
11. METSKER, Oleg; TROFIMOV, Egor; GRECHISHCHEVA, Sofia. Natural Language Processing of Russian Court Decisions for Digital Indicators Mapping for Oversight Process Control Efficiency: Disobeying a Police Officer Case. In: CHUGUNOV, Andrei; KHODACHEK, Igor; MISNIKOV, Yuri; TRUTNEV, Dmitrii (ed.). *Electronic Governance and Open Society: Challenges in Eurasia*. Cham: Springer International Publishing, 2020, s. 295–307. ISBN 978-3-030-39296-3. Dostupné z DOI: [10.1007/978-3-030-39296-3_22](https://doi.org/10.1007/978-3-030-39296-3_22).
12. HABBA, Eliya; KEYDAR, Renana; BAREKET, Dan; STANOVSKY, Gabriel. The Perfect Victim: Computational Analysis of Judicial Attitudes towards Victims of Sexual Violence. In: *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*. Braga, Portugal: Association for Computing Machinery, 2023, s. 111–120. ICAIL '23. Dostupné z DOI: [10.1145/3594536.3595168](https://doi.org/10.1145/3594536.3595168).
13. MUNTHULI, Adirek; SOCATIYANURAK, Vorada; SANGCHOCANONTA, Sirikorn; KOVUDHIKULRUNGSRI, Lalin; SAKSAKULKUNAKORN, Nantawat; CHAIRUANGSRI, Phornkanok; TANTIBUNDHIT, Charturong. Trans-

- formers for Multi-Intent Classification and Slot Filling of Supreme Court Decisions Related to Sexual Violence Law. *IEEE Access*. 2023, roč. 11, s. 76448–76467. Dostupné z doi: [10.1109/ACCESS.2023.3296261](https://doi.org/10.1109/ACCESS.2023.3296261).
14. ANANTATHANAVIT, Rungpilin; CHONGTHANAKORN, Jomchai; KONG-SANTINART, Warisa; PRAIWATTANA, Pisit; THAIPISUTIKUL, Tipajin. Utilizing AI and Natural Language Processing to Predict Supreme Court Decisions in Thailand. In: *Proceedings of the 16th International Conference on Knowledge and Smart Technology (KST)*. 2024, s. 45–50. Dostupné z doi: [10.1109/KST61284.2024.10499665](https://doi.org/10.1109/KST61284.2024.10499665).
15. VARGA, Dávid; SZOPLÁK, Zoltán; SOKOL, Pavol; KRAJČI, Stanislav; GURSKÝ, Peter. Analysis and Prediction of Legal Judgements in the Slovak Criminal Proceedings. In: *Proceedings of Conference on Theory and Practice of Information Technologies*. 2021. Dostupné tiež z: <https://ceur-ws.org/Vol-2962/paper13.pdf>.
16. SZOPLÁK, Zoltán; VARGA, Dávid; HORVÁT, Šimon; GURSKÝ, Peter; ANTONI, Ľubomír; KRÍDLO, Ondrej; KRAJČI, Stanislav. Analysis of Slovak Court Decisions by Formal Concept Analysis and Machine Learning Methods. In: OJEDA-ACIEGO, Manuel; SAUERWALD, Kai; JÄSCHKE, Robert (ed.). *Graph-Based Representation and Reasoning*. Cham: Springer Nature Switzerland, 2023, s. 201–204. Dostupné z doi: [10.1007/978-3-031-40960-8_18](https://doi.org/10.1007/978-3-031-40960-8_18).
17. VARGA, Dávid; HORVÁT, Šimon; SZOPLÁK, Zoltán; ANTONI, Ľubomír; KRAJČI, Stanislav; GURSKÝ, Peter; RÓZENFELDOVÁ, Laura Bachňáková. Keyphrase Extraction from Slovak Court Decisions. In: *Proceedings of ITAT'22: Information Technologies – Applications and Theory Conference, September 23–27, 2022, Zuberec, Slovakia*. 2022. Dostupné tiež z: <https://ceur-ws.org/Vol-3226/paper16.pdf>.
18. VARGA, Dávid; GOJDIČ, Martin; SZOPLÁK, Zoltán; GURSKÝ, Peter; HORVÁT, Šimon; KRAJČI, Stanislav; ANTONI, Ľubomír. Extraction of Legal References from Court Decisions. In: *Proceedings of ITAT'23: Information Technologies – Applications and Theory Conference, September 22–26, 2023, Tatranské matliare, Slovakia*. 2022. Dostupné tiež z: <https://ceur-ws.org/Vol-3226/paper16.pdf>.
19. SZOPLÁK, Zoltán; GURSKÝ, Peter; HORVÁT, Šimon; VARGA, Dávid; KRAJČI, Stanislav. Automated Construction of Legal Terminology Thesaurus. In: *Proceedings of ITAT'25: Information Technologies – Applications and Theory Con-*

- ference, September 26--30, 2025, Telgárt, Slovakia. Telgárt, Nízke Tatry, Slovakia, 2025, zv. 4092. Dostupné tiež z: <https://ceur-ws.org/Vol-4092/paper11.pdf>.
20. HARAŠTA, Jakub; ŠAVELKA, Jaromír; KASL, František; KOTKOVÁ, Adéla; LOUTOCKÝ, Pavel; MÍŠEK, Jakub; PROCHÁZKOVÁ, Daniela; PULLMANOVÁ, Helena; SEMENIŠÍN, Petr; ŠEJNOVÁ, Tamara; ŠIMKOVÁ, Nikola; VOSINEK, Michal; ZAVADILOVÁ, Lucie; ZIBNER, Jan. Annotated Corpus of Czech Case Law for Reference Recognition Tasks. In: SOJKA, Petr; HORÁK, Aleš; KOPEČEK, Ivan; PALA, Karel (ed.). *Text, Speech, and Dialogue*. Cham: Springer International Publishing, 2018, s. 239–250. ISBN 978-3-030-00794-2. Dostupné z DOI: [10.1007/978-3-030-00794-2_26](https://doi.org/10.1007/978-3-030-00794-2_26).
21. COELLO, Juan; NETO, Armando. Sentiment Analysis of Tweets in Brazilian Portuguese with Convolutional Neural Networks. *International Journal of Innovation Education and Research*. 2019, roč. 7, s. 29–41. Dostupné z DOI: [10.31686/ijer.Vol7.Iss6.1547](https://doi.org/10.31686/ijer.Vol7.Iss6.1547).
22. CHAI, Christine P. Comparison of text preprocessing methods. *Natural Language Engineering*. 2023, roč. 29, č. 3, s. 509–553. ISBN 13513249. Dostupné tiež z: <https://www.proquest.com/scholarly-journals/comparison-text-preprocessing-methods/docview/2815043087/se-2>. Copyright - © The Author(s), 2022. Published by Cambridge University Press; Last updated - 2024-10-03.
23. JOHNSON, S. J.; MURTY, M. R.; NAVAKANTH, I. A detailed review on word embedding techniques with emphasis on word2vec. *Multimedia Tools and Applications*. 2024, roč. 83, s. 37979–38007. Dostupné z DOI: [10.1007/s11042-023-17007-z](https://doi.org/10.1007/s11042-023-17007-z).
24. DAI, Shuying; LI, Keqin; LUO, Zhuolun; ZHAO, Peng; HONG, Bo; ZHU, Armando; LIU, Jiabei. AI-based NLP section discusses the application and effect of bag-of-words models and TF-IDF in NLP tasks. *Journal of Artificial Intelligence General science (JAIGS) ISSN:3006-4023*. 2024, roč. 5, č. 1, s. 13–21. Dostupné z DOI: [10.60087/jaigs.v5i1.149](https://doi.org/10.60087/jaigs.v5i1.149).
25. HOFMANN, Thomas. Probabilistic latent semantic indexing. In: *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Berkeley, California, USA: Association for Computing Machinery, 1999, s. 50–57. SIGIR '99. ISBN 1581130961. Dostupné z DOI: [10.1145/312624.312649](https://doi.org/10.1145/312624.312649).

26. LIU, Zelong; LI, Maozhen; LIU, Yang; PONRAJ, Mahesh. Performance evaluation of Latent Dirichlet Allocation in text mining. In: *Proceedings of the Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. 2011, zv. 4, s. 2695–2698. Dostupné z doi: [10.1109/FSKD.2011.6020066](https://doi.org/10.1109/FSKD.2011.6020066).
27. OGUNDARE, Adedotun; SALEH, A.; JAMES, O.; AJAYI, E.; GOSTOJI, S. Performance evaluation of Latent Dirichlet Allocation on legal documents. *Applied and Computational Engineering*. 2024, roč. 52, s. 96–101. Dostupné z doi: [10.54254/2755-2721/52/20241322](https://doi.org/10.54254/2755-2721/52/20241322).
28. MAO, Xingliang; CHANG, Shuai; SHI, Jinjing; LI, Fangfang; SHI, Ronghua. Sentiment-Aware Word Embedding for Emotion Classification. *Applied Sciences*. 2019, roč. 9, č. 7. ISSN 2076-3417. Dostupné z doi: [10.3390/app9071334](https://doi.org/10.3390/app9071334).
29. KRENKER, Andrej; BEŠTER, Janez; KOS, Andrej. Introduction to the artificial neural networks. *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech*. 2011, s. 1–18. Dostupné z doi: [10.5772/15751](https://doi.org/10.5772/15751).
30. SHARMA, Siddharth; SHARMA, Simone; ATHAIYA, Anidhya. ACTIVATION FUNCTIONS IN NEURAL NETWORKS. *International Journal of Engineering Applied Sciences and Technology*. 2020, roč. 04, s. 310–316. Dostupné z doi: [10.33564/IJEAST.2020.v04i12.054](https://doi.org/10.33564/IJEAST.2020.v04i12.054).
31. LUAN, Yuandong; LIN, Shaofu. Research on Text Classification Based on CNN and LSTM. In: *Proceedings of the IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*. 2019, s. 352–355. Dostupné z doi: [10.1109/ICAICA.2019.8873454](https://doi.org/10.1109/ICAICA.2019.8873454).
32. RASAMOELINA, Andrinandrasana David; ADJAILIA, Fouzia; SINČÁK, Peter. A Review of Activation Function for Artificial Neural Network. In: *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*. 2020, s. 281–286. Dostupné z doi: [10.1109/SAMI48414.2020.9108717](https://doi.org/10.1109/SAMI48414.2020.9108717).
33. CLEVERT, Djork-Arné; UNTERTHINER, Thomas; HOCHREITER, Sepp. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In: BENGIO, Yoshua; LECUN, Yann (ed.). *Proceedings of the 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. 2016. Dostupné z doi: <https://doi.org/10.48550/arXiv.1511.07289>.

34. KIM, Yoon. Convolutional Neural Networks for Sentence Classification. In: MOSCHITTI, Alessandro; PANG, Bo; DAELEMANS, Walter (ed.). *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014, s. 1746–1751. Dostupné z doi: [10.3115/v1/D14-1181](https://doi.org/10.3115/v1/D14-1181).
35. LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. *Nature*. 2015, roč. 521, č. 7553, s. 436–444. Dostupné z doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
36. ALOYSIUS, Neena; GEETHA, M. A review on deep convolutional neural networks. In: *Proceedings of the International Conference on Communication and Signal Processing (ICCSP)*. 2017, s. 0588–0592. Dostupné z doi: [10.1109/ICCSP.2017.8286426](https://doi.org/10.1109/ICCSP.2017.8286426).
37. MIENYE, Ibomoiye Domor; SWART, Theo G.; OBAIDO, George. Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. *Information*. 2024, roč. 15, č. 9. ISSN 2078-2489. Dostupné z doi: [10.3390/info15090517](https://doi.org/10.3390/info15090517).
38. SHI, Heng; XU, Minghao; LI, Ran. Deep Learning for Household Load Forecasting — A Novel Pooling Deep RNN. *IEEE Transactions on Smart Grid*. 2018, roč. 9, č. 5, s. 5271–5280. Dostupné z doi: [10.1109/TSG.2017.2686012](https://doi.org/10.1109/TSG.2017.2686012).
39. MIENYE, Ibomoiye Domor; SUN, Yanxia. A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection. *IEEE Access*. 2023, roč. 11, s. 30628–30638. Dostupné z doi: [10.1109/ACCESS.2023.3262020](https://doi.org/10.1109/ACCESS.2023.3262020).
40. NOSOUHIAN, Shiva; NOSOUHIAN, Fereshteh; KHOSHOUEI, Abbas. A Review of Recurrent Neural Network Architecture for Sequence Learning: Comparison between LSTM and GRU. 2021. Dostupné z doi: [10.20944/preprints202107.0252.v1](https://doi.org/10.20944/preprints202107.0252.v1).
41. VASWANI, Ashish; SHAZEER, Noam; PARMAR, Niki; USZKOREIT, Jakob; JONES, Llion; GOMEZ, Aidan N; KAISER, Łukasz; POLOSUKHIN, Illia. Attention is All you Need. In: GUYON, I.; LUXBURG, U. Von; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017, zv. 30. Dostupné z doi: <https://doi.org/10.48550/arXiv.1706.03762>.
42. RAJARAMAN, V. From ELIZA to ChatGPT. *Resonance*. 2023, roč. 28, č. 6, s. 889–905. ISSN 0973-712X. Dostupné z doi: [10.1007/s12045-023-1620-6](https://doi.org/10.1007/s12045-023-1620-6).

43. LEE, Raymond M. "The most important technique ...": Carl Rogers, Hawthorne, and the rise and fall of nondirective interviewing in sociology. *Journal of the History of the Behavioral Sciences*. 2011, roč. 47, č. 2, s. 123–146. Dostupné z DOI: <https://doi.org/10.1002/jhbs.20492>.
44. JAFARI, Amir Reza; HEIDARY, Behnam; FARAHBAKHS, Reza; SALEHI, Mostafa; CRESPI, Noel. Language Models for Multi-Lingual Tasks - A Survey. *International Journal of Advanced Computer Science and Applications*. 2024, roč. 15, č. 6. Dostupné z DOI: [10.14569/IJACSA.2024.01506146](https://doi.org/10.14569/IJACSA.2024.01506146).
45. OPENAI et al. *GPT-4 Technical Report*. 2024. Dostupné z arXiv: [2303.08774 \[cs.CL\]](https://arxiv.org/abs/2303.08774).
46. OUYANG, Long; WU, Jeff; JIANG, Xu; ALMEIDA, Diogo; WAINWRIGHT, Carroll L.; MISHKIN, Pamela; ZHANG, Chong; AGARWAL, Sandhini; SLAMA, Katarina; RAY, Alex; SCHULMAN, John; HILTON, Jacob; KELTON, Fraser; MILLER, Luke; SIMENS, Maddie; ASKELL, Amanda; WELINDER, Peter; CHRISTIANO, Paul; LEIKE, Jan; LOWE, Ryan. Training language models to follow instructions with human feedback. In: *Proceedings of the 36th International Conference on Neural Information Processing Systems*. New Orleans, LA, USA: Curran Associates Inc., 2022. NIPS '22. ISBN 9781713871088. Dostupné tiež z: <https://dl.acm.org/doi/10.5555/3600270.3602281>.
47. YAO, Yifan; DUAN, Jinhao; XU, Kaidi; CAI, Yuanfang; SUN, Zhibo; ZHANG, Yue. A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly. *High-Confidence Computing*. 2024, roč. 4, č. 2, s. 100211. ISSN 2667-2952. Dostupné z DOI: <https://doi.org/10.1016/j.hcc.2024.100211>.
48. ZHOU, Ce; LI, Qian; LI, Chen; YU, Jun; LIU, Yixin; WANG, Guangjing; ZHANG, Kai; JI, Cheng; YAN, Qiben; HE, Lifang; PENG, Hao; LI, Jianxin; WU, Jia; LIU, Ziwei; XIE, Pengtao; XIONG, Caiming; PEI, Jian; YU, Philip; SUN, Li-chao. A comprehensive survey on pretrained foundation models: a history from BERT to ChatGPT. *International Journal of Machine Learning and Cybernetics*. 2024, s. 1–65. Dostupné z DOI: [10.1007/s13042-024-02443-6](https://doi.org/10.1007/s13042-024-02443-6).
49. DEVLIN, Jacob; CHANG, Ming-Wei; LEE, Kenton; TOUTANOVA, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: BURSTEIN, Jill; DORAN, Christy; SOLORIO, Thamar (ed.). *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long*

- and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, 2019, s. 4171–4186. Dostupné z doi: [10.18653/v1/N19-1423](https://doi.org/10.18653/v1/N19-1423).
50. WEISS, Karl; KHOSHGOFTAAR, Taghi M.; WANG, DingDing. A survey of transfer learning. *Journal of Big Data*. 2016, roč. 3, č. 1, s. 9. ISSN 2196-1115. Dostupné z doi: [10.1186/s40537-016-0043-6](https://doi.org/10.1186/s40537-016-0043-6).
 51. MARTIN, Louis; MULLER, Benjamin; ORTIZ SUÁREZ, Pedro Javier; DUPONT, Yoann; ROMARY, Laurent; CLERGERIE, Éric de la; SEDDAH, Djamel; SAGOT, Benoît. CamemBERT: a Tasty French Language Model. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020. Dostupné z doi: [10.18653/v1/2020.acl-main.645](https://doi.org/10.18653/v1/2020.acl-main.645).
 52. DE VRIES, Wietse; VAN CRANENBURGH, Andreas; BISAZZA, Arianna; CASELLI, Tommaso; VAN NOORD, Gertjan; NISSIM, Malvina. BERTje: A Dutch BERT Model. arXiv, 2019-12. WorkingPaper. arXiv. Dostupné z doi: [10.48550/arXiv.1912.09582](https://doi.org/10.48550/arXiv.1912.09582).
 53. POLIGNANO, Marco; BASILE, Pierpaolo; GEMMIS, Marco de; SEMERARO, Giovanni; BASILE, Valerio. ALBERTO: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In: *CEUR Workshop Proceedings*. 2019, zv. 2481. Dostupné z doi: <https://doi.org/10.4000/ijcol.472>.
 54. PIKULIAK, Matúš; GRIVALSKÝ, Štefan; KONÔPKA, Martin; BLŠTÁK, Miroslav; TAMAJKA, Martin; BACHRATÝ, Viktor; SIMKO, Marian; BALÁŽIK, Pavol; TRNKA, Michal; UHLÁRIK, Filip. SlovakBERT: Slovak Masked Language Model. In: GOLDBERG, Yoav; KOZAREVA, Zornitsa; ZHANG, Yue (ed.). *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, s. 7156–7168. Dostupné z doi: [10.18653/v1/2022.findings-emnlp.530](https://doi.org/10.18653/v1/2022.findings-emnlp.530).
 55. HOFFMANN, Jordan; BORGEAUD, Sebastian; MENSCH, Arthur; BUCHATSKAYA, Elena; CAI, Trevor; RUTHERFORD, Eliza; LAS CASAS, Diego de; HENDRICKS, Lisa Anne; WELBL, Johannes; CLARK, Aidan; HENNIGAN, Tom; NOLAND, Eric; MILLICAN, Katie; DRIESSCHE, George van den; DAMOC, Bogdan; GUY, Aurelia; OSINDERO, Simon; SIMONYAN, Karen; ELSSEN, Erich; RAE, Jack W.; VINYALS, Oriol; SIFRE, Laurent. Training Compute-Optimal Large Language Models. 2022, roč. 35, s. 39884–39908. Dostupné tiež z: <https://dl.acm.org/doi/10.5555/3600270.3602446>.

56. ACHARYA, Kamal; VELASQUEZ, Alvaro; SONG, Houbing Herbert. A Survey on Symbolic Knowledge Distillation of Large Language Models. *IEEE Transactions on Artificial Intelligence*. 2024, roč. 5, č. 12, s. 5928–5948. ISSN 2691-4581. Dostupné z DOI: [10.1109/TAI.2024.3428519](https://doi.org/10.1109/TAI.2024.3428519).
57. WETZSTEIN, Gordon; OZCAN, Aydogan; GIGAN, Sylvain; FAN, Shanhui; ENGLUND, Dirk; SOLJAČIĆ, Marin; DENZ, Cornelia; MILLER, David A. B.; PSALTIS, Demetri. Inference in artificial intelligence with deep optics and photonics. *Nature*. 2020, roč. 588, č. 7836, s. 39–47. ISSN 1476-4687. Dostupné z DOI: [10.1038/s41586-020-2973-6](https://doi.org/10.1038/s41586-020-2973-6).
58. JIANG, Albert Q.; SABLAYROLLES, Alexandre; MENSCH, Arthur; BAMFORD, Chris; CHAPLOT, Devendra Singh; CASAS, Diego de las; BRESSAND, Florian; LENGYEL, Gianna; LAMPLE, Guillaume; SAULNIER, Lucile; LAVAUD, Léo Renard; LACHAUX, Marie-Anne; STOCK, Pierre; SCAO, Teven Le; LAVRIL, Thibaut; WANG, Thomas; LACROIX, Timothée; SAYED, William El. *Mistral 7B*. 2023. Dostupné z arXiv: [2310.06825](https://arxiv.org/abs/2310.06825) [cs.CL].
59. AINSLIE, Joshua; LEE-THORP, James; JONG, Michiel de; ZEMLYANSKIY, Yury; LEBRON, Federico; SANGHAI, Sumit. GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints. In: BOUAMOR, Houda; PINO, Juan; BALI, Kalika (ed.). *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Singapore: Association for Computational Linguistics, 2023, s. 4895–4901. Dostupné z DOI: [10.18653/v1/2023.emnlp-main.298](https://doi.org/10.18653/v1/2023.emnlp-main.298).
60. BELTAGY, Iz; PETERS, Matthew E.; COHAN, Arman. *Longformer: The Long-Document Transformer*. 2020. Dostupné z arXiv: [2004.05150](https://arxiv.org/abs/2004.05150) [cs.CL].
61. DOBEŠ, Marek. Evaluation of Quality of Slovak Language Use in LLMs. *Acta Electrotechnica et Informatica*. 2025, roč. 25, č. 1, s. 28–33. Dostupné z DOI: <https://doi.org/10.2478/aei-2025-0004>.
62. PAPINENI, Kishore; ROUKOS, Salim; WARD, Todd; ZHU, Wei-Jing. Bleu: a Method for Automatic Evaluation of Machine Translation. In: ISABELLE, Pierre; CHARNIAK, Eugene; LIN, Dekang (ed.). *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, 2002, s. 311–318. Dostupné z DOI: [10.3115/1073083.1073135](https://doi.org/10.3115/1073083.1073135).
63. LIU, Yunyi; WANG, Zhanyu; LI, Yingshu; LIANG, Xinyu; LIU, Lingqiao; WANG, Lei; ZHOU, Luping. MRScore: Evaluating Medical Report with LLM-Based Reward System. In: LINGURARU, Marius George; DOU, Qi; FERA-

- GEN, Aasa; GIANNAROU, Stamatia; GLOCKER, Ben; LEKADIR, Karim; SCHNABEL, Julia A. (ed.). *Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*. Cham: Springer Nature Switzerland, 2024, s. 283–292. ISBN 978-3-031-72384-1. Dostupné z DOI: https://doi.org/10.1007/978-3-031-72384-1_27.
64. BANERJEE, Satanjeev; LAVIE, Alon. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In: GOLDSTEIN, Jade; LAVIE, Alon; LIN, Chin-Yew; VOSS, Clare (ed.). *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, 2005, s. 65–72. Dostupné tiež z: <https://aclanthology.org/W05-0909/>.
65. LIN, Chin-Yew. ROUGE: A Package for Automatic Evaluation of Summaries. In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, 2004, s. 74–81. Dostupné tiež z: <https://aclanthology.org/W04-1013/>.
66. VEDANTAM, Ramakrishna; ZITNICK, C. Lawrence; PARIKH, Devi. CIDER: Consensus-based image description evaluation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, s. 4566–4575. Dostupné z DOI: [10.1109/CVPR.2015.7299087](https://doi.org/10.1109/CVPR.2015.7299087).
67. ZHANG, Tianyi; KISHORE, Varsha; WU, Felix; WEINBERGER, Kilian Q.; ARTZI, Yoav. *Proceedings of The Eighth International Conference on Learning Representations (ICLR 2020)*. BERTScore: Evaluating Text Generation with BERT. 2020. Dostupné tiež z: <https://arxiv.org/abs/1904.09675>.
68. HANNA, Michael; BOJAR, Ondřej. A Fine-Grained Analysis of BERTScore. In: BARRAULT, Loic; BOJAR, Ondrej; BOUGARES, Fethi; CHATTERJEE, Rajen; COSTA-JUSSA, Marta R.; FEDERMANN, Christian; FISHEL, Mark; FRASER, Alexander; FREITAG, Markus; GRAHAM, Yvette; GRUNDKIEWICZ, Roman; GUZMAN, Paco; HADDOW, Barry; HUCK, Matthias; YEPES, Antonio Jimeno; KOEHN, Philipp; KOCMI, Tom; MARTINS, Andre; MORISHITA, Makoto; MONZ, Christof (ed.). *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, 2021, s. 507–517. Dostupné tiež z: <https://aclanthology.org/2021.wmt-1.59/>.

69. CHEN, Stanley F; BEEFERMAN, Douglas; ROSENFELD, Roni. Evaluation Metrics For Language Models. 2018. Dostupné z doi: [10.1184/R1/6605324.v1](https://doi.org/10.1184/R1/6605324.v1).
70. XU, Zhenyu; SHENG, Victor S. Detecting AI-Generated Code Assignments Using Perplexity of Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2024, roč. 38, č. 21, s. 23155–23162. Dostupné z doi: [10.1609/aaai.v38i21.30361](https://doi.org/10.1609/aaai.v38i21.30361).
71. DUDOR, Ladislav; ANDOROVÁ, Petra. Niekoľko úvah k uplatňovaniu zásady zákaz reformatio in peius v správnom konaní. In: *Princípy trestania a správne delikty*. Ed. KOŠIČIAROVÁ, Soňa. Trnavská univerzita v Trnave, Právnická fakulta, 2016. Dostupné tiež z: <https://www.truni.sk/sites/default/files/uk/f000153.pdf>.

Zoznam skratiek

BERT Bidirectional Encoder Representations from Transformers.

BiGRU Bidirectional Gated Recurrent Units.

BiLSTM Bidirectional Long Short-Term Memory.

BiRNN Bidirectional Recurrent Neural Network.

BLEU Bilingual Evaluation Understudy.

BoW Bag-Of-Words.

CIDEr Consensus-based Image Description Evaluation.

CNN Convolutional Neural Network.

CRISP-DM Cross Industry Standard Process for Data Mining.

ECLI European Case Law Identifier.

ELU Exponential Linear Unit.

FNN Feedforward Neural Network.

GPT Generative Pre-trained Transformers.

GQA Grouped-Query Attention.

GRU Gated Recurrent Units.

IAA Inter-Annotator Agreement.

IDF Inverse Document Frequency.

JSON JavaScript Object Notation.

LDA Latent Dirichlet Allocation.

LLaMA Large Language Model Meta AI.

LLM Large Language Model.

LReLU Leaky Rectified Linear Unit.

LSA Latent Semantic Analysis.

LSI Latent Semantic Indexing.

LSTM Long Short-Term Memory.

METEOR Metric for Evaluation of Translation with Explicit ORdering.

MHA Multi-Head Attention.

MIT Massachusetts Institute of Technology.

MLM Masked Language Model.

MMLU Multi-Task Language Understanding.

MQA Multi-Query Attention.

NLG Natural Language Generation.

PLSI Probabilistic Latent Semantic Indexing.

PReLU Parametric Rectified Linear Unit.

RAG Retrieval-Augumented Generation.

ReLU Rectified Linear Unit.

RLHF Reinforcement Learning from Human Feedback.

RNN Reccurent Neural Network.

ROUGE Recall-Oriented Understudy for Gisting Evaluation.

SVD Singular Value Decomposition.

SVM Support Vector Machine.

SWA Sliding-Window Attention.

T-PTLMs Transformer-based Pretrained Language Models.

TCN Temporal Convolutional Networks.

TF-IDF Term Frequency and Inverse Document Frequency.

UPJŠ Univerzita Pavla Jozefa Šafárika.

Slovník termínov

Inferencia proces, kedy sa niektoré vstupné údaje, napríklad obrázok, odošlú cez sieť raz, v doprednom priechode, aby sa vypočítal požadovaný výsledok.

Open-source softvér s otvoreným zdrojovým kódom.

Reformatio in peius zmeny v neprospech obvineného, resp. zásada zákazu zmeny k horšiemu.

Zoznam príloh

Príloha A CD médium – záverečná práca v elektronickej podobe,

Príloha B Systémová príručka dostupná na adrese

<https://github.com/kkuichi/bs749dy>