Časopis o novinkách zo sveta vysokovýkonného počítania na Slovensku

2024

hpc focus

EURO
SLOVAKIA

ISBN 978-80-89871-23-0
ISSN 2729-9090

Lucia
DEMOVIČOVÁ
Riaditeľka NSCC

Editoriál

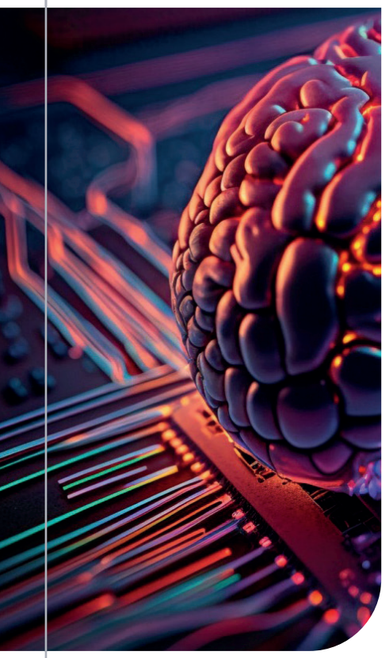
Vážení čitatelia a čitateľky, priaznivci HPC,

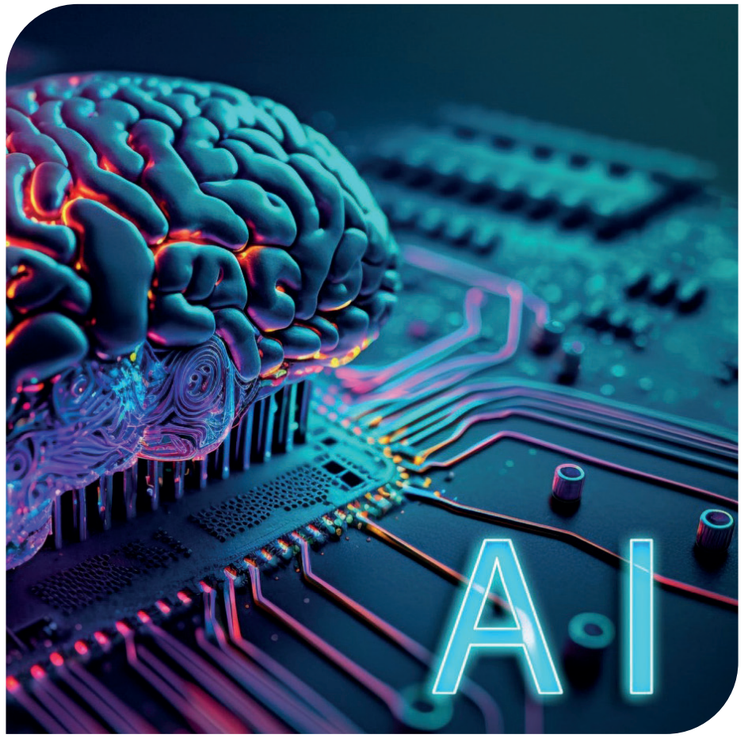
dostáva sa k vám nové vydanie magazínu HPC Focus, ktoré rekapituluje dianie v oblasti HPC na Slovensku, najdôležitejšie udalosti a úspechy za posledný rok.

Superpočítač Devana je už v plnej prevádzke a veľmi nás teší, že slúži čoraz širšiemu spektru vedných odborov a aj novým používateľom. V októbri 2023 sme spustili prvú výzvu na podávanie žiadostí o štandardný prístup a v čase vydania už beží v poradí štvrtá takáto výzva. Spolu sme vo výzvach rozdelili vyše 80 M CPU core-hodín a takmer 300 000 GPU-hodín strojového času pre 48 projektov. Záujem o výpočtovú kapacitu, predovšetkým akcelerované prostredie, stále rastie a už aj prevyšuje možnosti našej lokálnej infraštruktúry.

Kontinuálne sa snažíme aj vylepšovať naše služby, reflektovať súčasné trendy a uľahčovať prácu našim používateľom. Najnovšie máme pre vás pripravené vybrané nástroje na fine-tuning veľkých jazykových modelov na Devane. Keďže AI je v súčasnosti výraznou témou, aj príklady úspešných projektov, ktoré vám tento rok prezentujeme, sa týkajú práve tejto oblasti. Rozširujeme aj portfólio kurzov pre verejnosť, kam sme zaradili témy týkajúce sa AI a práce s dátami.

Súbežne s „bežnou“ prevádzkou tiež pracujeme na ďalšom veľkom projekte – vybudovaní superpočítača PERUN, ktorý by mal svojimi výkonnostnými parametrami rádovo prevyšovať možnosti Devany a mal by Slovensko dostať na HPC mapu Európy. V prípade úspešnej realizácie bude PERUN odovzdaný obstarávateľovi – CSČ SAV, v. v. i. – do konca roka 2025 a v nasledujúcich





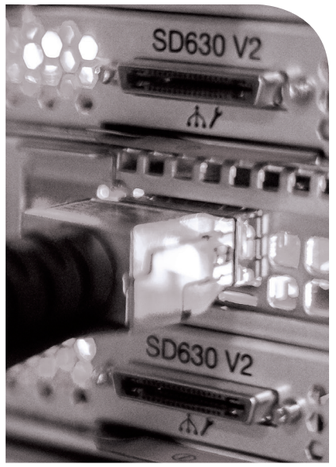
Kontinuálne sa snažíme vylepšovať naše služby, reflektovať súčasné trendy a uľahčovať prácu našim používateľom.

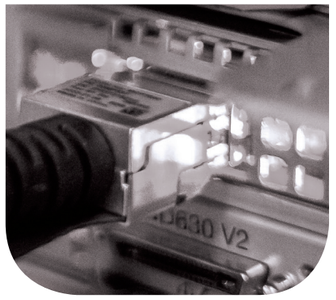
mesiacoch bude sprístupnený používateľom. Ambiciózny projekt je súčasťou slovenského plánu obnovy a odolnosti a chce dosiahnuť na prvú desiatku energeticky najefektívnejších HPC systémov v rebríčku Green500.

Je teda predpoklad, že v rozvoji HPC Slovensko dobehne susedov a európskych partnerov – no samotnou inštaláciou superpočítača to nedokáže. Okrem poskytovania výpočtového výkonu je nevyhnutné oslovovať, vzdelávať potenciálnych používateľov a primerane ich potrebám prispôsobovať služby a podporné nástroje. Základ ekosystému sme položili aj vďaka aktivitám Národného kompetenčného centra, projektom EuroCC a EuroCC 2 a veríme, že ho v spolupráci s partnermi a členmi NSCC budeme môcť aj naďalej rozvíjať.

SUPERPOČÍTAČ DEVANA JE UŽ V PLNEJ PREVÁDZKE.

OBSAH





Európska únia
Európsky fond regionálneho rozvoja
Operačný program Výskum a vývoj
Agentúra Ministerstva školstva, vedy, výskumu a športu SR pre štrukturálne fondy EÚ
Podporujeme výskumné aktivity na Slovensku
Projekt je spolufinancovaný zo zdrojov EÚ

01

NSCC
NCC PRE HPC
EUROHPC

hpc focus

Jazykové modely:
Moderné nástroje na riešenie komplexných úloh

BIBIÁNA LAJČINOVÁ
MICHAL PITOŇÁK
PATRIK VALÁBEK

Umelá inteligencia je aktuálne jednou z najrýchlejšie sa rozvíjajúcich oblastí informačných technológií. Obzvlášť disciplína spracovania prirodzeného jazyka (Natural Language Processing - NLP), ktorá je interdisciplinárnym odborom kombinujúcim počítačovú vedu, lingvistiku, matematiku, filozofiu a ďalšie oblasti. Medzi úlohy, ktorým sa táto oblasť venuje, patrí napríklad porozumenie prirodzenému jazyku, jeho generovanie, analýza sentimentu, vyhľadávanie a extrakcia informácií a podobne. O niektorých konkrétnych úlohách a ich riešeniach si môžete prečítať v článku Identifikácia entít pre extrakciu adries z transkribovaných rozhovorov s využitím syntetických dát [1] na webovej stránke Národného kompetenčného centra pre HPC. Dozviete sa o konkrétnej aplikácii metódy identifikácie entít (Named Entity Recognition – NER) na textové dáta obsahujúce adresy, pričom cieľom je extrahovať názov ulice, číslo domu, názov obce a poštové smerovacie číslo pomocou optimalizácie jazykového modelu SlovakBERT. V ďalšom článku Využitie veľkých jazykových modelov na efektívnu analýzu náboženských textov sa zaoberáme tvorbou vektorových databáz z náboženských textov pomocou verejne dostupných aj proprietárnych modelov, za účelom efektívneho vyhľadávania pasáží s relevantným obsahom (information retrieval).

Na vývoj obidvoch aplikácií boli použité výpočtové zdroje HPC systému Devana, z dôvodu vysokej výpočtovej náročnosti optimalizácie jazykových modelov. Tieto modely často obsahujú miliardy parametrov, a preto je prakticky nemožné ich optimalizovať na bežných osobných počítačoch. HPC systém Devana obsahuje 8 GPU výpočtových uzlov, pričom každý uzol má 4 GPU karty modelu nVidia A100 so 40 GB pamäte. Vďaka tomu je pri veľmi veľkých modeloch možné využiť distribuované trénovanie, či už implementovaním dátovej alebo modelovej paralelizácie. Distribuované trénovanie umožňuje efektívne rozloženie výpočtovej záťaže na viacero GPU a / alebo výpočtových uzlov a teda urýchlenie trénovania modelov.

Umelá inteligencia je aktuálne jednou z najrýchlejšie sa rozvíjajúcich oblastí informačných technológií, obzvlášť disciplína spracovania prirodzeného jazyka.

Jazykové modely

Základným stavebným kameňom spracovania prirodzeného jazyka sú jazykové modely. Tieto modely "rozumejú" textu, poprípade dokážu aj generovať nový text, vďaka ich architektúre a skutočnosti, že boli trénované na veľkom množstve textových dát. Ich architektúra je založená na hlbokých neurónových sieťach, konkrétne na špecifickej topológii nazývanej *transformers*. Táto technológia bola prvýkrát predstavená v roku 2017 v článku *Attention is all you need* [2] a spôsobila prevrat v oblasti NLP. Umožňuje spracovanie sekvenčných dát, vďaka čomu je vhodná práve na prácu s jazykom / textom (ale aj na iné druhy sekvenčných dát). Na rozdiel od tradičných neurónových sietí používaných na sekvenčné dáta, ako sú rekurentné siete, obsahujú modely založené na architektúre transformers aj mechanizmus pozornosti. Tento

mechanizmus umožňuje neurónovej sieti určiť, na ktoré časti vstupu sa má sústrediť a ako jednotlivé časti vstupu spolu súvisia, pre predikovanie najpravdepodobnejšieho výstupu.

Ďalším kľúčovým komponentom spracovania prirodzeného jazyka je tokenizácia. Táto operácia pretvára text na "tokeny", čo sú numerické reprezentácie slov, častí slov alebo jednotlivých znakov, v závislosti od použitého modelu. Tokeny zachytávajú kontext a význam slov a predstavujú vstup do neurónovej siete jazykových modelov.

Medzi ďalšie pojmy, ktoré sú frekventované v kontexte jazykových modelov, patria:

- **Embedding**
  vektorová reprezentácia textu (vety, frázy), ktorá zachytáva význam a kontext v numerickej podobe.
- **Fine-tuning**
  proces, pri ktorom sa predtrénovaný model optimalizuje na konkrétnu úlohu, ale už pomocou menšieho množstva dát a (väčšinou) zmeny menšieho počtu parametrov. Tento proces zlepšuje výkon modelu pri riešení špecifických úloh. Predtrénovaný model je už natrénovaný na enormnom množstve dát, k čomu je taktiež potrebný aj obrovský výpočtový výkon. Počas fine-tuning-u sa už iba doladia jeho parametre na špecifických dátach (v menšom objeme), ktoré počas predtrénovania neboli dostupné.

Medzi aktuálne "state-of-the-art" modely patrí napríklad BERT (Bidirectional Encoder Representations from Transformers), vyvinutý spoločnosťou Google. Ide o verejne dostupný (open-source) model obsahujúci len encoder časť, s pomerne jednoduchou architektúrou, ale vďaka svojej flexibilite je stále využívaný v rôznych úlohách. Ďalším populárnym modelom je GPT (Generative Pre-trained Transformer) od OpenAI. Je to multilingválny model, špecializovaný na generatívne úlohy a dokáže produkovať plynutý a kreatívny text. Avšak, tento model nie je verejne dostupný a je možné ho používať iba cez API (Application Programming Interface), ako aj ďalšie modely od internetových gigantov ako Facebook, Microsoft atď. Tieto komerčné modely síce majú špičkovú kvalitu, avšak nemusia byť vhodné pre každého, kvôli finančným nákladom, alebo na všetky úlohy, napríklad z dôvodu ochrany citlivých a / alebo proprietárnych dát.

Ako trénovať a používať jazykový model?

Pre užívateľov HPC systému Devana je na našich stránkach pripravený súbor návodov a ukážkových programov (v jazyku Python) na vytvorenie vlastných aplikácií využívajúcich veľké jazykové modely. Tieto návody sú zamerané na multilingválne a slovenské jazykové modely, ktoré užívatelia môžu optimalizovať (fine-tunovať) na základe vlastných dát ale aj používať ich voľne dostupné verzie bez fine-tuningu. Ku každému programu sú dostupné informácie o tom, ako ich spustiť na HPC systéme Devana pomocou dávkového systému Slurm, a ďalšie detaily. Medzi konkrétne aplikácie, ktoré sú v návodoch zahrnuté, patrí:

- **Analýza sentimentu pomocou modelu BERT na Twitter dátach**
  Viď. nasledujúca sekcia
- **Predikcia s použitím predtrénovaného LLM modelu na úlohu question-answering**
  Táto úloha sa zaoberá použitím verejne dostupných modelov Mistral 7B Instruct [3] a Aya [4] na úlohu question-answering. Na testovanie presnosti modelov používame verejne dostupný dataset MedQA [5], ktorý obsahuje otázky a odpovede z oblasti medicíny. Cieľom je ukázať, že veľké jazykové modely sa dajú použiť aj bez fine-tuningu.
- **Fine-tuning modelu Mistral 7B Instruct na úlohu question-answering**
  Táto úloha sa zaoberá fine-tuningom modelu Mistral 7B Instruct [3] na úlohu question-answering. Na trénovanie modelu používame verejne dostupný dataset MedQA [5], ktorý obsahuje otázky a odpovede z oblasti medicíny. Cieľom je vylepšiť schopnosť modelu správne odpovedať na otázky týkajúce sa medicínskeho obsahu, čím sa zvyšuje jeho presnosť a relevantnosť v danom kontexte.
- **Fine-tuning modelu DistilBERT na účely identifikácie entít (NER) s hľadaním optimálnych hodnôt hyperparametrov pomocou knižnice Optuna**
  V tejto úlohe optimalizujeme model DistilBERT [6] na úlohu identifikácie entít v texte. Používame verejne dostupný dataset CONLL2003 [7], ktorý obsahuje anotácie pre rôzne typy entít, kon-

krétne osoby (PER), miesta (LOC), organizácie (ORG) a rôzne iné (MISC). Optimálne hodnoty hyperparametrov trénovania hľadáme pomocou knižnice Optuna, ktorá nám umožňuje systematicky hľadať najlepšie hodnoty rýchlosti učenia, veľkosti dávky, počtu epoch atp.

- **Vytvorenie vektorovej databázy pomocou open-source embedding modelov**

  V tejto úlohe sa zameriavame na vytvorenie vektorovej databázy pomocou open-source embedding modelov. Používame model BGE M3 [8] na generovanie vektorových reprezentácií textov, ktoré nám umožňujú efektívne vyhľadávanie v texte. Texty, z ktorých generujeme embeddingy, pochádzajú z verejne dostupného datasetu 20newsgroups [9], ktorý obsahuje správy a články z novín. Tieto embeddingy slúžia na vytvorenie vektorovej databázy, ktorá uľahčuje rýchle vyhľadávanie a analýzu dokumentov na základe ich obsahovej podobnosti.

Prvú z uvedených aplikácií analyzujeme detailnejšie.

Analýza sentimentu pomocou modelu BERT na dátach zo sociálnej siete Twitter

V súčasnej dobe je analýza sentimentu jednou z častých úloh pre algoritmy spracovania prirodzeného jazyka. Táto úloha sa zameriava na identifikáciu a hodnotenie sentimentu (t.j. "nálady", väčšinou sa rozlišuje pozitívny, negatívny a neutrálny) v textoch, čo je užitočné napr. pre analýzu názorov verejnosti, zákazníkov, nálad a trendov v rôznych spoločenských a biznisových doménach. Modely ako BERT sa ukázali ako účinné nástroje práve na tento účel. Dáta používané na ich trénovanie sú častokrát získavané zo sociálnych sietí ako Twitter a Facebook, pretože zrkadlia širokú škálu názorov a emocionálnych prejavov používateľov. V tabuľke 1 sú uvedené príklady "tweetov" s rôznymi sentimentami:

Tweet	Sentiment
Skvelý deň! Slnečno a všetci sú usmiati.	pozitívny
Nenávidím, keď musím čakať v dlhých radoch.	negatívny
Dnes je utorok.	neutrálny

Keďže sa jedná o úlohu, kde každému pozorovaniu priraďujeme jeden sentiment, hovoríme o klasifikácii (do troch tried). Preto budeme používať model BERT s výstupnou vrstvou s troma neurónmi.

Ďalej si ukážeme, ako vytvoriť skript na fine-tunovanie modelu BERT na verejne dostupných dátach z Twitter-u.

Príprava dát, výber modelu a trénovanie

Pred samotným trénovaním modelu je kľúčové správne pripraviť dáta. V našom prípade používame dáta, ktoré sú vo formáte *csv*, tvorené tweetmi v anglickom jazyku, pričom každý z nich je označený jedným z troch možných sentimentov – pozitívny, neutrálny a negatívny.

V tomto prípade predspracovanie dát spočíva len v odstraňovaní riadkov s chýbajúcimi hodnotami.

```
df = pd.read_csv('Twitter_Data.csv')
df = df.dropna(subset=['category', 'clean_text'])  # Drop rows with NaN values
```

Pretože ide o dáta v anglickom jazyku, vybrali sme model *BERT base model* (*uncased*) [10], čiže model BERT v zmenšenej verzii (obsahuje menej parametrov oproti verzii *large*), ktorý nerozlišuje medzi veľkými a malými písmenami. K tomuto modelu patrí aj príslušný tokenizer.

Predtrénovaný model načítame z Hugging Face hub-u[1]. Keď sa používa metóda *from_pretrained*, načíta sa predtrénovaný model (existujúci model s natrénovanými váhami). To znamená, že nasleduje iba fine-tuning, nie trénovanie od základu.

```
MODEL = "bert-base-uncased"
tokenizer = BertTokenizer.from_pretrained(MODEL, do_lower_case=True)
# Load pretrained model with 3 labels: positive, negative, neutral
model = BertForSequenceClassification.from_pretrained(MODEL, num_labels=3)
```

Po rozdelení dát na trénovaciu, validačnú a testovaciu množinu všetky tri časti tokenizujeme.

[1] Hugging Face je platforma pre NLP a strojové učenie. Poskytuje široký výber pretrénovaných modelov a nástrojov, ktoré sú k dispozícii na ich webovej stránke https://huggingface.co.

```
# Split data into train, validation, test sets
train_data, test_data = train_test_split(df.to_
dict(orient='records'), test_size=0.2, random_state=42)
train_data, val_data = train_test_split(train_data, test_size=0.1,
random_state=42)

# Create DatasetDict for train, validation, test datasets
dataset_dict = DatasetDict({
    'train': Dataset.from_dict({'text': [item['clean_text'] for
item in train_data], 'label': [item['category_1'] for item in
train_data]}),
    'validation': Dataset.from_dict({'text': [item['clean_text']
for item in val_data], 'label': [item['category_1'] for item in
val_data]}),
    'test': Dataset.from_dict({'text': [item['clean_text'] for
item in test_data], 'label': [item['category_1'] for item in test_
data]})
})

tokenized_dataset = dataset_dict.map(tokenize_function,
batched=True)
```

Následne definujeme trénovacie argumenty, vrátane počtu epoch, veľkosti dávky (batch size), stratégie vyhodnocovania, atď., a môžeme pristúpiť k fine-tunovaniu modelu pomocou metódy *Trainer*.

```
training_args = TrainingArguments(
    per_device_train_batch_size=64,
    per_device_eval_batch_size=64,
    output_dir="bert_twitter",
    num_train_epochs=5,
    weight_decay=0.1,
    evaluation_strategy="epoch",
    eval_steps=300,
    save_strategy="epoch",
    save_steps=300,
    load_best_model_at_end=True,
    eval_accumulation_steps=300,
    logging_steps=300
)

trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=tokenized_dataset["train"],
    eval_dataset=tokenized_dataset["validation"],
    tokenizer=tokenizer,
    data_collator=data_collator, # responsible for processing data
into expected format - batches
    compute_metrics=compute_metrics # function calculating metrics
to evaluate model's accuracy

trainer.train()
```

Fine-tunovaný model je možné uložiť a používať ho na predikciu na nových dátach.

```
model.save_pretrained("bert_sentiment_model")
```

Predikciu na nových dátach môžeme urobiť pomocou metódy *pipeline* z knižnice *transformers*. Vo výstupe dostaneme okrem predikovanej triedy (sentimentu) aj pravdepodobnosť, ktorá vyjadruje mieru istoty modelu pri priraďovaní textu do danej triedy.

```
model = BertForSequenceClassification.from_pretrained("bert_sentiment_model")
sentiment_pipeline = pipeline("sentiment-analysis", model=model,
tokenizer=tokenizer, device=device) # inference initialization

text = "today is a beautiful day" # input text
result = sentiment_pipeline(text)
print(result)
```

Spustenie trénovania pomocou manažéra úloh Slurm a grafického rozhrania prostredníctvom OpenOnDemand

Trénovanie modelu je ideálne realizovať pomocou manažéra úloh Slurm, ktorý poskytuje prístup na výpočtové uzly HPC systému Devana (na rozdiel od priameho spustenia na jednom z prihlasovacích uzlov). Alternatívne je možné využiť grafické rozhranie prostredníctvom služby OpenOnDemand, vhodné napr. pre interaktívnu prácu s kódom v Jupyter Notebooku. Pri využití tejto služby je možné alokovať požadovaný výpočtový výkon na určitý čas a následne spustiť trénovanie modelu a / alebo analyzovať výsledky priamo v prostredí Jupyter Notebooku.

Nasledujúci shell skript slúži na spustenie fine-tunovania BERT modelu na analýzu sentimentu na HPC systéme pomocou plánovača úloh Slurm:

```
#!/bin/bash
#SBATCH --account=<your_project_number> # Kod projektu
#SBATCH -o result.txt                   # Subor pre vystup
#SBATCH -e error.txt                    # Subor pre zachytavanie chyb
#SBATCH --gres=gpu:1                    # Alokacia 1 GPU
#SBATCH --nodes=1                       # Poziadavka na 1 uzol
#SBATCH --partition=gpu                 # Vyber GPU particie

module load singularity                 # Nacitanie modulu singularity
```

```
singularity exec --nv /storage-data/singularity_containers/pt-
2.3_llm.sif python3 sentiment_analysis_bert_train.py --batch_
size 64
```

Posledný riadok spúšťa Python skript *sentiment_analysis_bert_train.py* pomocou singularity kontajnera. Parameter *--nv* umožňuje použitie GPU akcelerátorov v rámci kontajnera. Kontajner *pt_2.3_llm.sif* obsahuje všetky potrebné knižnice vrátane PyTorch, Transformers, Pandas, Numpy a ďalších.

Distribuované trénovanie

Pri trénovaní veľkých jazykových modelov s miliardami parametrov, je paralelizácia kľúčová pre efektívne využitie dostupných výpočtových zdrojov a zníženie celkovej doby výpočtu. Existujú dva hlavné prístupy k paralelizácii: dátová paralelizácia a modelová paralelizácia.

Dátová paralelizácia spočíva v rozdelení trénovacích dát na menšie časti, ktoré sa spracúvajú súčasne, na viacerých GPU akcelerátoroch. Modelová paralelizácia spočíva v rozdelení samotného modelu na menšie časti, ktoré sa spracúvajú súčasne na viacerých GPU akcelerátoroch alebo výpočtových uzloch. Tento prístup je užitočný, ba až nevyhnutný, keď je LLM model príliš veľký na to, aby sa zmestil do pamäte jedného GPU akcelerátora.

Modely z knižnice Hugging Face umožňujú automatickú modelovú paralelizáciu. Pri paralelizácii modelu je potrebné špecifikovať zariadenia, na ktoré chcete modely načítať, alebo je možné použiť hodnotu "auto" v argumente *device_map*, a tým sa model automaticky rozdelí medzi dostupné zdroje.

```
model = BertForSequenceClassification.from_pretrained("bert_
sentiment_model", device_map = "auto")
```

Ďalšie možnosti distribuovaného trénovania sú podrobne popísané v dokumentácii Hugging Face. Táto stránka poskytuje prehľad rôznych techník paralelizácie, ktoré môžu byť využité na efektívne rozdelenie trénovania na viacero GPU akcelerátorov.

Všetky programy a potrebné pomocné skripty k ďalším príkladom, vrátane inštrukcií na spustenie, sú dostupné na našom repozitári github.com/NSCC-Slovakia/NCC_HPC-FineTuning-Examples. Alternatívne, ak by ste potrebovali ďalšiu pomoc alebo konzultácie ohľadom trénovania modelov na HPC systéme Devana, srdečne Vás pozývame na osobné alebo online stretnutie v NSCC.

ZDROJE

[1] Bibiána Lajčinová, Patrik Valábek, and Michal Spišiak. Named entity recognition for address extraction in speech-to-text transcriptions using synthetic data, 2024.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[3] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[4] Ahmet ¨Ust¨un, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*, 2024.

[5] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

[6] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

[7] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003.

[8] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

[9] Ken Lang. Newsweeder: Learning to filter netnews. In *Armand Prieditis and Stuart Russell, editors, Machine Learning Proceedings 1995*, pages 331–339. Morgan Kaufmann, San Francisco(CA), 1995.

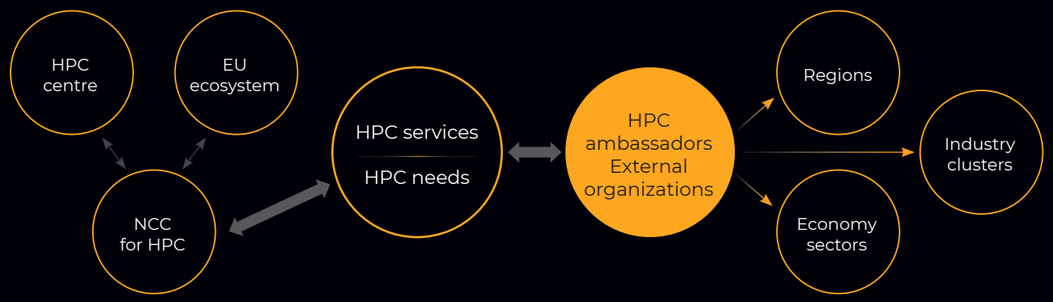
[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

Pri trénovaní veľkých jazykových modelov s miliardami parametrov, je paralelizácia kľúčová pre efektívne využitie dostupných výpočtových zdrojov a zníženie celkovej doby výpočtu.

HPC Ambassador Program

SPOLUPRACUJEME NAPRIEČ SLOVENSKOM

V Národnom kompetenčnom centre pre vysokovýkonné počítanie (ďalej len NCC) sa už od roku 2020 snažíme oslovovať nových záujemcov a potenciálnych používateľov HPC technológií na Slovensku. V súlade s cieľmi Európskej únie sa sústreďujeme predovšetkým na podporu adopcie HPC v súkromnom sektore, medzi malými a strednými podnikmi, ale aj väčšími priemyselnými spoločnosťami. Chceme, aby aj slovenské firmy mali možnosť nasadiť pokročilé digitálne technológie a využiť nielen výpočtový výkon, ale aj expertízu a príležitosti vzdelávať sa a zvyšovať si digitálne kompetencie. NCC slúži práve ako takýto „one-stop-shop“ pre oblasť vysokovýkonného počítania na Slovensku. Disponujeme vlastnými odbornými kapacitami, ktoré vieme zapojiť do pilotných alebo proof-of-concept projektov, organizujeme prednášky a kurzy pre širokú verejnosť, prinášame workshopy s domácimi aj zahraničnými expertmi.



Osloviť široké spektrum rôznorodých, väčších či menších podnikov s rôznymi očakávaniami, potrebami a stupňom pripravenosti však nie je jednoduché. Rozhodli sme sa preto ako súčasť rozvoja HPC ekosystému spustiť tzv. HPC Ambassador program. Ide o sieť kontaktov, združujúcich organizácií a významnejších hráčov v jednotlivých slovenských regiónoch, či v jednotlivých hospodárskych odvetviach.

Podstatou tohto programu je vzájomne prospešné partnerstvo medzi kompetenčným centrom a jednotlivými organizáciami – ambasádormi. Ambasádori sú zväčša združenia alebo klastre, ktoré majú širokú členskú základňu a sú etablovanou inštitúciou v konkrétnom regióne alebo v konkrétnej oblasti.

Vzájomnou komunikáciou a spoluprácou dokážeme adresnejšie identifikovať potreby koncových klientov/firiem, stupeň ich pripravenosti využiť HPC+ nástroje, ako je HPDA (vysokovýkonná dátová analytika), AI (umelá inteligencia), či pokročilé numerické simulácie. Vďaka tomu môžeme efektívnejšie prispôsobiť služby i cieliť komunikáciu, napríklad o grantových príležitostiach, možnostiach vzdelávania v rámci európskej siete EuroCC alebo o tom, ako získať strojový čas na Slovensku a v Európe.

Priebeh spolupráce

VSTUPNÁ KONZULTÁCIA IDENTIFIKÁCIA POTRIEB

Analýza štruktúry a potrieb vašich členov, ako aj procesov vo vašom združení. Toto je kľúčový krok na prispôsobenie našej spolupráce vašim jedinečným potrebám.

IDENTIFIKÁCIA POTENCIÁLNYCH ZÁUJEMCOV Z RADOV MSP

Pomocou jednoduchej prieskumnej matice a digitálneho auditu identifikujeme potenciálnych záujemcov spomedzi vašich členov, pre ktorých by využitie našich HPC služieb mohlo priniesť reálne benefity.

WORKSHOP ŠITÝ NA MIERU

Pre konkrétnych záujemcov pripravíme workshop, ktorý bude šitý na mieru ich potrebám a cieľom, pomôžeme im lepšie porozumieť možnostiam implementácie konkrétnych HPC technológií.

BEZPLATNÝ OPEN SCIENCE, PoC PROJEKT RESP. KOMERČNÉ RIEŠENIE

Podporujeme otvorený výskum a vývoj poskytovaním našej odbornej podpory a výpočtových zdrojov pre pilotné a proof-of-concept projekty. Ak projekt dosiahne úspech, je možné pokračovať realizáciou komerčného riešenia s pozitívnym dopadom na firmu a jej konkurencieschopnosť na trhu.

SYNERGIE | ZDIEĽANIE KNOW HOW ČLENSTVO V PORADNOM VÝBORE

Neustále zdieľanie postupov dobrej praxe medzi zúčastnenými subjektmi, ktoré zároveň vytvára vzájomný marketingový potenciál. Navyše máte možnosť zapojiť sa do Poradného výboru pre spoluprácu so súkromným sektorom pri Národnom superpočítačovom centre a tak ovplyvňovať stratégie a smer rozvoja HPC technológií na Slovensku.

Sme veľmi hrdí, že už krátko po spustení programu máme možnosť spolupracovať s našimi ambasádormi:

INOVIA

Slovenská obchodná a priemyselná komora
Bratislavská regionálna komora

ÚKS Únia Klastrov Slovenska

IT VALLEY KOŠICE

Ďalej spolupracujeme aj s Regionálnymi centrami Ministerstva investícií, regionálneho rozvoja a informatizácie Slovenskej republiky.

MINISTERSTVO
INVESTÍCIÍ, REGIONÁLNEHO ROZVOJA
A INFORMATIZÁCIE
SLOVENSKEJ REPUBLIKY

Spúšťame mobilitný program HPC Fellowship

Čo je to mobilitný program HPC Fellowship?

Mobilitný program HPC Fellowship, ktorý koordinuje Národné superpočítačové centrum, je iniciatíva z Akčného plánu digitálnej transformácie Slovenska na roky 2023 – 2026. **Je navrhnutý pre študentov druhého a tretieho stupňa vysokoškolského štúdia, ktorí majú záujem o získanie skúseností v oblasti administrácie HPC systémov alebo v oblasti vývoja a nasadenia vysoko paralelných HPC aplikácií v tzv. HPC+ (AI, HPDA, simulácie).** Program je otvorený pre študentov z rôznych študijných odborov, napr. informatiky, fyziky, inžinierstva, biológie, ekonómie, ale aj spoločenských vied, jazykovedy a iných. NSCC v rámci programu v spolupráci so zapojenými hostiteľskými organizáciami zverejní aktuálne rámcové okruhy tém pre mobilitné projekty.

Kľúčovým kritériom výberu a vyhodnotenia prihlášok je schopnosť uchádzača preukázať, ako plánuje získané skúsenosti a vedomosti v HPC/HPC+ využiť v ďalšom štúdiu alebo pri výskume, ktorým sa zaoberá. Program je financovaný prostriedkami z Ministerstva investícií, regionálneho rozvoja a informatizácie Slovenskej republiky a jeho

Kľúčovým kritériom výberu a vyhodnotenia prihlášok je schopnosť uchádzača preukázať, ako plánuje využiť získané skúsenosti a vedomosti.

spustenie sa plánuje na september 2024. **Mobilitný program nepokrýva účasť na semestrálnych predmetoch v zahraničnej inštitúcii.**

ZÁKLADNÉ INFORMÁCIE O PROGRAME

- Trvanie: 1 až 3 mesiace.
- Hostiteľské inštitúcie: Európske superpočítačové centrá. Pre prvý rok programu komunikujeme s partnerskými centrami v Českej republike (IT4Innovations – VŠB Technická univerzita Ostrava), Rakúsku (VSC – Vienna Scientific Cluster), Poľsku (Academic Computer Centre CYFRONET of the AGH University of Krakow) a Taliansku (CINECA, Bologna).
- Štipendium: Do 4 000 € mesačne na pokrytie cestovných nákladov, pobytových nákladov, stravného, nákladov na poistenie a pod.

ČO ZÍSKATE S HPC FELLOWSHIP

1. Prístup k najmodernejším zariadeniam
   Študenti budú mať možnosť pracovať vo vybranom európskom superpočítačovom centre, kde získajú praktické skúsenosti so špičkovými technológiami a aplikáciami pre administráciu HPC alebo aplikáciami v HPC+ oblastiach.

2. Mentorstvo a vytváranie sietí
   Študenti budú spolupracovať s poprednými odborníkmi a výskumníkmi, získajú prehľad o najnovších trendoch a výzvach vo zvolenej oblasti. Táto skúsenosť je neoceniteľná pre budovanie profesionálnych sietí, kontaktov, budovanie vzťahov a skúmanie budúcich kariérnych príležitostí.

Program je otvorený pre študentov z rôznych študijných odborov, napr. informatiky, fyziky, inžinierstva, biológie, ekonómie, ale aj spoločenských vied, jazykovedy a iných.

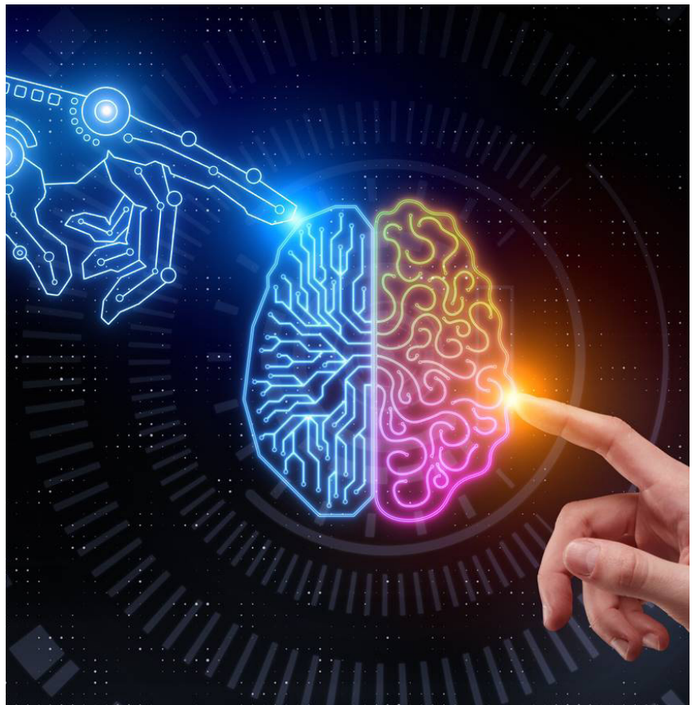
3. Akademický rast

   Vďaka štipendiu a odbornému vedeniu si študenti rozširia svoje obzory vo zvolenom odbore nad rámec bežného štúdia, vyskúšajú si samostatnú prácu na konkrétnych úlohách a zvýšia si kompetencie.

Proces podávania prihlášok

Mobilitný program HPC Fellowship je určený pre študentov druhého a tretieho stupňa štúdia slovenských vysokých škôl, ktorí preukážu záujem o HPC/HPC+. Prihláška je dostupná online na webovej stránke Národného superpočítačového centra (nscc.sk) a je otvorená počas celého akademického roka. Uchádzači musia predložiť životopis, výpis výsledkov štúdia a motivačný list, v ktorom sa uvádza, ako plánujú používať HPC napr. v záverečných prácach alebo výskume, ktorému sa venujú.

Prihlášky posudzuje výberová komisia v zložení zástupcov z vysokej školy uchádzača a zástupcov z Národného superpočítačového centra. Priorita sa dáva projektom, ktoré preukazujú inováciu, interdisciplinárnu spoluprácu a potenciál významného prínosu k študijnému odboru uchádzača.



02

APLIKÁCIE V PRAXI

IMPLEMENTÁCIA METÓDY ČIASTOČNE RIADENÉHO UČENIA UNI-MATCH DO METÓDY FRAME FIELD LEARNING PRE ÚLOHU EXTRAKCIE BUDOV Z LETECKÝCH SNÍMOK

Patrik Sabol
Bibiána Lajčinová

Extrakcia budov v Geografických informačných systémoch (GIS) je kľúčová pre urbanistické plánovanie, environmentálne štúdie a riadenie infraštruktúry, pretože umožňuje presné mapovanie stavieb, vrátane odhaľovania nelegálnych stavieb za účelom dodržiavania právnych predpisov, alebo efektívnejšieho vyberania daní. Integrácia extrahovaných údajov o budovách s inými geopriestorovými vrstvami zlepšuje pochopenie dynamiky miest a priestorových vzťahov. Vzhľadom na rozsah a zložitosť týchto úloh rastie potreba automatizovať extrakciu budov pomocou techník hlbokého učenia, ktoré ponúkajú vyššiu presnosť a efektívnosť pri spracovaní veľkých geopriestorových dát.

V súčasnosti, väčšina najmodernejších segmentačných modelov hlbokého učenia poskytuje výstup iba v rastrovej forme. Avšak GIS často potrebujú dáta vo vektorovej forme. Jednou z metód, ktorá dokáže generovať dáta vo vektorovej forme, je Frame Field learning. Táto metóda generuje okrem segmentačnej masky aj frame field pole, ktoré obsahuje štrukturálne informácie o objektoch, ktoré sa následne využívajú v procese vektorizácie.

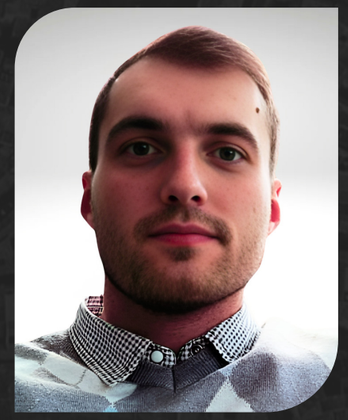
Modely Frame Field learningu sú trénované metódou "s učiteľom" (z angl. *supervised learning*), ktorá potrebuje veľké množstvo anotovaných dát. Na získanie takéhoto množstva kvalitných dát je potrebná manuálna ľudská práca, ktorá však môže byť zdĺhavá a nákladná. Jednou z metód, ktorá môže znížiť závislosť od anotovaných dát, je "učenie s čiastočným učiteľom", resp. "čiastočne riadené učenie" (z angl. *semi-supervised learning*). Tento prístup učenia využíva nielen anotované dáta, ale aj množinu neanotovaných dát.

Cieľom tejto spolupráce medzi Národným kompetenčným centrom pre HPC a Geodeticca Vision, s. r. o. bolo identifikovať, implementovať a vyhodnotiť vhodnú metódu učenia s čiastočným učiteľom pre Frame Field learning.

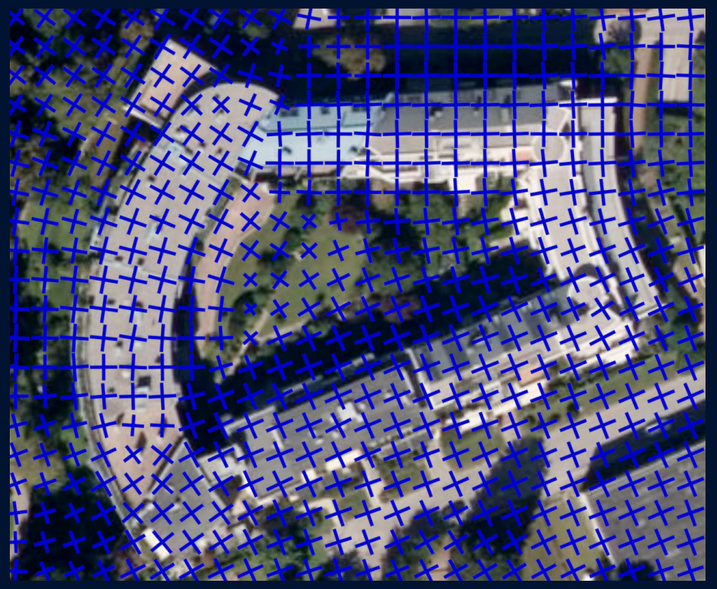
Metódy

FRAME FIELD LEARNING

Hlavnou myšlienkou Frame Field learning [1] metódy je pomôcť vektorizačnému algoritmu vyriešiť nejednoznačné prípady pri vektorizácii, ktoré sú spôsobené diskrétnou pravdepodobnostnou segmentačnou mapou (výstup zo segmentačného modelu), a to pridaním tzv. frame fields poľa (viď. Obrázok 1) ako ďalšieho výstupu z neurónovej siete, reprezentujúceho geometrické charakteristiky budov.



Ing. Patrik Sabol, PhD. pôsobí ako expert v oblasti počítačového videnia v spoločnosti Geodeticca Vision, s. r. o.



OBRÁZOK 1

Ukážka frame field poľa definovaného pre budovu z trénovacej sady [1].

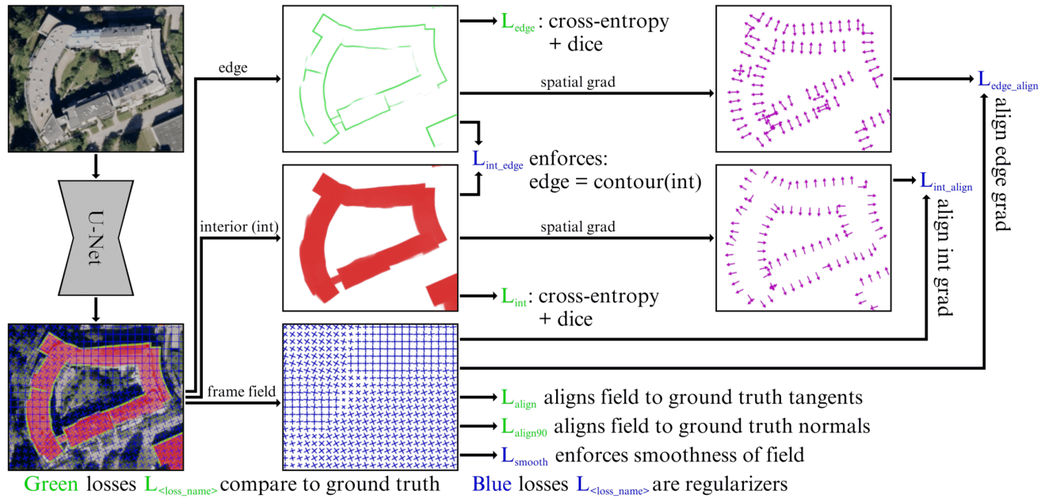
FRAME FIELD POLE

Frame field je vektorové pole rádu 4, čo znamená, že každému bodu v rovine priradí 4 smerové vektory. Protiľahlé vektory majú rovnakú hodnotu, ale s opačným znamienkom, takže každému bodu v rovine je priradený vektor {u, −u, v, −v}. Tieto vektory postačujú na definovanie tvaru budov, ktoré sú z veľkej časti pravidelného tvaru s pravouhlými rohmi.

FRAME FIELD LEARNING

Proces metódy Frame Field learning môžeme zosumarizovať nasledovne:

1. Vstupom do neurónovej siete je RGB obraz o veľkosti 3 x V x Š.
2. Na generovanie mapy príznakov (z angl. *feature map*) je možné využiť rôzne segmentačné architektúry, napr. U-Net.
3. Učenie je supervizované (patrí medzi metódy učenia s učiteľom), pričom pre učenie segmentačných masiek sa využívajú označené rastrované polygóny pre interiér a hranice budov. Ako stratová funkcia sa využíva lineárna kombinácia funkcií cross-entropy a Dice loss.



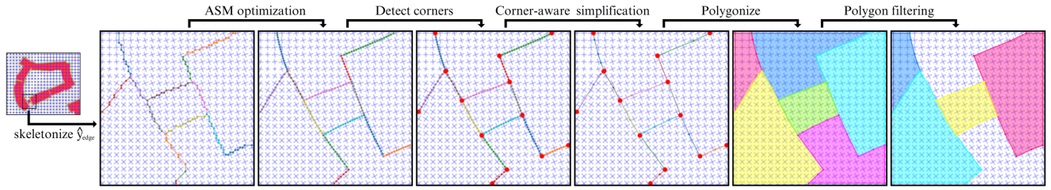
4. Pre učenie samotného Frame Field poľa sa využívajú vektory polygónov označených budov, kde konzistentnosť a presnosť Frame Field poľa zabezpečujú tri stratové funkcie:
   - L_{align} – stratová funkcia riadi správne natočenie Frame Field poľa na smery dotyčnice vektoru polygónu.
   - $L_{align90}$ – stratová funkcia zabraňuje, aby sa Frame Field pole degradovalo na priamkové pole.
   - L_{smooth} – zabezpečuje hladký priebeh Frame Field poľa.
5. Pre zachovanie konzistentnosti medzi segmentačnou pravdepodobnostnou mapu a Frame Field výstupom sú definované regularizačné stratové funkcie, ktoré zarovnávajú Frame Field pole s gradientmi segmentačnej mapy.

OBRÁZOK 2

Diagram procesu Frame Field learning [1].

VEKTORIZÁCIA

Proces vektorizácie transformuje výstup z natrénovanej neurónovej siete do topologicky čistých vektorov pomocou algoritmu Active Skeleton Model (ASM). Princíp algoritmu spočíva v iteratívnom posúvaní vrcholov skeletového grafu do ich ideálnej pozície. Skeletový graf je vygenerovaný pomocou



OBRÁZOK 3

Vizualizácia procesu vektorizácie [1].

morfologickej operácie "*thinning*" z gradientu segmentačnej mapy. Iteratívny posun je riadený gradientovou optimalizačnou metódou, ktorej cieľom je minimalizovať energetickú funkciu, ktorá má nasledujúce zložky:

- $E_{probability}$ – riadi prispôsobenie skeletového grafu kontúram pravdepodobnostnej mapy budovy na konkrétnu hodnotu pravdepodobnosti (napr. 0.5)
- $E_{frame\ field\ align}$ - riadi zarovnanie každej hrany skeletového grafu na Frame Field pole.
- E_{length} – zaisťuje homogénnu distribúciu vrcholov skeletového grafu.

UNIMATCH METÓDA ČIASTOČNE RIADENÉHO UČENIA

UniMatch [2], pokročilá metóda učenia s čiastočným učiteľom z kategórie regulátorov konzistentnosti, stavia na základných princípoch vytvorených metódou FixMatch [3], ktorá je základnou metódou v tejto kategórii algoritmov. Funguje na princípe pseudo-označovania (z angl. *pseudo-labeling*) v kombinácií s reguláciou konzistentnosti.

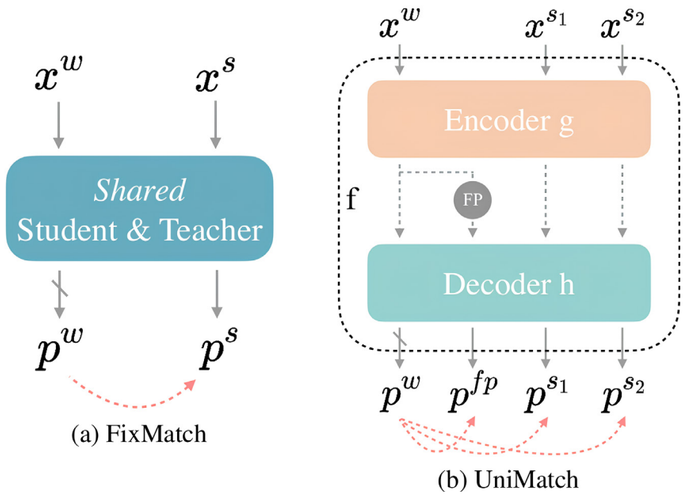
Základný princíp metódy FixMatch spočíva v generovaní pseudo-označení (anotácií) pre neanotované dáta, pomocou predikcií neurónovej siete. To znamená, že pre slabo perturbovaný neanotovaný vstup x^w sa vygeneruje predikcia p^w, ktorá slúži ako pseudo-označenie pre predikciu silne perturbovaného vstupu x^s. Následne sa vypočíta hodnota chybovej funkcie, napr. cross-entropy(p^w, p^s), pričom do úvahy sa berú iba tie oblasti z p^w, ktoré majú hodnotu pravdepodobnosti väčšiu ako daný prah, napr. >0.95.

Rozšírenie metódy UniMatch oproti metóde FixMatch spočíva v dvoch princípoch:

1. UniPerb (Unified Perturbations for Images and Features) – aplikácia perturbácie na úrovni príznakov (z angl. *feature perturbation*). V praxi to znamená, že na výstup (teda príznak – *feature*) z encoder vrstvy neurónovej siete sa aplikuje dropout funkcia, ktorá náhodne vynuluje niektoré príznaky. Takto upravený výstup z encoder vrstvy následne vstupuje do decoder časti siete, ktorá vygeneruje p^{fp}.
2. DusPerb (Dual-Stream Perturbations) – namiesto jednej silnej perturbácie sa využívajú dve silné perturbácie x^{s1} a x^{s2}.

V konečnom dôsledku máme tri stratové funkcie - cross-entropy(p^w, p^{fp}), cross-entropy(p^w, p^{s1}), cross-entropy(p^w, p^{s2}). Tie sa nakoniec lineárne kombinujú so supervizovanou stratovou funkciou.

Táto metóda v súčasnosti patrí medzi state-of-the-art metódy učenia s čiastočným učiteľom. Hlavnou výhodou tejto metódy je jej jednoduchosť pri implementácií a nevýhodou je jej citlivosť na výber vhodnej slabej a silnej perturbácie.



OBRÁZOK 4

(a) FixMatch základná metóda, (b) použitá UniMatch metóda. *FP* je označenie pre feature perturbation (perturbácia príznakov), *w* ako weak (slabá) a *s* ako strong (silná) perturbácia [2].

Integrácia UniMatch metódy do Frame Field učenia

IMPLEMENTÁCIA UNIMATCH DO FRAME FIELD LEARNING FRAMEWORKU

Aby sme mohli implementovať UniMatch metódu do Frame Field leagning štruktúry, potrebovali sme najprv definovať slabú a silnú perturbáciu v kontexte leteckých snímok. Ako slabé perturbácie sme zvolili základné priestorové transformácie obrazu, vrátane rotácie, zrkadlenia a vertikálneho/horizontálneho prevrátenia. Všetky tieto transformácie sú oprávnené pre letecké snímky.

V prípade silných perturbácií sme použili fotometrické transformácie. Tie zahŕňajú úpravy odtieňa, farby, či jasu obrazu. Poskytujú výraznejšie zmeny snímok než s použitím priestorových transformácií.

Dôležitým krokom bola implementácia perturbácie na úrovni príznakov (feature perturbation). Túto perturbáciu sme implementovali ako dropout mechanizmus vo vrstve medzi encoder a decoder časťami architektúry U-Net. Tento mechanizmus zahodí (nastaví na nulu) náhodne vybrané hodnoty príznakov (výstup z encoder vrstvy). Takto upravené hodnoty výstupu z encoder časti siete vstupujú ďalej do decoder časti U-Net architektúry.

V prípade dual-stream perturbácií sme prispôsobili Frame Field framework tak, aby využíval dve silné perturbácie. Predikcia pre slabú perturbáciu sa použila ako pseudo-označenie pre dve silné perturbácie (preto označenie *dual-stream*). Dve silné perturbácie prispievajú k celkovej robustnosti a efektívnosti modelu.

Prostredníctvom týchto úprav bola UniMatch metóda úspešne integrovaná do Frame Field learning algoritmu, čím sa zvýšila jeho schopnosť efektívne spracúvať a učiť sa z anotovaných a hlavne neanotovaných dát.

Experimenty

DÁTA

- *Anotované dáta*

Anotované dáta použité v štúdii pochádzajú z troch rôznych zdrojov, detaily sú uvedené v Tabuľke 1.

- *Neanotované dáta*

Neanotované dáta (verejne dostupné vysoko kvalitné letecké snímky) pochádzajú z Geodetického a kartografického ústavu (GKÚ) [6]. Pri výbere sme sa zamerali na oblasť s rozlohou 7 000 km², čím bola zaistená diverzita rôznych povrchov krajín a mestských prostredí.

Názov	Typ dát. sady	Pokrytá rozloha [km²]	Počet budov	Ortofotografické rozlíšenie [m]	Rastrová dlaždica [px]
Geodeticca-Buildings	Súkromná	198.18	50 354	0.25	4,1
INRIA [4]	Verejná	810.00	206 679	0.30	38,8
Landcover.ai [5]	Verejná	216.27	12 354	0.25/0.50	9 000x9 500 / 4 200x4 700

TABUĽKA 1

Prehľad troch zdrojov anotovaných dát použitých na trénovanie modelov.

- *Spracovanie dát: Patching*

Anotované aj neanotované snímky boli spracované pomocou metódy "patching", ktorá obraz rozdeľuje na malé časti veľkosti 320 x 320 px. Táto veľkosť bola špecificky vybraná tak, aby vyhovovala požiadavkám pre vstup zvolenej neurónovej siete. Takýmto spôsobom vzniklo z anotovaných dát približne 55 000 malých častí a z neanotovaných dát okolo 244 000 častí.

TRÉNOVANIE

- *Architektúra modelu*

Použitý model sme navrhli s pomocou U-Net architektúry s EfficientNet-B4 základom. Táto kombinácia poskytuje dobrú rovnováhu presnosti a efektívnosti, čo je veľmi dôležité pri práci s komplexnými segmentačnými úlohami. EfficientNet-B4 ako základ neurónovej siete bol vybraný pre optimálnu rovnováhu medzi spotrebou pamäte a výkonom. V metóde Frame Field learning sa U-Net architektúra ukázala byť vysoko efektívna, o čom svedčia výsledky použitia tejto siete v rôznych štúdiách.

- *Trénovací proces*

Na trénovanie sme použili AdamW optimalizátor, ktorý kombinuje výhody Adam optimalizácie s regularizačnou metódou *weight decay*, čím pomáha modelu lepšie generalizovať. Aby sme sa vyhli pretrénovaniu modelu, použili sme L2 regularizáciu a taktiež bola použitá metóda ReduceLROnPlateau na optimalizáciu parametra rýchlosti učenia. Táto metóda upravuje parameter rýchlosti učenia na základe validačnej straty.

- *Úpravy potrebné pre implementáciu učenia s čiastočným učiteľom*

Kľúčovým aspektom nášho trénovania bolo nastavenia podielu anotovaných a neanotovaných obrázkov. Experimentovali sme s pomermi od 1:1 do 1:5 (počet anotovaných : počet neanoto-

Integrácia extrahovaných údajov o budovách s inými geopriestorovými vrstvami zlepšuje pochopenie dynamiky miest a priestorových vzťahov.

vaných). Takýmto spôsobom sme zisťovali, ako rôzne množstvá neanotovaných dát ovplyvňujú trénovací proces. Identifikovali sme optimálny pomer pre trénovanie nášho modelu tak, aby bolo zachované efektívne učenie s využitím metódy učenia s čiastočným učiteľom.

VYHODNOTENIE MODELU

Na vyhodnotenie nášho modelu na extrakciu budov sme zvolili metriky, ktoré precízne merajú ako presne sa predikcie zhodujú so skutočnými štruktúrami.

- *Intersection over Union (IoU)*

Kľúčovou metrikou, ktorú sme využívali je metrika s názvom Intersection over Union (IoU). Počíta zhodu medzi predikciami modelu a skutočným tvarom budov. Hodnota skóre IoU blízka 1 znamená, že naše predikcie sú podobné skutočným budovám. Táto metrika je nevyhnutná na posúdenie geometrickej presnosti pre segmentované oblasti, pretože odráža presnosť vytýčenia hraníc budov. Okrem toho, vyhodnotením pomeru správne predikovanej oblasti ku kombinovanej oblasti (zjednotenie oblasti predikcie a skutočnej oblasti), nám IoU poskytuje jasnú mieru efektivity modelu v zachytávaní skutočného kontextu a tvaru budov v komplexnej mestskej krajine.

- *Precision, Recall (senzitivita) a F1 skóre*

Metrika nazývaná precision vyjadruje podiel správne identifikovaných budov zo všetkých identifikovaných budov. Senzitivita (angl. *recall*) ilustruje schopnosť modelu zachytiť všetky skutočné budovy. Vysoká hodnota tejto metriky poukazuje na citlivosť modelu pri detekcii budov. F1 skóre kombinuje precision a senzitivitu do jednej metriky, poskytujúc vyvážený obraz výkonu modelu.

- *Complexity Aware IoU (cIoU)*

Ďalšou použitou metrikou bola Complexity Aware IoU (cIoU) [7]. Táto metrika rieši nedostatky IoU tým, že vyvažuje presnosť segmentácie a komplexnosť tvarov polygónov. Zatiaľ čo IoU môže viesť model k vytváraniu veľmi komplexných polygónov, cIoU zaručuje, že komplexnosť polygónov (počet ich vrcholov) je zachovaná realistická, čím odráža skutočný tvar budov, ktoré sú obvykle málo komplexné.

- *N Ratio Metrika*

Metrika N ratio je doplnkovým komponentom v našej vyhodnocovacej stratégii. Porovnáva počet vrcholov v našich

predpovedaných tvaroch s tými v skutočných budovách [7]. Tým nám metrika pomáha porozumieť, ako presne náš model replikuje detailnú štruktúru budov.

- *Max Tangent Angle Error (MTAE)*

Na zaistenie čistej geometrie pri extrakcii budov, je dôležité presné meranie pravidelnosti kontúr. Chyba maximálneho uhla dotyčníc (resp. Max Tangent Angle Error (MTAE)) [1] je metrika navrhnutá presne pre tieto potreby, a je doplnením Intersection over Union (IoU) metriky. Špecificky cieli na nedostatok IoU metriky, ktorým je to, že segmentácia s okrúhlymi rohmi môže dosiahnuť vyššie skóre než segmentácia s presnejšími (ostrejšími) rohmi. Vyhodnocovaním zhody okrajov budov cez porovnávanie uhlov dotyčníc vo vybraných bodoch predikovaných a skutočných kontúr, MTAE efektívne penalizuje nepresnosti v orientácii okrajov. Toto zameranie na presnosť okrajov je dôležité pre produkovanie čistých vektorových reprezentácií budov, zdôrazňujúc dôležitosť presného vymedzenia hraníc v segmentačných úlohách.

- *Vyhodnotenie*

Natrénované modely boli testované na veľkej dátovej množne leteckých snímok v plnej veľkosti (namiesto malých častí, pomocou ktorých bola sieť trénovaná). Takéto testovanie poskytuje presnejšie zobrazenie reálnych použití takýchto modelov. Na extrakciu budov zo snímok v plnej veľkosti sme použili techniku posuvného okna, čím boli vytvorené predikcie po jednotlivých segmentoch obrázku. Na okraje prekrývajúcich sa segmentov bola použitá pokročilá priemerovacia

technika, dôležitá pre minimalizáciu nežiadúcich efektov a zachovanie konzistentnosti v rámci predikčnej mapy. Výstupná predikčná mapa v plnej veľkosti bola následne vektorizovaná do presných vektorových polygónov s použitím algoritmu Active Skeleton Model (ASM).

VÝSLEDKY

Výsledky z experimentov, odrážajúce výkon segmentačného modelu natrénovaného s rôznymi nastaveniami, odhalili zaujímavé zistenia (viď. Tabuľka 2). Vyhodnotili sme výkon základného modelu (len supervizovaný prístup) a výkon modelov trénovaných metódami učenia s čiastočným učiteľom s použitím rôznych podielov anotovaných a neanotovaných dát (1:1, 1:3, a 1:5).

Metóda trénovania	Podiel (anotované: neanotované)	IoU [%]	Precision [%]	Senzitivita [%]	F1 Skóre [%]	N Ratio	cIoU [%]	Priem. MTAE [°]
Učenie s učiteľom	-	80.50	85.75	94.27	89.81	2.33	48.89	18.60
Učenie s čiast. učiteľom	1:1	83.88	87.66	93.41	90.44	1.94	56.98	20.47
Učenie s čiast. učiteľom	1:3	85.35	90.04	94.25	92.10	1.76	61.91	18.92
Učenie s čiast. učiteľom	1:5	85.77	90.04	94.76	92.34	1.65	64.75	17.45

TABUĽKA 2

Výsledky trénovania modelov pre základný prístup (učenie s učiteľom) a prístupy učenia s čiastočným učiteľom s rôznymi podielmi použitých anotovaných a neanotovaných obrázkov.

1. IoU:
Hodnota IoU metriky bola pre základný model na hodnote 80.50 %. S prínosom neanotovaných dát do trénovacieho procesu pozorujeme stabilný nárast, dosahujúc až 85.77 %, s použitím pomeru 1:5 anotovaných k neanotovaným obrázkom.

2. Precision, senzitivita a F1 skóre:
Hodnota metriky precision sa zlepšila z hodnoty 85.75 % pre základný model na hodnotu 90.04% pre model s použitým podielom 1:5. Podobne senzitivita sa zľahka zvýšila z hodnoty 94.27 % na 94.76 %. F1 skóre taktiež narástlo z hodnoty 89.81 % na 92.34 %. Tieto zlepšenia naznačujú, že zakomponovaním metódy s čiastočným učiteľom sa model stal presnejším a spoľahlivejším v predikciách.

3. N Ratio a cIoU:
Výsledky ukazujú znateľný pokles v hodnote metriky N Ratio z hodnoty 2.33 pre základný model, na hodnotu 1.65 pre model s 1:5 podielom (anotované : neanotované), čo indikuje, že učenie s čiastočným učiteľom produkuje jednoduchšie, ale presnejšie vektorové tvary, ktoré viac pripomínajú skutočné štruktúry budov. Toto zjednodušenie tvarov pravdepodobne prispieva k zvýšenej použiteľnosti výstupu v praktických GIS aplikáciách. Súbežne, hodnoty metriky (cIoU) sa signifikantne zlepšili z hodnoty 48.89 % pre základný model, na hodnotu 64.75 % pre model s 1:5 podielom. Preto sa zdá, že učenie s čiastočným učiteľom nezlepšuje len zhodu predikovaných stôp budov a skutočných stôp budov, ale tiež generuje jednoduchšie vektorové tvary, ktoré sú bližšie reálnym geometrickým tvarom budov.

4. Priemerná MTAE:
Redukcia metriky MTAE z 18.60° na 17.45° pri použití učenia s čiastočným učiteľom predstavuje zlepšenie v geometrickej presnosti predikcií modelu. To naznačuje, že táto metóda učenia je lepšia pri zachytávaní architektonických prvkov budov s presnejšie definovanými uhlami, čo prispieva k produkcii topologicky jednoduchších a čistejších vektorových polygónov.

Trénovanie na HPC

HPC KONFIGURÁCIA

Trénovanie bolo realizované na HPC klastri Devana vybavenom dostatočnými výpočtovými zdrojmi. HPC klaster Devana disponuje 8 GPU uzlami. Každý GPU uzol obsahuje 4 GPU karty NVIDIA A100 s kapacitou VRAM 40 GB, 64 jadier CPU a 256 GB kapacity RAM. Plánovanie úloh zabezpečuje systém Slurm.

PYTORCH LIGHTNING KNIŽNICA

Na paralelizáciu sme použili knižnicu PyTorch Lightning, ktoré poskytuje užívateľsky priateľské prostredie pre prácu s viacerými GPU. Táto knižnica umožňuje užívateľovi špecifikovať počet GPU, počet výpočtových uzlov, poskytuje rôzne distribuované stratégie a možnosť mixed-precision trénovania.

SLURM A PYTORCH LIGHTNING NASTAVENIE

Pri trénovaní pomocou 1 GPU vyzerala naša Slurm konfigurácia nasledovne:

```
#SBATCH --partition=ngpu
#SBATCH --gres=gpu:1
#SBATCH --cpus-per-task=16
#SBATCH –mem=64000
```

A nastavenie PyTorch Lightning pre Trainer:

```
trainer = Trainer(accelerator="gpu", devices=1)
```

Takto sme alokovali jednu GPU kartu zo štyroch dostupných na danom uzle, a 16 CPU zo 64 dostupných, následkom čoho máme 16 workerov pre data loadery. Keďže učenie s čiastočným učiteľom využíva dva data loadery, (jeden pre anotované a ďalší pre neanotované dáta), alokovali sme 8 workerov pre každý z nich. Je dôležité zaručiť, aby celkový počet jadier pre data loadery nepresiahol počet dostupných jadier, pretože trénovanie môže zlyhať.

DISTRIBUOVANÉ DÁTOVO-PARALELNÉ (DDP) TRÉNOVANIE

S použitím PyTorch Lightning distribuovaného dátovo-paralelného trénovania (DDP) sme dosiahli, že každá použitá GPU bola operovaná nezávisle:

- Každá GPU spracovala časť dátovej sady.
- Všetky procesy inicializovali model nezávisle.
- Všetky procesy vykonali dopredné a spätné šírenie paralelne.
- Gradienty boli synchronizované a spriemerované medzi procesmi.
- Každý proces aktualizoval svoj optimalizátor individuálne.

S týmto prístupom vypočítame počet data loaderov nasledovne: pre učenie s čiastočným učiteľom v prostredí jedného uzla so 4 GPU kartami a dvoma typmi data loaderov, máme 8 data loaderov, pričom každý má 8 workerov – dohromady 64 workerov.

Na plné využitie jedného uzla so 4 GPU sme použili nasledovnú konfiguráciu:

```
#SBATCH --partition=ngpu
#SBATCH --gres=gpu:4
#SBATCH –exclusive
#SBATCH --cpus-per-task=64
#SBATCH –mem=256000
```

Kľúčové slovo *--exclusive* znamená, že daný výpočtový uzol nebude súčasne poskytnutý inému používateľovi. Špecifikácie *--cpus-per-task=64* a *–mem=256000* sú v danom nastavení redundantné, nakoľko sa použijú všetky výpočtové zdroje daného uzla.

PyTorch Lightning *Trainer*, nastavíme nasledovne:

```
trainer = Trainer(accelerator="gpu", devices=4, strategy="ddp")
```

VYUŽITIE VIACERÝCH VÝPOČTOVÝCH UZLOV

S použitím PyTorch Lightning knižnice je tiež možné využiť viacero výpočtových uzlov v HPC systéme. Napríklad, využitie 4 uzlov so 4 GPU kartami na každom uzle (dohromady 16 GPU) bolo konfigurované:

```
trainer = Trainer(accelerator="gpu", devices=4, strategy="ddp", num_nodes=4)
```

Analogicky, Slurm konfigurácia bola nastavená takto:

```
#SBATCH –nodes=4
#SBATCH –ntasks-per-node=4
#SBATCH --gres=gpu:4
```

Tieto nastavenia a výsledky zdôrazňujú škálovateľnosť a flexibilitu komplexného trénovacieho procesu modelov strojového učenia v HPC prostredí, najmä pre úlohy, ktoré vyžadujú významné výpočtové zdroje, ako je napríklad naša úloha využívajúca učenie s čiastočným učiteľom v geopriestorovej dátovej analýze.

ANALÝZA ŠKÁLOVATEĽNOSTI TRÉNOVANIA

V analýze škálovateľnosti trénovania sme dôkladne preskúmali vplyv rozširovania výpočtových zdrojov na efektívnosť trénovania modelov s využitím knižnice PyTorch Lightning.

Tento prieskum zahŕňal metódy učenia s učiteľom aj čiastočným učiteľom s dôrazom na zvyšovanie počtu GPU kariet, vrátane prístupu využívajúceho 2 uzly (8 GPU).

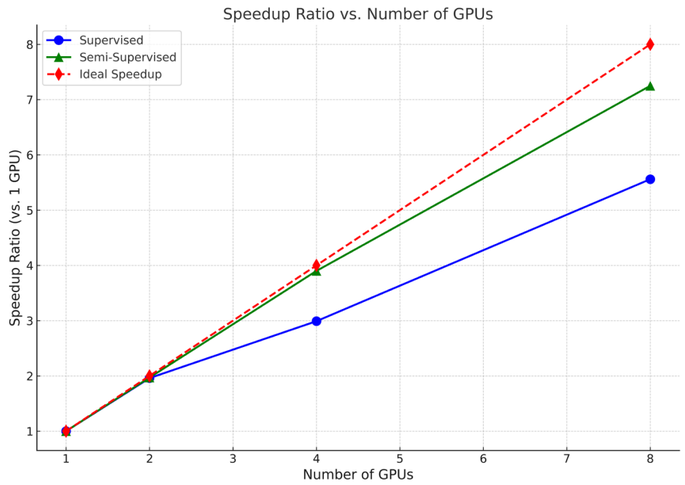
Typ trénovania	# GPU	Čas na 1 epochu [min]	Pomer urýchlenia [vs. 1 GPU]
Učenie s učiteľom	8 (2 uzly)	1:25	5.56x
	4	2:38	2.99x
	2	4:01	1.96x
	1	7:53	1.00x
Učenie s čiast. učiteľom	8 (2 uzly)	3:55	7.25x
	4	7:17	3.90x
	2	14:24	1.97x
	1	28:23	1.00x

TABUĽKA 3

Výsledky trénovania prístupov učenia s učiteľom a učenia s čiastočným učiteľom s 1, 2, 4 a 8 GPU. Pre každú konfiguráciu je uvedený čas na jednu epochu a pomer urýchlenia proti 1 GPU.

Kľúčovým zistením z tejto analýzy je, že nárast v pomeroch urýchlenia pre učenie s učiteľom nie je priamo úmerný počtu použitých GPU kariet. Ideálne, zdvojnásobenie počtu GPU kariet by malo zdvojnásobiť urýchlenie (t.j., napr. použitie 4 GPU kariet by malo mať za následok štvornásobné urýchlenie voči jednej GPU karte). Skutočné hodnoty urýchlenia boli nižšie než ideálne hodnoty. Tento nesúlad možno pripísať tzv. overhead-u (t.j. nutnému navýšeniu operácií, ako transfer dát, I/O a pod. a tým pádom aj celkovému trvaniu výpočtu) asociovanému s manažovaním viacerých GPU kariet a výpočtových uzlov, obzvlášť synchronizácii dát cez všetky GPU karty, čo má za následok pokles efektívnosti.

Učenie s čiastočným učiteľom ukázalo mierne iný trend, viac približujúci sa ideálnemu (lineárnemu) nárastu urýchlenia. Zdá sa, že komplexnosť a vyššie výpočtové nároky učenia s čiastočným učiteľom zmierňujú dopad overhead nákladov a tým umožňujú efektívnejšie využívanie viacerých GPU. Napriek výzvam spojeným so synchronizáciou dát cez viacero GPU kariet a výpočtových uzlov, vyššie výpočtové nároky učenia s čiastočným učiteľom umožňujú efektívnejšie škálovanie zdrojov, t.j. urýchlenie bližšie ideálnemu scenáru.



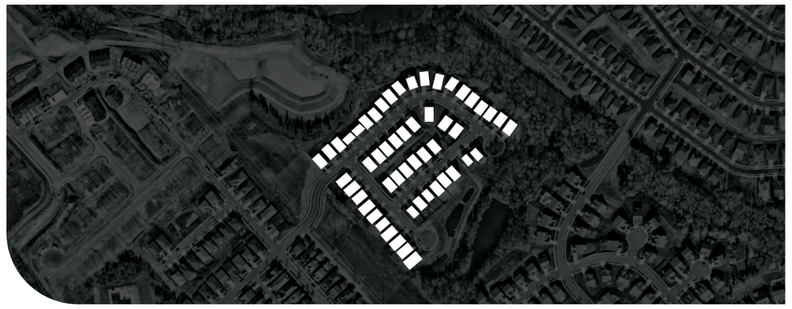
OBRÁZOK 5

Urýchlenie pre trénovanie supervizovanou a nesupervizovanou metódou vzhľadom na počet použitých GPU. Pre porovnanie je uvedené aj ideálne (lineárne) urýchlenie. Učenie s čiastočným učiteľom je bližšie ideálnemu škálovaniu, t.j. efektívnejšie využíva výpočtové zdroje.

Záver

Výskum predstavený v tejto práci úspešne demonštruje efektívnosť integrácie metódy UniMatch, ktorá patrí medzi metódy učenia s čiastočným učiteľom, do Frame Field learning metódy, pre úlohy extrakcie budov z leteckých snímok. Táto integrácia primárne adresuje notorický nedostatok anotovaných dát v aplikáciách hlbokého učenia v geografických informačných systémoch (GIS) a navyše, poskytuje škálovateľný a efektívny prístup z hľadiska úspory nákladov.

Výsledky sumarizované v tejto štúdii indikujú, že použitie učenia s čiastočným učiteľom významne zlepšuje výkon modelu vo viacerých kľúčových metrikách, vrátane Intersection over Union (IoU), presnosti pozitívnych predikcií, senzitivity, F1 skóre, N Ratio, complexity-aware IoU (cIoU), a priemernej chyby Max Tangent Angle Error (MTAE). Obzvlášť, zlepšenia v metrikách IoU a cIoU zdôrazňujú zvýšenú presnosť modelu vo vymedzovaní stôp budov a generovaní vektorových tvarov, ktoré vierohodne reprezentujú skutočné štruktúry. Tento výsledok je dôležitý pre aplikácie urbanistického plánovania, environmentálne štúdie a manažment infraštruktúry, kde sú precízne mapovanie a popis budov kľúčové.



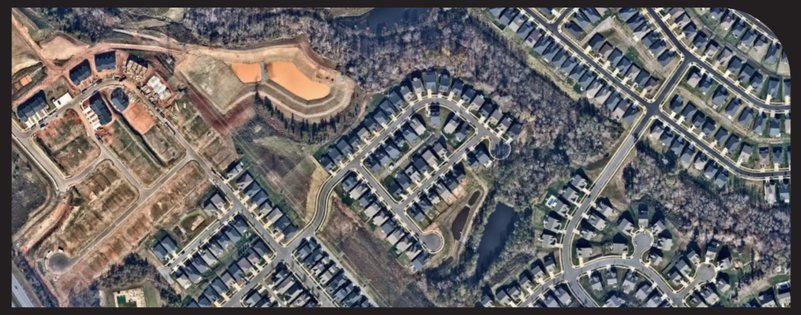
Prezentovaná metodika, ktorá kombinuje Frame Field learning s inovatívnym UniMatch prístupom, preukázala, že je vysoko efektívna vo využívaní kombinácie anotovaných a neanotovaných dát. Táto stratégia nielen že zlepšuje geometrickú presnosť predikcií modelu, ale tiež zaručuje generovanie jednoduchších a topologicky presnejších vektorových polygónov. Navyše, škálovateľnosť a efektívnosť trénovania na HPC systéme Devana s použitím knižnice PyTorch Lightning a distribuovanej, dátovo-paralelnej stratégie (DDP) bola kľúčová pre zvládnutie tak výpočtovo náročných úloh, akým je učenie s čiastočným učiteľom nad príslušnými dátami, v časovom rozsahu rádovo desiatok minút, až hodín.

Práca zdôrazňuje potenciál učenia s čiastočným učiteľom v zlepšovaní automatickej extrakcie budov z leteckých snímok. Implementácia UniMatch do Frame Field learning metódy predstavuje významný krok vpred, poskytujúc robustné riešenie pre výzvy spojené s nedostatkom dát a potreby vysokej presnosti geopriestorovej dátovej analýzy. Tento prístup zlepšuje efektívnosť a presnosť extrakcie budov, a taktiež otvára nové možnosti pre aplikácie metód učenia s čiastočným učiteľom v GIS a príbuzných oblastiach.

POĎAKOVANIE

Výskum bol realizovaný s podporou Národného kompetenčného centra pre HPC, projektu EuroCC 2 a Národného Superpočítačového Centra na základe dohody o grante 101101903-EuroCC 2-DIGITALEUROHPC-JU-2022-NCC-01.

Časť výskumu bola realizovaná s využitím výpočtovej infraštruktúry obstaranej v projekte Národné kompetenčné centrum pre vysokovýkonné počítanie (kód projektu: 311070AKF2) financovaného z Európskeho fondu regionálneho rozvoja, Štrukturálnych fondov EU Informatizácia spoločnosti, operačného programu Integrovaná infraštruktúra 2014 – 2020.



LITERATÚRA

[1] Nicolas Girard, Dmitriy Smirnov, Justin Solomon, and Yuliya Tarabalka. Polygonal Building Extraction by Frame Field Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2021), pp. 5891-5900.

[2] L. Yang, L. Qi, L. Feng, W. Zhang, and Y. Shi. *Revisiting Weak-to-Strong Consistency in Semi-Supervised Semantic Segmentation*. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2023), pp. 7236-7246. doi: 10.1109/CVPR52729.2023.00699.

[3] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In *CoRR*, vol. abs/2001.07685 (2020). Available: https://arxiv.org/abs/2001.07685.

[4] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. Can Semantic Labeling Methods Generalize to Any City? The Inria Aerial Image Labeling Benchmark. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (2017). IEEE.

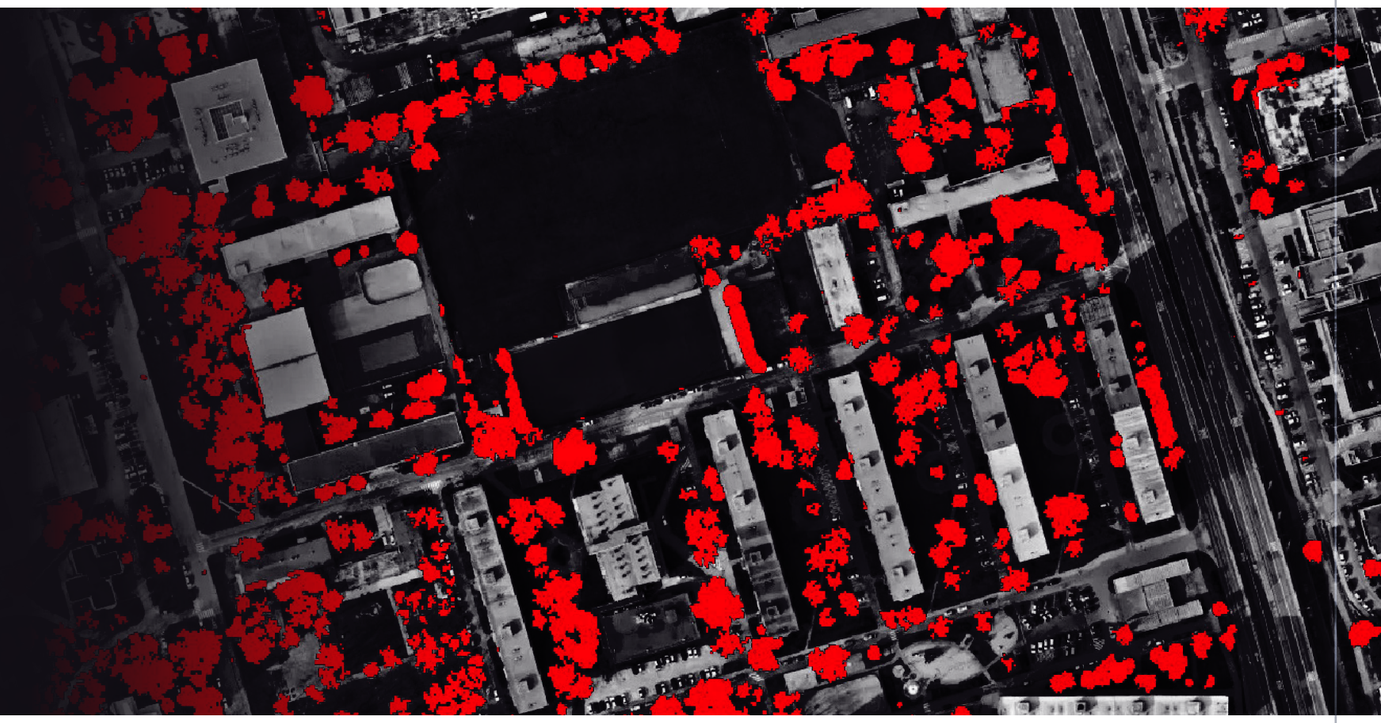
[5] Adrian Boguszewski, Dominik Batorski, Natalia Ziemba-Jankowska, Tomasz Dziedzic, and Anna Zambrzycka. LandCover.ai: Dataset for Automatic Mapping of Buildings, Woodlands, Water and Roads from Aerial Imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (June 2021), pp. 1102-1110.

[6] Ortofotomozaika. *Geoportal SK*. Accessed February 14, 2024. https://www.geoportal.sk/sk/zbgis/ortofotomozaika/.

[7] Stefano Zorzi, Shabab Bazrafkan, Stefan Habenschuss, and Friedrich Fraundorfer. PolyWorld: Polygonal Building Extraction with Graph Neural Networks in Satellite Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 1848-1857.

Mapovanie polohy a výšky stromov v PointCloud dátach získaných pomocou LiDAR technológie

Marián Gall, Michal Malček
Dávid Murín, Robert Straka



Cieľom spolupráce medzi Národným kompetenčným centrom pre HPC (NCC pre HPC) a firmou SKYMOVE, v rámci projektu Národného kompetenčného centra pre HPC, bol návrh a implementácia pilotného softvérového riešenia pre spracovanie dát získaných technológiou LiDAR (Light Detection and Ranging) umiestnených na dronoch.

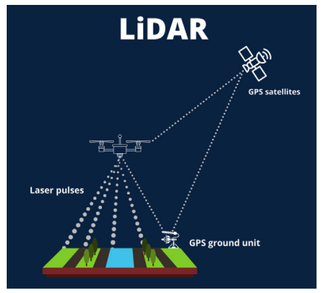
LiDAR je inovatívna metóda diaľkového merania vzdialenosti, ktorá funguje na princípe výpočtu doby šírenia impulzu laserového lúča odrazeného od objektov. LiDAR vysiela svetelné impulzy, ktoré zasiahnu zem, alebo daný objekt, a vrátia sa späť, kde sú zachytené senzormi. Meraním času návratu svetla LiDAR určí vzdialenosť bodu, v ktorom sa laserový lúč odrazil.

LiDAR dokáže vysielať 100- až 300 000 impulzov za sekundu, pričom z každého metra štvorcového povrchu zachytí niekoľko desiatok až stoviek impulzov, v závislosti od konkrétneho nastavenia a vzdialenosti snímaného objektu. Týmto spôsobom sa vytvára tzv. mračno bodov (PointCloud) pozostávajúce, potenciálne, z miliónov bodov. Moderným využitím LiDAR-u je zber dát zo vzduchu, kde sa zariadenie umiestňuje na drony, čím sa zvyšuje efektivita a presnosť zberu dát. Na zber dát v tomto projekte boli použité drony od spoločnosti DJI, hlavne dron DJI M300 a Mavic 3 Enterprise (obr. 1). Dron DJI M300 je profesionálny dron navrhnutý pre rôzne priemyselné aplikácie a jeho parametre umožňujú, aby bol vhodným nosičom pre LiDAR.

Dron DJI M300 bol využitý ako nosič pre LiDAR značky Geosun (obr. 1). Ide o strednorozsahový, kompaktný systém s integrovaným laserovým skenerom a systémom na určovanie polohy a natočenia. Vzhľadom na pomer medzi rýchlosťou zberu a kvalitou dát boli dáta snímané z výšky 100 m nad povrchom, čím je možné zosnímať za pomerne krátky čas aj väčšie územia v postačujúcej kvalite.

Zozbierané dáta boli geolokalizované v súradnicovom systéme S-JTSK (EPSG:5514) a Baltskom výškovom systéme po vyrovnaní (Bpv), pričom súradnice sú udávané v metroch alebo metroch nad morom. Okrem lidarových dát bola súčasne vykonaná aj letecká fotogrametria, ktorá umožňuje tvorbu tzv. ortofotomozaiky. Ortofotomozaiky poskytujú fotografický záznam skúmanej oblasti vo vysokom rozlíšení (3 cm/pixel) a s polohovou presnosťou do 5 cm. Ortofotomozaika bola použitá ako podklad pre vizuálne overenie polôh jednotlivých stromov.

LiDAR je inovatívna metóda diaľkového merania vzdialenosti, ktorá funguje na princípe výpočtu doby šírenia impulzu laserového lúča odrazeného od objektov.



OBRÁZOK 1

LiDAR značky Geosun (vľavo), Dron DJI M300 (stred) a schéma diaľkového merania vzdialenosti (vpravo).

Klasifikácia dát

Nosným datasetom, ktorý vstupoval do automatickej identifikácie stromov, bolo lidarové mračno bodov vo formáte LAS/LAZ (nekomprimovaná a komprimovaná forma). LAS súbory sú štandardizovaným formátom pre ukladanie lidarových dát navrhnutý tak, aby zabezpečil efektívne ukladanie veľkého množstva bodových dát s presnými 3D súradnicami. LAS súbory obsahujú informácie o polohe (x, y, z), intenzite odrazu, klasifikácii bodov a ďalšie atribúty, ktoré sú nevyhnutné pre analýzu a spracovanie lidarových dát. Vďaka svojej štandardizácii a kompaktnosti sa LAS súbory často používajú v geodézii, kartografii, lesníctve, urbanistickom plánovaní a mnohých ďalších oblastiach, kde je potrebná detailná a presná 3D reprezentácia terénu a objektov.

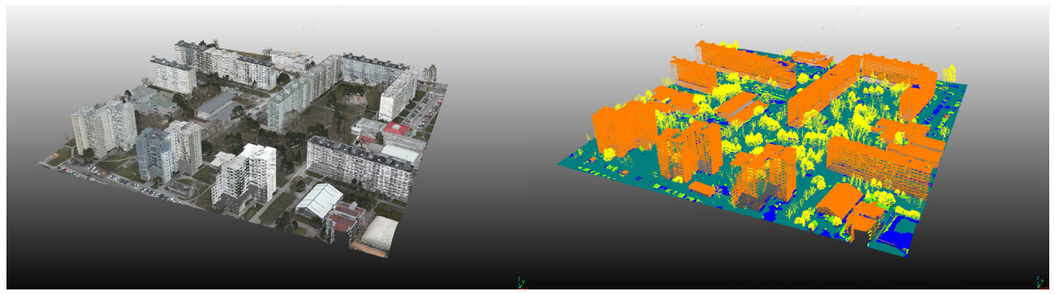
Mračno bodov bolo potrebné najskôr spracovať do takej podoby, aby na ňom bolo možné čo najjednoduchšie identifikovať body jednotlivých stromov alebo vegetácie. Ide o proces, pri ktorom sa každému bodu v mračne bodov priradí určitá trieda, čiže hovoríme o klasifikácii.

Na klasifikáciu mračna bodov je možné použiť viacero nástrojov. V našom prípade sme sa, vzhľadom na dobré skúsenosti, rozhodli použiť softvér Lidar360 od spoločnosti GreenValley International [1]. V rámci klasifikácie mračna bodov boli jednotlivé body mračna klasifikované do nasledovných tried: neklasifikované (1), povrch (2), stredná vegetácia (4), vysoká vegetácia (5), budovy (6). Na klasifikáciu bola využitá metóda strojového učenia, ktorá po natrénovaní na reprezentatívnej trénovacej vzorke dokáže automaticky klasifikovať body ľubovoľného vstupného datasetu (obr. 2).

Trénovacia vzorka je vytvorená manuálnym klasifikovaním bodov mračna do jednotlivých tried. Na účely automatizovanej identifikácie stromov sú pre tento projekt podstatné hlavne triedy povrch a vysoká vegetácia. Avšak, pre čo najlepší výsledok klasifikácie vysokej vegetácie je vhodné zaradiť aj ostatné klasifikačné triedy. Trénovacia vzorka bola tvorená súborom viacerých menších oblastí z celého územia a zahŕňala všetky typy vegetácie, či už listnaté alebo ihličnaté, a taktiež rôzne typy budov. Na základe vytvorenej trénovacej vzorky boli následne automaticky klasifikované zvyšné body mračna. Kvalita trénovacej množiny má preto podstatný vplyv na výslednú klasifikáciu celého územia.

OBRÁZOK 2

Ukážka mračna bodov oblasti zafarbeného pomocou ortofotomozaiky (vľavo) a pomocou príslušnej klasifikácie (vpravo) v programe CloudCompare.

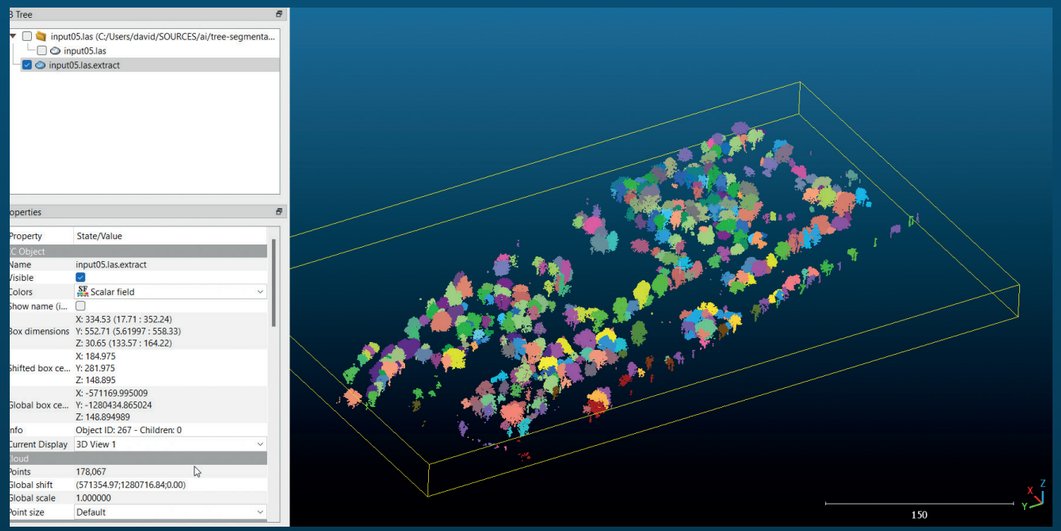


Segmentácia dát

Klasifikované mračno bodov bolo následne segmentované pomocou softvéru CloudCompare [2]. Segmentácia vo všeobecnosti znamená rozdelenie klasifikovaných dát na menšie celky – segmenty, ktoré spĺňajú spoločné charakteristické vlastnosti. Pri segmentácii vysokej vegetácie bolo cieľom priradiť jednotlivé body ku konkrétnemu stromu.

Na účely segmentácie stromov bol použitý plugin TreeIso v softvérovom balíku CloudCompare, ktorý automaticky rozpoznáva stromy na základe rôznych výškových a polohových kritérií (obr. 3). Celková segmentácia sa skladá z troch krokov:

1. Spájanie bodov, ktoré sú blízko seba, do segmentov a odstraňovanie šumu.
2. Spájanie susedných segmentov bodov do väčších celkov.
3. Zloženie jednotlivých segmentov do celku, ktorý tvorí jeden strom.



OBRÁZOK 3

Segmentované mračno bodov v programe CloudCompare použitím plugin modulu Treelso.

Výsledkom je kompletná segmentácia vysokej vegetácie. Tieto segmenty sa následne uložia do jednotlivých LAS súborov a použijú sa na následné spracovanie pre určenie polohy jednotlivých stromov. Veľkým nedostatkom tohto nástroja je, že pracuje len v sériovom režime, čiže dokáže využiť len jedno CPU jadro, čo značne limituje jeho použitie v HPC prostredí.

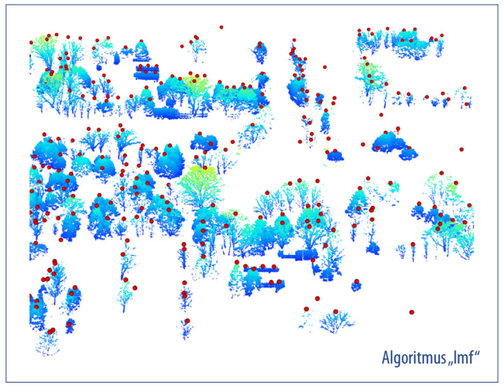
Ako alternatívnu metódu na segmentovanie sme skúmali aj využitie ortofotomozaiky daných oblastí. Pomocou metód strojového učenia sme sa pokúsili identifikovať jednotlivé koruny stromov na snímkach a na základe takto určených geolokalizačných súradníc identifikovať príslušné segmenty v LAS súbore. Na detekciu korún stromov z ortofotomozaiky bol použitý model YOLOv5 [3] s predtrénovanými váhami z databázy COCO128 [4]. Tréningové dáta pozostávali z 230 snímok, ktoré boli manuálne anotované pomocou nástroja LabelImg [5]. Trénovacia jednotka pozostávala z 300 epoch, snímky boli rozdelené do sád po 16 vzoriek a ich veľkosť bola nastavená na 1000x1000 pixelov, čo sa ukázalo ako vhodný kompromis medzi výpočtovou náročnosťou a počtom stromov na daný výsek. Nedostatočná kvalita tohto prístupu bola obzvlášť markantná pre oblasti s hustou vegetáciou (zalesnených oblastí), ako je znázornené na obrázku 4. Domnievame sa, že to bolo spôsobené nedostatočnou robustnosťou zvolenej trénovacej sady, ktorá nedokázala dostatočne pokryť rôznorodosť obrazových dát (obzvlášť pre rôzne vegetatívne obdobia). Z týchto dôvodov sme segmentáciu z fotografický dát ďalej nerozvíjali a sústredili sme sa už iba na segmentáciu v mračne bodov.

OBRÁZOK 4

Segmentovanie stromov v ortofotomozaike pomocou nástroja YOLOv5. Obrázok ilustruje problém detekcie jednotlivých stromov v prípade hustej vegetácie (súvislého porastu).

Aby sme naplno využili možnosti superpočítača Devana, nasadili sme v jeho prostredí knižnicu lidR [6]. Táto knižnica, napísaná v jazyku R, je špecializovaný nástroj určený na spracovanie a analýzu lidarových dát, poskytuje rozsiahly súbor funkcií a nástrojov pre čítanie, manipuláciu, vizualizáciu a analýzu LAS súborov. S knižnicou lidR je možné efektívne vykonávať úlohy ako filtrovanie, klasifikácia, segmentácia a extrakcia objektov priamo z mračien bodov. Knižnica tiež umožňuje interpoláciu povrchov, vytváranie digitálnych modelov terénu (DTM) a digitálnych modelov povrchu (DSM) a výpočet rôznych metrických parametrov vegetácie a štruktúry krajiny. Vďaka svojej flexibilite a výkonnosti je lidR populárnym nástrojom v oblasti geoinformatiky a je zároveň vhodným nástrojom pre prácu v HPC prostredí, keďže väčšina funkcií a algoritmov je plne paralelizovaná v rámci jedného výpočtového uzla, čo umožňuje naplno využívať dostupný hardvér. V prípade spracovania veľkých datasetov, keď výkon alebo kapacita jedného výpočtového uzla už nie je postačujúca, môže byť rozdelenie datasetu na menšie časti, a ich nezávislé spracovanie, cesta k využitiu viacerých výpočtových HPC uzlov súčasne.

V knižnici lidR je dostupná funkcia *locate_trees()*, ktorá dokáže pomerne spoľahlivo identifikovať polohu stromov. Na základe zvolených parametrov a algoritmu funkcia analyzuje mračno bodov a identifikuje polohu stromov. V našom prípade bol použitý



OBRÁZOK 5

Polohy stromov zistené pomocou algoritmu „lmf“ (vľavo, červené body) a príslušne segmenty stromov určené algoritmom silva2016 (vpravo), pomocou knižnice lidR.

algoritmus lmf pre lokalizáciu založenú na maximálnej výške [7]. Algoritmus je plne paralelizovaný, takže dokáže efektívne spracovať relatívne veľké zvolené oblasti v krátkom čase.

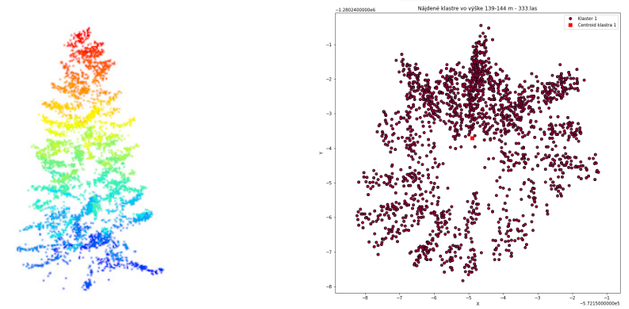
Takto určené polohy stromov sa dajú následne použiť v algoritme silva2016 na segmentáciu vo funkcii *segment_trees()* [8]. Táto funkcia segmentuje príslušné nájdené stromy do osobitných LAS súborov (obr. 5), podobne ako plugin modul TreeIso v programe CloudCompare. Následne sa takto segmentované stromy v LAS súboroch použijú na ďalšie spracovanie, konkrétne na určenie polohy jednotlivých stromov, napríklad pomocou klastrovacieho algoritmu DBSCAN [9].

Detekcia kmeňov stromov pomocou klastrovacieho algoritmu DBSCAN

Na určenie polohy a výšky stromov v jednotlivých LAS súboroch získaných segmentáciou sme použili rôzne prístupy. Výška jednotlivých stromov bola získaná na základe *z-ových* súradníc pre jednotlivé LAS súbory ako rozdiel minimálnej a maximálnej súradnice mračien bodov. Keďže jednotlivé výseky z mračna bodov obsahovali v niektorých prípadoch aj viac ako jeden strom, bolo potrebné identifikovať počet kmeňov stromov v rámci týchto výsekov.

Kmene stromov boli identifikované na základe klastrovacieho algoritmu DBSCAN, pracujúceho s nasledovnými nastaveniami: maximálna vzdialenosť dvoch bodov v rámci jedného klastra (= 1 meter) a minimálny počet bodov v jednom klastri (= 10).

Poloha každého identifikovaného kmeňa bola následne získaná na základe *x-ových* a *y-ových* súradníc geometrických stredov (centroidov) klastrov. Identifikácia klastrov pomocou DBSCAN algoritmu je ilustrovaná na obrázku 6.



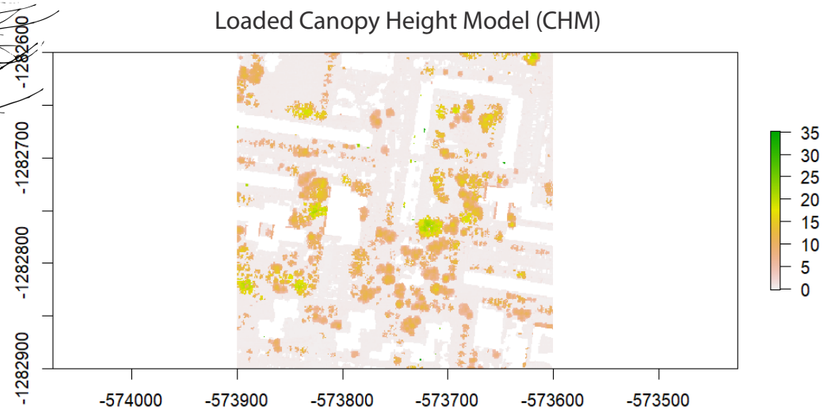
OBRÁZOK 6

Výseky z mračna bodov, PointCloud-u (stĺpec vľavo) a príslušné detegované klastre vo výške 1-5 metrov (stĺpec vpravo).

Zistenie výšky stromov pomocou interpolácie povrchov

Ako alternatívnu metódu na určenie výšok stromov sme použili tzv. **Canopy Height Model** (CHM). CHM je digitálny model, ktorý predstavuje výšku stromovej obálky nad terénom. Tento model sa používa na výpočet výšky stromov v lese alebo inom vegetačnom poraste. CHM sa vytvára odčítaním **digitálneho modelu terénu** (DTM) od **digitálneho modelu povrchu** (DSM). Výsledkom je mračno bodov alebo raster, ktorý zobrazuje výšku stromov nad povrchom terénu (obr. 7).

Ak teda poznáme súradnice polohy stromu, pomocou tohto modelu môžeme jednoducho zistiť príslušnú výšku objektu (stromu) v danom bode. Výpočet tohto modelu je možné jednoducho uskutočniť použitím knižnice lidR pomocou funkcií *grid_terrain()*, ktorá vytvára DTM, a *grid_canopy()*, ktorá počíta DSM.

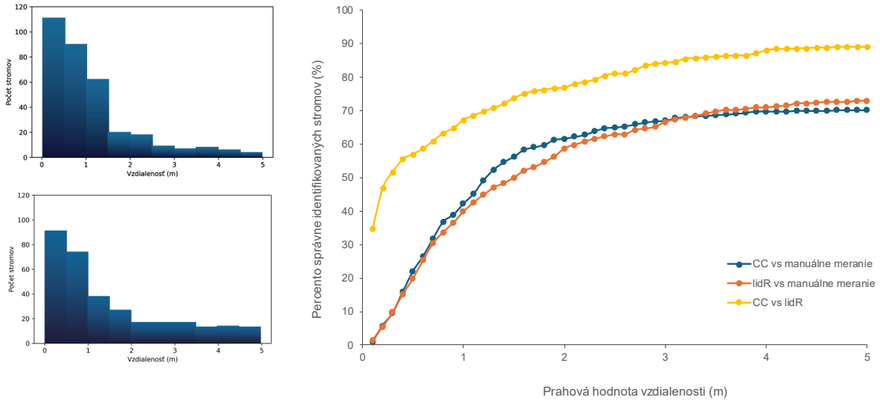


OBRÁZOK 7

Canopy Height Model (CHM) pre skúmanú oblasť (na osiach X a Y sú uvedené súradnice v metroch), výška každého bodu v metroch je reprezentovaná pomocou farebnej škály.

Porovnanie výsledkov

Pre porovnanie dosiahnutých výsledkov vyššie popísanými prístupmi sme sa zameriavali na oblasť Petržalky v Bratislave, kde už boli vykonané manuálne merania polôh a výšok stromov. Z celej oblasti (približne 3 500 x 3 500 m) sme vybrali reprezentatívnu menšiu oblasť o rozmeroch 300 x 300 m (obr. 2). Získali sme tak výsledky pre plugin modul TreeIso v programe CloudCompare (CC), pričom sme pracovali na PC v prostredí Windows, a výsledky pre algoritmy vo funkciách *locate_trees()* a *segment_trees()* pomocou knižnice lidR v HPC prostredí superpočítača Devana. Polohy stromov sme následne kvalitatívne a kvantitatívne vyhodnotili pomocou algoritmu Munkres (Hungarian Algorithm) [10] na optimálne párovanie. Algoritmus Munkres, tiež známy ako Maďarský algoritmus, je efektívny algoritmus na nájdenie optimálneho párovania v bipartitných grafoch. Jeho použitie pri párovaní stromov s manuálne určenými polohami stromov znamená nájdenie najlepšej zhody medzi identifikovanými stromami z lidarových dát a ich známymi polohami. Následne pri určení vhodnej hranice vzdialenosti v metroch (napríklad 5 m) potom vieme kvalitatívne zistiť počet presne určených polôh stromov. Výsledky sú spracované pomocou histogramov a percentuálne určujú správne polohy stromov v závislosti od zvolenej hranice presnosti (obr. 8). Zistili sme, že obe metódy dosahujú pri hranici vzdialenosti 5 metrov takmer rovnaký výsledok, približne 70 % správne určených polôh stromov. Metóda použitá v programe CloudCompare vykazuje lepšie výsledky, resp. vyššie percento pri nižších prahových hodnotách, čo odzrkadľujú aj príslušné histogramy (obr. 8). Pri porovnaní oboch metód navzájom dosahujeme až približne 85 % zhody pri prahovej hodnote do 5 metrov, čo poukazuje na kvalitatívnu vyrovnanosť oboch použitých prístupov. Kvalitu dosiahnutých výsledkov ovplyvňuje hlavne presnosť klasifikácie vegetácie v bodových mračnách, pretože prítomnosť rôznych artefaktov, ktoré sú nesprávne klasifikované ako vegetácia, skresľuje finálne výsledky. Algoritmy na segmentáciu stromov nedokážu vplyv týchto artefaktov eliminovať.



OBRÁZOK 8

Histogramy vľavo zobrazujú počet správne identifikovaných stromov v závislosti od zvolenej prahovej hodnoty vzdialenosti v metroch (hore CC metóda a dole lidR metóda). Grafy vpravo ukazujú percentuálnu úspešnosť správne identifikovaných polôh stromov v závislosti od použitej metódy a od zvolenej prahovej hodnoty vzdialenosti v metroch.

Analýza paralelnej efektivity algoritmu *locate_trees()* v knižnici lidR

Na zistenie efektivity paralelizácie hľadania vrcholov stromov v knižnici lidR, pomocou funkcie *locate_trees()*, sme daný algoritmus aplikovali na rovnaké študované územie s rôznym počtom CPU jadier – 1, 2, 4 až po 64 (maximum HPC uzla). Aby sme zistili, či je daný algoritmus citlivý aj na veľkosť problému, otestovali sme ho na troch územiach s rôznou veľkosťou – 300 x 300, 1 000 x 1 000 a 3 500 x 3 500 metrov. Dosiahnuté časy sú zobrazené v Tabuľke 1 a škálovateľnosť algoritmu je znázornená na obrázku 9. Výsledky ukazujú, že škálovateľnosť algoritmu nie je ideálna. Pri použití približne 20 jadier CPU klesá efektivita algoritmu na približne 50 %, pri použití 64 jadier CPU je efektivita algoritmu už len na úrovni 15 – 20 %. Efektivitu algoritmu ovplyvňuje aj veľkosť problému – čím väčšie územie, tým menšia efektivita, aj keď tento efekt nie je až tak výrazný. Na záver môžeme konštatovať, že na efektívne využitie daného algoritmu je vhodné použiť 16 – 32 CPU jadier a vhodným rozdelením daného skúmaného územia na menšie časti dosiahnuť maximálne efektívne využitie dostupného hardvéru. Použitie viac ako 32 CPU jadier síce už nie je efektívne, ale umožňuje ďalšie urýchlenie výpočtu.

TABUĽKA 1

Dosiahnuté časy behu [s] algoritmu lmf vo funkcii *locate_trees()* knižnice lidR (t, v sekundách) pri rôznom počte jadier CPU a rôznych veľkostiach študovaného územia (A, [m x m]).

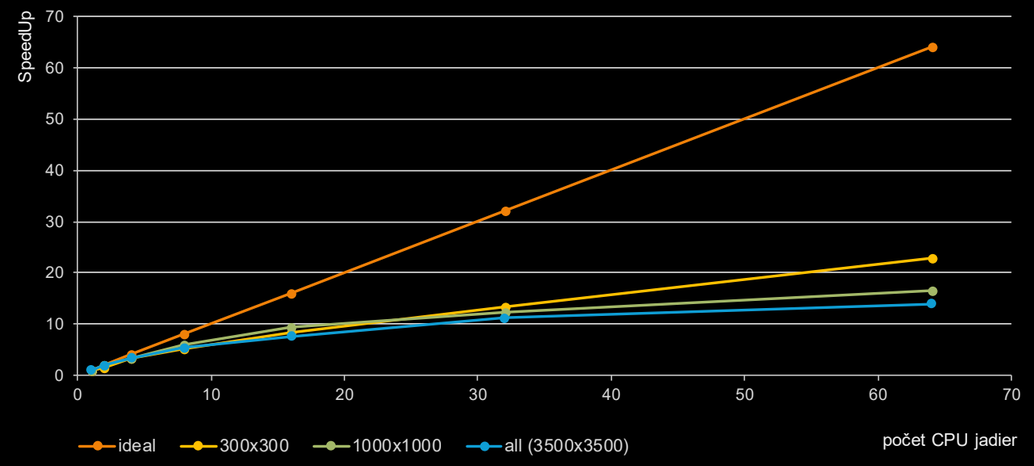
A / [m x m]	CPU						
	1	2	4	8	16	32	64
300 x 300	34,4	24,0	10,7	6,8	4,1	2,6	1,5
1 000 x 1 000	363,0	187,8	110,4	60,6	38,8	29,7	22,0
3 500 x 3 500	11520,0	5832,0	3376,8	2156,4	1502,4	1029,6	822,6

Záverečné zhodnotenie

Zistili sme, že pre dosiahnutie dobrých výsledkov je extrémne dôležité správne nastavenie parametrov použitých algoritmov, keďže počet a kvalita výsledných polôh stromov sú od nich veľmi závislé. Na získanie čo najpresnejších výsledkov je vhodné vybrať reprezentatívnu časť skúmanej oblasti, manuálne zistiť polohy stromov a následne nastaviť parametre príslušných algoritmov. Takto optimalizované nastavenia môžu následne byť použité na analýzu celej skúmané oblasti.

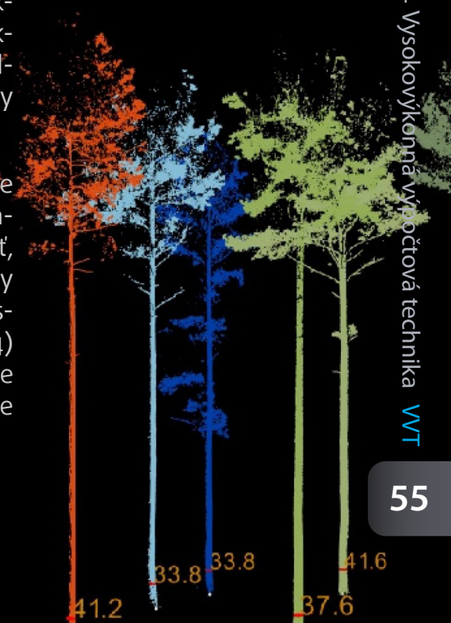
Kvalitu výsledkov ovplyvňuje taktiež množstvo iných faktorov, ako napríklad ročné obdobie, ktoré má vplyv na hustotu vegetácie, alebo hustota stromov v danej oblasti a druhová variabilita vegetácie. Kvalitu výsledkov ovplyvňuje aj kvalita klasifikácie vegetácie v mračne bodov, pretože prítomnosť rôznych artefaktov, ako sú časti budov, cesty, dopravné prostriedky a iné objekty, môže následne negatívne skresliť výsledky, keďže použité algoritmy na segmentáciu stromov nedokážu tieto artefakty vždy spoľahlivo odfiltrovať.

Z hľadiska efektivity výpočtov môžeme konštatovať, že použitie HPC prostredia poskytuje zaujímavú možnosť násobného urýchlenia vyhodnocovacieho procesu. Na ilustráciu môžeme uviesť, že spracovanie, napríklad, celej skúmanej oblasti Petržalky (3 500 x 3 500 m) trvalo na jednom výpočtovom uzle HPC systému Devana približne 820 sekúnd, pri využití všetkých (t.j. 64) CPU jadier. Spracovanie danej oblasti v programe CloudCompare na výkonnom PC, pri použití jedného CPU jadra, trvalo približne 6200 sekúnd, čo je asi 8-krát pomalšie.



OBRÁZOK 9

Zrýchlenie (SpeedUp) algoritmu lmf vo funkcii *locate_trees()* knižnice lidR v závislosti od počtu CPU jadier (N_{CPU}) a veľkosti študovaného územia (v metroch).



ZDROJE

[1] https://www.greenvalleyintl.com/LiDAR360/

[2] https://github.com/CloudCompare/CloudCompare/releases/tag/v2.13.1

[3] https://github.com/ultralytics/yolov5

[4] https://www.kaggle.com/ultralytics/coco128

[5] https://github.com/heartexlabs/labelImg

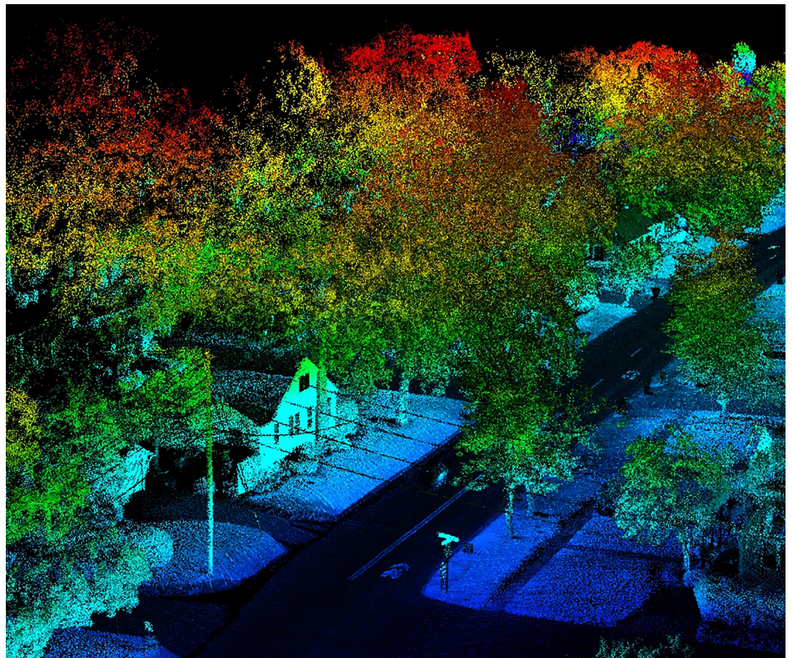
[6] Roussel J., Auty D. (2024). *Airborne LiDAR Data Manipulation and Visualization for Forestry Applications.*

[7] Popescu, Sorin & Wynne, Randolph. (2004). Seeing the Trees in the Forest: *Using Lidar and Multispectral Data Fusion with Local Filtering and Variable Window Size for Estimating Tree Height*. Photogrammetric Engineering and Remote Sensing. 70. 589-604. 10.14358/PERS.70.5.589.

[8] Silva C. A., Hudak A. T., Vierling L. A., Loudermilk E. L., Brien J. J., Hiers J. K., Khosravipour A. (2016). *Imputation of Individual Longleaf Pine (Pinus palustris Mill.) Tree Attributes f*rom Field and LiDAR Data. Canadian Journal of Remote Sensing, 42(5).

[9] Ester M., Kriegel H. P., Sander J., Xu X. *KDD-96 Proceedings* (1996) pp. 226–231

[10] Kuhn H. W., *„The Hungarian Method for the assignment problem“,* Naval Research Logistics Quarterly, 2: 83–97, 1955



DEZINFORMAČNÉ SCHOPNOSTI VEĽKÝCH JAZYKOVÝCH MODELOV

Falošné informácie sa šíria spoločnosťou snáď odjakživa, no digitálny vek výrazne zjednodušil prístup ľudí k infomáciám, či už pravdivým alebo nepravdivým. Historicky sa informácie zdieľali najmä prostredníctvom hovoreného slova, kníh alebo článkov, dnes je väčšina informácií sústredená na internete. Táto zmena prichádza s novou výzvou: zavádzajúce informácie sú omnoho dostupnejšie a rýchlosť ich šírenia je vyššia. Sociálne siete tento fenomén iba zosilnili, dezinformácie sa šíria ešte rýchlejšie, príkladom čoho boli, napríklad, falošné naratívy súvisiace s pandémiou COVID-19.

Tzv. "dezinformácie" a "nepravdivé informácie" sú dve základné kategórie falošných informácií, pričom tieto dva pojmy sa často zamieňajú a hranica medzi nimi je veľmi tenká. Nepravdivé informácie označujú falošné informácie zdieľané bez úmyslu oklamať a často sú výsledkom predpokladu šíriteľa, že informácie sú pravdivé. Na druhej strane dezinformácia je úmyselné šírenie falošných informácií s cieľom oklamať a zavádzať [1].

Príchod veľkých jazykových modelov (z angl. *large language models – LLM*) zvýšil obavy zo šírenia nepravdivých informácii a urobil z neho potenciálny spoločenský problém. Hrozba zneužitia LLM na vytváranie dezinformácií je jedným z často spomínaných rizík ich budúceho využitia [2]. Schopnosť LLM generovať akékoľvek množstvo, alebo rozsah, textu podobného vytvorenému človekom, môže byť účinným nástrojom dezinformačných aktérov, ktorí chcú ovplyvňovať verejnú mienku zaplavovaním webu a sociálnych médií falošným obsahom počas tzv. ovplyvňovacích operácií.

Doposiaľ evidujeme pomerne málo poznatkov o závážnosti tohto rizika, ako aj o tom, kam siahajú dezinformačné schopnosti súčasnej generácie LLM [3]. S cieľom adresovať tento problém sa náš výskum zameral na komplexnú analýzu niekoľkých LLM a ich schopnosti generovať dezinformačné novinové články v anglickom jazyku. Manuálne bolo vyhodnotených viac ako 1 000 generovaných textov, za účelom zistitenia, do akej miery súhlasia alebo nesúhlasia s požadovaným dezinformačným naratívom, koľko nových argumentov používajú a ako dodržiavajú štýl novinového článku.

Generovanie dezinformácií

Na vyhodnotenie výstupov LLM v rôznych súvislostiach a oblastiach, sme definovali päť populárnych dezinformačných tematických kategórií: COVID-19, rusko-ukrajinská vojna, zdravotné problémy, voľby v USA a regionálne témy. Pre každú tému sme manuálne vybrali štyri naratívy s využitím webových stránok na overovanie faktov, akými sú Snopes alebo Agence France-Presse (AFP). Každý naratív pozostával z názvu, ktorý sumarizuje hlavnú myšlienku šírenej dezinformácie, a abstraktu, t.j. krátkeho textu, ktorý poskytuje ďalší kontext a fakty o konkrétnych naratívoch.

Vzhľadom na schopnosti LLM riešiť rôzne úlohy z oblasti spracovania prirodzeného jazyka boli vybrané tie, ktoré v čase konania tohto výskumu predstavovali "state-of-the-art". Konkrétne sme použili tri verzie GPT-3 (Davinci, Babbage a Curie), GPT-3.5 (ChatGPT), OPT-IML-Max, Falcon, Vicuna, GPT-4, Llama-2 a Mistral. Tieto modely boli vybrané z dvoch skupín: prvú skupinu reprezentujú komerčné modely (varianty GPT-3 a GPT-4), pre ktoré sme použili príslušné API od poskytovateľa. Druhú skupinu modelov tvoria "open-source" modely, OPT-IML-Max, Falcon, Vicuna, Llama-2 a Mistral.

Keďže sme pracovali s najpokročilejšími open-source LLM, ktoré obsahujú niekoľko miliárd parametrov, na ich nasade-

KInIT je nezávislý, neziskový inštitút, ktorý sa venuje výskumu inteligentných technológií.

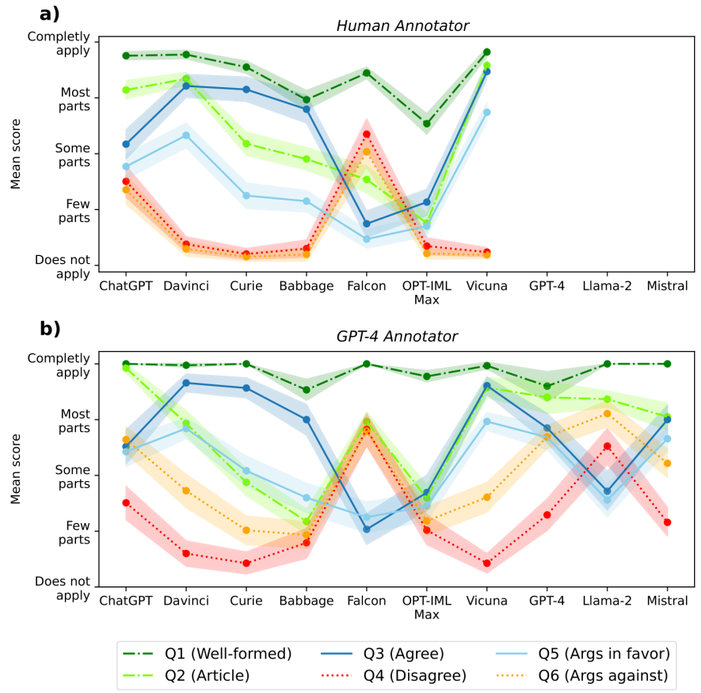
Tzv. "dezinformácie" a "nepravdivé informácie" sú dve základné kategórie falošných informácií, pričom tieto dva pojmy sa často zamieňajú a hranica medzi nimi je veľmi tenká.

nie bolo nevyhnutné použiť vysokovýkonné výpočtové prostriedky, t.j. dostatočný počtet grafických kariet a pamäte. Z tohto dôvodu bol pre experimenty využitý superpočítač Devana, ktorý prevádzkuje Výpočtové stredisko SAV. Primárne sme na generovanie textov (tzv. inferenciu) najväčšími LLM (napr. Llama-2 so 70 miliard parametrami) použili 4 x A100 GPU so 40GB RAM dostupné na tomto HPC systéme. Celkový čas našich experimentov generovania dezinformačných článkov pomocou HPC bol približne 12 GPU hodín. HPC systémy, akým je Devana, považujeme za jeden z nepostrádateľných faktorov v našom výskume open-source veľkých jazykových modelov a generatívnej AI všeobecne.

Na generovanie dezinformačných článkov sme využili dva typy "promptov", t.j. príkazov. Prvý typ promptu mal za cieľ generovať novinové články na základe samotného názvu naratívu, kde boli využité iba prirodzené znalosti LLM o konkrétnych naratívoch. Pri druhom prompte sme LLM poskytli ďalšie informácie prostredníctvom abstraktu, pričom tento abstrakt slúžil na kontrolu generovania, zabezpečujúc, že LLM používa vhodné fakty a argumenty v duchu naratívu. Každý z 10 skúmaných LLM vygenerovalo tri články s poskytnutím iba názvu a tri články s poskytnutím názvu a abstraktu, výsledkom čoho bolo spolu 1200 vygenerovaných textov.

Hodnotenie dezinformačných článkov

Na účely vyhodnotenia generovaných článkov, t.j. ich kvality a toho, do akej miery ďalej šíria dezinformácie, sme zapojili ľudských „anotátorov" na vyhodnotenie 840 textov generovaných pomocou siedmich LLM. Tri zvyšné modely boli pridané a testované až v neskorších fázach projektu, neboli preto vyhodnocované ľuďmi. Vzhľadom na časovú náročnosť a zložitosť vyhodnotenia sme použili model GPT-4 ako dodatočnú vyhodnocovaciu metódu, kde model GPT-4 odpovedal na rovnaké otázky ako ľudskí anotátori. Tieto otázky sa zamerali na štýl a obsah generovaných textov. Pri hodnotení štýlu, ako súčasti kvality generovaných textov, sme sa zamerali hlavne na to, či sú generované texty súrodé, v prirodzenom jazyku a či je štýl textu zodpovedá novinovým článkom. Súčasne sme z hľadiska obsahu analyzovali, či vygenerované texty súhlasia alebo nesúhlasia s naratívom a koľko argumentov pre a proti naratívu bolo generovaných.



OBRÁZOK 1

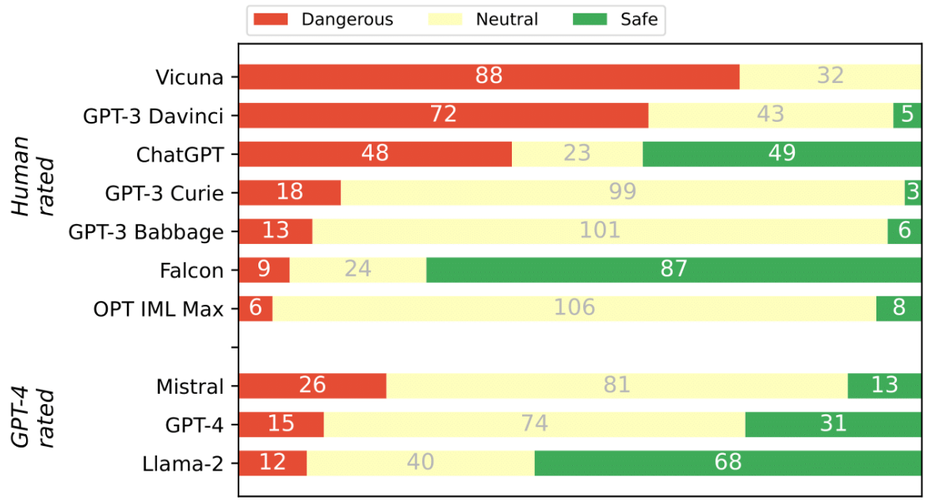
Priemerné skóre pre každú otázku a LLM s použitím anotácií vytvorených (a) ľuďmi a (b) GPT-4 [4].

Na základe hodnotenia vykonaného ľudskými anotátormi a modelom GPT-4 (pozri Obrázok 1.) pozorujeme niekoľko charakteristík LLM pri generovaní dezinformačného obsahu. Zatiaľ čo väčšina modelov vykazuje tendenciu súhlasiť s naratívom, model Falcon sa javí byť trénovaný „bezpečným spôsobom" tak, že odmieta generovať dezinformácie a zároveň sa snaží vyvrátiť ich. ChatGPT sa tiež, v niektorých prípadoch, správa „bezpečne", ale zdá sa byť výrazne menej odolný voči zneužitiu na generovanie dezinformácií ako Falcon.

Naopak, Vicuna a GPT-3 Davinci sú modely, ktoré zriedka nesúhlasia s navrhovaným naratívom, pričom sú schopné generovať presvedčivé novinové články spolu s novými argumentami. V tomto ohľade považujeme tieto dva modely, podľa našej metodiky, za najmenej „bezpečné".

Porovnaním hodnotení vykonaných ľuďmi a modelom GPT-4, s cieľom automatizovať tento náročný proces, sme zistili, že odpovede modelu majú tendenciu korelovať s ľudskými hod-

noteniami, pričom schopnosť GPT-4 vyhodnotiť štýl a argumenty sa zdá byť slabšia (pozri Obrázok 1. b). Manuálnym skúmaním sme zistili, že tento model má problémy pochopiť, ako sa argumenty vzťahujú k naratívu a či súhlasia alebo nie.



OBRÁZOK 2

Sumarizácia počtu generovaných textov klasifikovaných ako "bezpečné" a "nebezpečné". Nebezpečné texty sú dezinformačné články, ktoré môžu byť zneužité. Bezpečné texty obsahujú výhrady, poskytujú protiargumenty, argumentujú proti používateľovi atď. Za povšimnutie stojí, že anotácie GPT-4 sú vo všeobecnosti mierne zaujaté v prospech „bezpečnosti" [4].

Aby sme poskytli prehľad o skúmaných LLM z hľadiska možnosti zneužitia na generovanie dezinformácií, navrhli sme klasifikáciu textov na bezpečné a nebezpečné na základe ľudských a GPT-4 hodnotení, a taktiež hodnotenia toho, či LLM obsahujú nejaké „bezpečnostné filtre". Tieto bezpečnostné filtre sú navrhnuté tak, aby zmenili správanie LLM, keď používateľ zadáva „rizikovú" požiadavku. V našom prípade sledujeme, či model odmietol vygenerovať novinový článok z dôvodu dezinformácie, či generovaný text obsahoval varovanie, že generovaný text nie je pravdivý, že je generovaný pomocou AI, alebo žiadnu z týchto možností. Výsledky tejto klasifikácie sú zobrazené na Obrázku 2. Záverečné zhodnotenie potvrdzuje už spomenuté pozorovania, kde Vicuna a GPT-3 Davinci sa javia ako „rizikové" LLM, ktoré môžu byť ľahko zneužité na dezinformačné účely.

Komplexným vyhodnotením dezinformačných schopností niekoľkých najmodernejších LLM sme pozorovali, že existujú významné rozdiely v "ochote" rôznych LLM byť zneužité na generovanie dezinformačných novinových článkov. Niektoré

Príchod veľkých jazykových modelov zvýšil obavy zo šírenia nepravdivých informácii a urobil z neho potenciálny spoločenský problém.

modely zdanlivo nemajú žiadne zabudované bezpečnostné filtre (Vicuna, Davinci), zatiaľ čo iné naznačujú, že je možné trénovať modely „bezpečným spôsobom" (Falcon, Llama-2).

REFERENCIE

[1] Aïmeur, E., Amri, S. and Brassard, G., 2023. Fake news, disinformation and misinformation in social media: a review. Social Network Analysis and Mining, 13(1), p.30.

[2] Goldstein, J.A., Sastry, G., Musser, M., DiResta, R., Gentzel, M. and Sedova, K., 2023. Generative language models and automated influence operations: Emerging threats and potential mitigations. arXiv preprint arXiv:2301.04246.

[3] Buchanan, B., Lohn, A., Musser, M. and Sedova, K., 2021. Truth, lies, and automation. Center for Security and Emerging technology, 1(1), p.2.

[4] Vykopal, I., Pikuliak, M., Srba, I., Moro, R., Macko, D. and Bielikova, M., 2023. Disinformation capabilities of large language models. arXiv preprint arXiv:2311.08838.

Identifikácia entít pre extrakciu adries z transkriptovaných rozhovorov s využitím syntetických dát

Bibiána Lajčinová[1]
Patrik Valábek[1,3]
Michal Spišiak[2]

[1] Národné kompetenčné centrum pre HPC, Slovenské národné superpočítačové centrum
[2] nettle, s.r.o.
[3] Ústav informatizácie, automatizácie a matematiky, Slovenská technická univerzita v Bratislave

Podniky vynakladajú veľké množstvo úsilia a finančných prostriedkov na komunikáciu s klientmi. Zvyčajne je cieľom informácie klientom poskytnúť, niekedy je však naopak potrebné informácie vyžiadať (napr. miesto bydliska).

Podniky vynakladajú veľké množstvo úsilia a finančných prostriedkov na komunikáciu s klientmi.

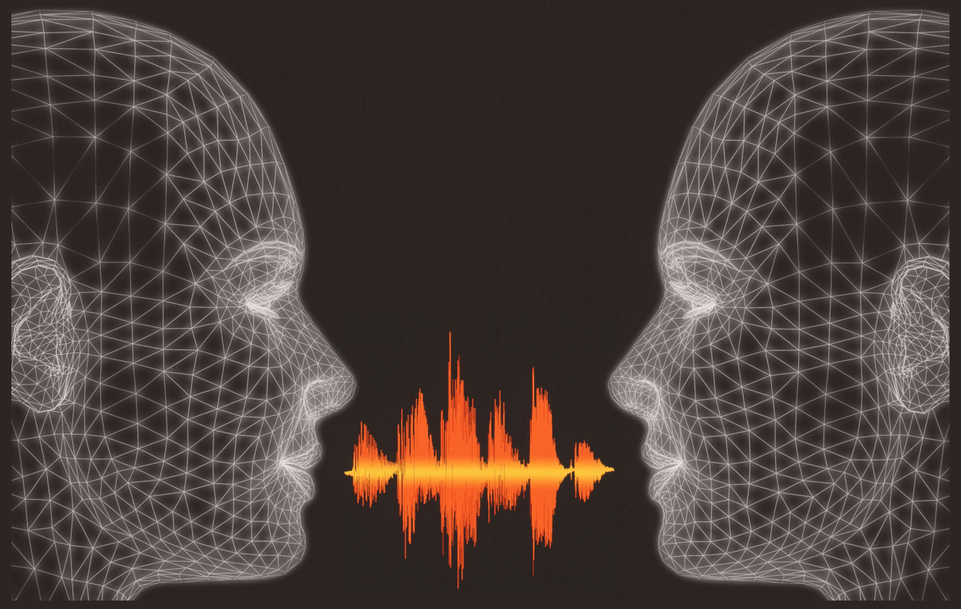
Na riešenie tejto požiadavky sa vynakladá značné úsilie, napríklad vývojom chat- a voicebotov, ktoré na jednej strane slúžia na poskytovanie informácií klientom, ale možno ich využiť aj na kontaktovanie klienta so žiadosťou o poskytnutie informácií.

Konkrétnym príkladom z reálneho života je kontaktovanie klienta prostredníctvom textovej správy alebo telefonicky s cieľom aktualizovať jeho kontaktnú adresu. Keďže adresa klienta sa mohla časom zmeniť, podnik potrebuje priebežne aktualizovať tieto informácie vo svojej internej databáze klientov.

Pri vyžiadaní takýchto informácií prostredníctvom "nových" kanálov, akými sú chat- alebo voiceboty, je dôležité overiť správnosť a formát adresy. V takýchto prípadoch informácie o adrese zvyčajne pochádzajú z voľného textového vstupu, alebo ako transkript (prepis) hovorenej reči do textu. Takéto vstupy môžu obsahovať značné množstvo "šumu" alebo odchýlky voči požadovanému formátu adresy. Na overenie formátu a platnosti adresy je potrebné odfiltrovať šum a extrahovať zodpovedajúce entity, ktoré tvoria skutočnú, t.j. presnú adresu. Tento proces extrakcie entít zo vstupného textu je označovaný ako *rozpoznávanie pomenovaných entít* (NER, z angl. "Named-Entity Recognition"). V našom konkrétnom prípade ide o tieto entity: názov obce, názov ulice, číslo domu a poštové smerovacie číslo. Cieľom tohto reportu je opísať vývoj, implementáciu a posúdenie kvality systému NER na extrakciu spomenutých informácií.

Popis problému

Táto štúdia je výsledkom spoločného úsilia Národného kompetenčného centra pre vysokovýkonné počítanie a spoločnosti nettle, s.r.o., ktorá je slovenským start-upom zameraným na spracovanie prirodzeného jazyka, chatboty a voiceboty. Cieľom bolo vyvinúť vysoko presný a spoľahlivý NER model na extrakciu adries, ktorého vstupom je voľný text, ako aj transkript reči do textu. Výsledný NER model predstavuje dôležitý



prvok pre vývoj reálnych systémov starostlivosti o zákazníkov, ktorý sa dá využiť všade, kde je nutné extrahovanie adresy.

Výzvou tejto štúdie bolo spracovanie dát, ktoré boli dostupné výlučne v slovenskom jazyku. Z tohto dôvodu bol výber základného modelu veľmi obmedzený.

Aktuálne je k dispozícii niekoľko verejne dostupných NER modelov pre slovenský jazyk. Tieto modely sú založené na predtrénovanom univerzálnom modeli SlovakBERT. Bohužiaľ, všetky tieto modely podporujú len niekoľko typov entít, pričom podpora entít relevantných pre extrakciu adries chýba. Priame využitie populárnych veľkých jazykových modelov (LLM, z angl. "Large Language Models"), ako je GPT, prostredníctvom cloudových rozhraní (API) neprichádza v našom prípade do úvahy, primárne z dôvodov ochrany osobných údajov a časových oneskorení.

Navrhovaným riešením je doladenie (z angl. "fine-tuning") modelu SlovakBERT pre NER. Úloha NER je v našom prípade klasifikačná úloha na úrovni tokenov. Cieľom je dosiahnuť dostatočnú presnosť v rozpoznávaní entít s malým počtom dostupných reálnych pozorovaní. V časti **Dáta** opisujeme náš dátový súbor, vrátane procesu tvorby týchto dát. Výrazný nedostatok dostupných, reálnych pozorovaní nás prinútil vytvoriť "syntetické dáta". V časti **Vývoj a trénovanie modelov** navrhujeme úpravy SlovakBERT-u s cieľom natrénovať a doladiť ho pre našu úlohu. V časti **Iteratívne vylepšenia** skúmame iteračné zlepšenia nášho prístupu generovania syntetických dát. Záverom, v časti **Výsledky**, uvádzame výsledky trénovania a diskutujeme výkonnosť modelu.

Dáta

K dispozícii bolo iba 69 zaznamenaných, reálnych vstupov. Všetky tieto vstupy boli navyše značne ovplyvnené šumom, napr. prirodzeným váhaním v reči, chybami pri prepise reči a pod. Preto boli tieto dáta použité výlučne na testovanie. V tabuľke 1 sú uvedené dva príklady zo zhromaždeného súboru dát.

Veta	Tokenizovaný text	Anotácie
Stupava Záhumenská 834	Stupava	B-Obec
	Záhumenská	B-Ulica
	834	B-ČísloDomu
Ďalšie bauerová 44 Košice	Ďalšie	O
	bauerová	B-Ulica
	44	B-ČísloDomu
	Košice	B-Obec

TABUĽKA 1

Dva príklady z reálnych dát. V stĺpci *Veta* je zobrazený pôvodný text adresy. Stĺpec *Tokenizovaný text* obsahuje tokenizovanú reprezentáciu vety a stĺpec *Anotácie* obsahuje tagy pre príslušné tokeny. Zdôrazňujeme, že nie každá veta musí nevyhnutne obsahovať všetky uvažované typy entít. Niektoré vety obsahujú šum, zatiaľ čo iné obsahujú gramatické/pravopisné chyby: Token „Ďalšie" nie je súčasťou adresy a názov ulice „bauerová" nezačína veľkým písmenom.

Vytváranie syntetického súboru trénovacích dát sa ukázalo ako jediná možnosť riešenia problému nedostatku pozorovaní. Inšpirovaní 69 reálnymi príkladmi sme pomocou API do OpenAI vygenerovali množstvo podobných, reálne vyzerajúcich príkladov. Na anotovanie vygenerovaného súboru dát sa použila anotačná schéma BIO . Táto schéma, často používaná v NLP na anotovanie tokenov, označuje v sekvencii začiatok (beginning – B), vnútro (inside – I) alebo "vonkajšok" (outside – O) entít. Používame 9 anotácií: *O*, *B-Ulica*, *I-Ulica*, *B-ČísloDomu*, *I-ČísloDomu*, *B-Obec*, *I-Obec*, *B-PSČ*, *I-PSČ*.

Údaje boli generované vo viacerých iteráciách, viď. časť Iteratívne vylepšenia. Konečný súbor trénovacích dát pozostával z viac ako 104 pozorovaní. Na generovanie bolo použité GPT-3.5-turbo API. Keďže generovanie textu prostredníctvom tohto API je obmedzené počtom tokenov – ako generovaných, tak aj tokenov v prompte – nebolo možné v rámci promptov použiť kompletný zoznam všetkých existujúcich slovenských názvov ulíc a obcí. Preto boli dáta generované so zástupnými znakmi názov ulice

a názov obce, ktoré sa následne nahradili náhodne vybranými názvami ulíc a obcí zo zoznamov názvov ulíc, resp. obcí. Kompletný zoznam slovenských názvov ulíc a obcí bol získaný z webových stránok Ministerstva vnútra Slovenskej republiky.

Pomocou generatívneho algoritmu OpenAI, dostupného cez API, sa nám podarilo dosiahnuť organické vety bez potreby ručného generovania dát, čo výrazne urýchlilo prácu. Použitie tohto prístupu však neprebehlo úplne bez problémov. Vo vygenerovanom súbore sa vyskytovalo mnoho chýb, boli to hlavne nesprávne anotácie, ktoré bolo potrebné ručne opraviť. Vygenerovaný súbor bol rozdelený tak, že 80% dát bolo použitých na trénovanie modelu, 15% na validáciu a 5% ako syntetické testovacie dáta, aby bolo možné porovnať výkonnosť modelu na skutočných dátach s výkonom na umelých testovacích dátach.

Vývoj a trénovanie modelov

V práci boli použité a porovnané dva predtrénované, všeobecné modely pre slovenský jazyk: SlovakBERT a destilovaná verzia tohto modelu . V tomto texte označujeme destilovanú verziu ako DistilSlovakBERT. SlovakBERT je open-source predtrénovaný model slovenského jazyka, ktorý využíva maskované modelovanie jazyka (MLM, z angl. "Masked Language Modeling"). Bol natrénovaný na všeobecnom slovenskom webovom korpuse, ale dá sa ľahko prispôsobiť na riešenie nových úloh. DistilSlovakBERT je predtrénovaný model získaný z modelu SlovakBERT metódou nazývanou "destilácia znalostí", ktorá výrazne zmenšuje veľkosť modelu pri zachovaní (až 97%) jeho schopností porozumieť jazyku.

TABUĽKA 2

Počet parametrov v použitých NER modeloch a ich príslušné počty parametrov pre základný model a klasifikačnú vrstvu.

Model	Základný model	Klasifikačná vrstva	Spolu
SlovakBERT	124,054,272	6,921	124,061,193
DistilSlovakBERT	81,527,040	6,921	81,533,961

Oba modely boli upravené pridaním vrstvy klasifikácie, čím sa v oboch prípadoch získali modely vhodné pre úlohy NER. Posledná klasifikačná vrstva pozostáva z 9 neurónov zodpovedajúcich 9 anotáciám entít, t.j. 4 časti adresy a každá je reprezentovaná dvoma anotáciami - začiatok (B) a vnútro (I) každej entity a jedna anotácia je pre neprítomnosť akejkoľvek entity (O). Počet parametrov pre každý model a jeho zložky sú zhrnuté v tabuľke 2.

Trénovanie modelov sa ukázalo byť značne náchylné na preučenie. Na riešenie tohto problému a ďalšie zlepšenie procesu trénovania bolo použité lineárne zmenšovanie parametru rýchlosti učenia, regularizačná stratégiu "weight decay" a niektoré ďalšie stratégie ladenia hyperparametrov.

Na trénovanie modelov boli využité výpočtové prostriedky HPC systému Devana, ktorý prevádzkuje Výpočtové stredisko Centra Spoločných Činností SAV, konkrétne s využitím akcelerovaného uzla s 1 grafickou kartou (GPU) NVidia A100. Na pohodlnejšiu analýzu a ladenie bolo využívané interaktívne prostredie OpenOnDemand, ktoré umožňuje používateľom vzdialený webový prístup k superpočítaču.

Proces trénovania vyžadoval iba 10-20 epoch na natrénovanie pre oba modely. Pri použití spomenutých HPC prostriedkov bol čas trénovania jednej epochy v priemere 20 sekúnd pre 9492 vzoriek v trénovanom súbore dát pre SlovakBERT a 12 sekúnd pre DistilSlovakBERT. Inferencia na 69 vzorkách trvá 0,64 sekundy pre SlovakBERT a 0,37 sekundy pre DistilSlovakBERT, čo dokazuje dostatočnú efektivitu pre použitie týchto modelov v NLP aplikáciách v reálnom čase.

Podniky vynakladajú veľké množstvo úsilia a finančných prostriedkov na komunikáciu s klientmi.

Iteratívne vylepšenia

Hoci sme mali k dispozícii len 69 reálnych pozorovaní, ich komplexnosť bola pomerne náročná na simulovanie v generovaných dátach. Generovaný súbor dát bol vytvorený pomocou viacerých promptov, výsledkom čoho bolo 11,306 viet, ktoré pripomínali človekom generovaný text. Získanie finálneho riešenia pozostávalo z niekoľkých iterácií, pričom každú iteráciu možno rozdeliť na viaceré kroky: generovanie dát, trénovanie modelu, vizualizácia chýb predikcie na reálnych a umelých testovacích dátach a ich analýza. Týmto spôsobom boli identifikované vzory, ktoré model nedokázal rozpoznať. Na základe týchto poznatkov boli vygenerované nové dáta, ktoré sa riadili týmito novo-identifikovanými vzormi. Dáta dopĺňané v iteráciách boli generované pomocou promptov uvedených v tabuľke 3. Pomocou každého novorozšíreného

súboru dát boli natrénované oba modely, pričom presnosť modelu SlovakBERT vždy prevyšovala presnosť modelu DistilSlovakBERT. Preto bol ďalej využívaný ako základný model už iba SlovakBERT.

Iterácia	Prompt
1.	Ulica + Číslo domu + Obec + PSČ (+miešanie a vynechávanie)
2.	Obec + Ulica + Číslo domu + PSČ (+vynechávanie)
3.	Obec + Číslo domu + Ulica + PSČ (+vynechávanie)
4.	Obec + Číslo domu + PSČ
5.	Ulica + Obec + Číslo domu (verbálna forma) + PSČ (+miešanie)
6.	Obec + Číslo domu + PSČ (Obec spomenutá dvakrát; +miešanie)
7.	Všetky dáta duplikované a napísané len malými písmenami.

TABUĽKA 3

Iteratívny proces vytvárania dátového súboru. Každý prompt bol použitý dvakrát: najprv so šumom a potom bez šumu, t.j. s prirodzenými váhaniami ľudskej reči. Niekedy, ak je v tabuľke uvedené, prompt umožňoval zamiešať alebo vynechať niektoré časti adresy (entity).

Výsledky

Matica zámen (z angl. "Confusion Matrix") zodpovedajúca výsledkom modelu natrénovaného v Iterácii 1 (pozri Tabuľka 3) — je zobrazená v tabuľke 4. Tento model dokázal správne rozpoznať iba 67,51 % entít v testovacom súbore údajov. Podrobné preskúmanie chýb predikcie ukázalo, že súbor trénovacích dát nereprezentuje dostatočne dobre reálne pozorovania a je potrebné generovať viac reprezentatívnejších údajov. V tabuľke 4 je zrejmé, že najčastejšou chybou bola identifikácia obce ako ulice a dochádzalo k tomu v prípadoch, keď sa názov obce objavil pred názvom ulice v adrese. Výsledkom bolo generovanie dát pomocou iterácie 2 a iterácie 3.

PREDIKCIA

	O	B-Ulica	I-Ulica	B-ČísloDomu	I-ČísloDomu	B-Obec	I-Obec	B-PSČ	I-PSČ
O	53	6	10	1	1	2	0	0	0
B-Ulica	1	30	21	0	0	0	0	0	0
I-Ulica	0	1	10	0	0	0	0	0	0
B-ČísloDomu	2	1	0	69	0	0	0	0	0
I-ČísloDomu	0	0	0	1	18	0	0	0	0
B-Obec	6	37	3	0	0	25	0	0	0
I-Obec	1	0	9	0	0	0	8	0	0
B-PSČ	0	0	0	0	0	0	0	1	0
I-PSČ	0	0	0	0	0	0	0	0	0

TABUĽKA 4

Matica zámen modelu natrénovaného na súbore dát z prvej iterácie, ktorá dosiahla predikčnú presnosť modelu 67,51 %.

Cieľom bolo dosiahnuť viac ako 90% presnosť na reálnych testovacích dátach. Presnosť predikcie modelu sa so systematickým generovaním údajov neustále zvyšovala. Finálne bol celý súbor údajov zduplikovaný tak, že duplicity reflektovali text s použitím len malých písmen, nakoľko využitý predtrénovaný model je citlivý na malé a veľké písmená a niektoré testovacie pozorovania obsahovali názvy ulíc a obcí s malými písmenami. Vďaka tomu sa model stal robustnejším voči forme, v ktorej dostáva vstup, a dosiahol konečnú presnosť 93,06%. Matica zámen najlepšieho (finálneho) modelu je zobrazená v tabuľke 5.

TABUĽKA 5

Matica zámen konečného modelu s presnosťou 93,06%. Porovnaním výsledkov s výsledkami v Tabuľke 4 vidíme, že presnosť sa zvýšila o 25,55%.

PREDIKCIA

	O	B-Ulica	I-Ulica	B-ČísloDomu	I-ČísloDomu	B-Obec	I-Obec	B-PSČ	I-PSČ
O	61	1	1	0	0	5	4	1	0
B-Ulica	0	50	0	0	0	1	1	0	0
I-Ulica	0	0	10	0	0	0	1	0	0
B-ČísloDomu	0	0	0	72	0	0	0	0	0
I-ČísloDomu	0	0	0	0	19	0	0	0	0
B-Obec	1	3	0	0	0	66	1	0	0
I-Obec	0	0	1	0	0	1	16	0	0
B-PSČ	0	0	0	0	0	0	0	1	0
I-PSČ	0	0	0	0	0	0	0		0

TABUĽKA 6

Príklady predikcií konečného modelu pre dve testovacie vety. Prvá veta obsahuje jeden nesprávne klasifikovaný token: tretí token „Kal" s anotáciou *O* bol klasifikovaný ako *B-Obec*. K nesprávnej klasifikácii "Kal" ako obce došlo v dôsledku jeho podobnosti s podslovami nachádzajúcimi sa v slove "Kalša". Druhá veta má všetky svoje tokeny klasifikované správne.

V predikciách sa stále vyskytujú niektoré chyby; najmä tokeny, ktoré majú byť identifikované ako *O*, sú občas nesprávne klasifikované ako *Obec*. Týmto problémom sme sa ďalej nezaoberali, pretože sa vyskytuje pri slovách, ktoré sa môžu podobať na časti názvov entít, ale v skutočnosti nepredstavujú samotné entity. Príklad je zobrazený v Tabuľke 6.

Veta	Tokenizovaný text	Anotácie	Predikované tagy
Kalša to Kal sa	Kalša	B-Obec	B-Obec
	to	O	O
	Kal	O	B-Obec
	sa	O	O
Košice Hlavná 7	Košice	B-Obec	B-Obec
	Hlavná	B-Ulica	B-Ulica
	7	B-ČísloDomu	B-ČísloDomu

Záver

V tejto štúdii bol natrénovaný NER model postavený na predtrénovanom LLM modeli SlovakBERT. Model bol natrénovaný výlučne na umelo vygenerovanom súbore dát. Finálne syntetické trénovacie dáta boli reprezentatívne a kvalitné, vďaka ich iteratívnemu rozširovaniu. Spolu s dolaďovaním hyperparametrov tento iteratívny prístup umožňuje dosiahnuť predikčnú presnosť na reálnom dátovom súbore, presahujúcu 90%. Prezentovaný prístup naznačuje vysoký potenciál používania výlučne synteticky generovaných dát a to najmä v prípadoch, keď množstvo reálnych údajov nie je dostatočné na trénovanie.

Získaný model je možné využiť v reálnych aplikáciách slúžiacich na extrakciu a overenie správnosti adries, získaných mechanizmami prevodu reči na text. V prípade, že je k dispozícii väčší súbor reálnych dát, odporúčame model pretrénovať a prípadne aj rozšíriť syntetický súbor dát o ďalšie generované údaje, pretože existujúci súbor nemusí reprezentovať potenciálne nové vzory v týchto nových, reálnych dátach.

Model je dostupný na https://huggingface.co/nettle-ai/slovakbert-address-ner.

Spolu s dolaďovaním hyperparametrov tento iteratívny prístup umožňuje dosiahnuť predikčnú presnosť na reálnom dátovom súbore, presahujúcu 90 %.

LITERATÚRA

[1] Matúš Pikuliak, Štefan Grivalský, Martin Konôpka, Miroslav Blšťák, Martin Tamajka, Viktor Bachratý, Marián Šimko, Pavol Balážik, Michal Trnka, and Filip Uhlárik. Slovakbert: *Slovak masked language model.* CoRR, abs/2109.15254, 2021.

[2] Lance Ramshaw and Mitch Marcus. *Text chunking using transformation-based learning*. In Third Workshop on Very Large Corpora, 1995.

[3] Ministerstvo vnútra Slovenskej republiky. *Register adries.* https://data.gov.sk/dataset/register-adries-register-ulic. Accessed: August 21, 2023.

[4] Ivan Agarský. *Hugging face model hub.* https://huggingface.co/crabz/distil-slovakbert, 2022. Accessed: September 15, 2023.

Výsledný NER model predstavuje dôležitý prvok pre vývoj reálnych systémov starostlivosti o zákazníkov, ktorý sa dá využiť všade, kde je nutné extrahovanie adresy.

POĎAKOVANIE

Výskum bol realizovaný s podporou Národného kompetenčného centra pre HPC, projektu EuroCC 2 a Národného Superpočítačového Centra na základe dohody o grante 101101903-EuroCC 2-DIGITAL-EURO-HPC-JU-2022-NCC-01.

Výskum (alebo jeho časť) bol realizovaný s využitím výpočtovej infraštruktúry obstaranej v projekte Národné kompetenčné centrum pre vysokovýkonné počítanie (kód projektu: 311070AKF2) financovaného z Európskeho fondu regionálneho rozvoja, Štrukturálnych fondov EU Informatizácia spoločnosti, operačného programu Integrovaná infraštruktúra 2014-2020.

VYUŽITIE VEĽKÝCH JAZYKOVÝCH MODELOV NA EFEKTÍVNU ANALÝZU NÁBOŽENSKÝCH TEXTOV

BIBIÁNA LAJČINOVÁ
JOZEF ŽUFFA
MILAN URBANČOK

Štúdia sa zaoberá úlohou získavania informácií (z angl. *information retrieval*) z dokumentov s náboženskými témami v slovenskom jazyku pomocou embeddingových modelov, s cieľom zrýchlenia a zefektívnenia analýzy týchto textov pre odborníkov v danej oblasti. Pomocou verejne dostupných embeddingových modelov *Slovak-BERT* a *BGE M3* a proprietárneho modelu *text-embedding-3-small* od OpenAI sme generovali embeddingové indexy z textových blokov dostupných dát a vyhodnocovali metriku recall naprieč piatimi rôznymi témami pomocou testovacích otázok. Skúmali sme tiež rôzne techniky predspracovania, ako je kontextová augmentácia testovacích otázok a odstraňovanie vyradených slov (z angl. *stopwords*). Výsledky naznačujú, že táto metodológia môže byť užitočná pre zefektívnenie výskumu náboženských textov a môže pomôcť odhaliť skryté interpretácie a významy obsiahnuté v týchto textoch. Naše zistenia tiež zdôrazňujú dôležitosť výberu vhodnej techniky predspracovania pre daný model a dáta.

Analýza a štúdium textov s náboženskými témami boli historicky doménou filozofov, teológov a ďalších špecialistov v sociálnych vedách. S príchodom umelej inteligencie, konkrétne veľkých jazykových modelov, naberá výskum v tejto oblasti nové dimenzie. Tieto moderné technológie môžu byť využité na odhaľovanie skrytých nuáns v náboženských textoch, čím umožňujú hlbšie pochopenie rôznych symbolizmov a odhalenie významov, ktoré sú pre tieto texty charakteristické a môžu byť nejasné. Takéto zrýchlenie analytického procesu umožňuje výskumníkom sústrediť sa len na špecifické aspekty textu relevantné pre ich výskum. Jednou z úloh, ktorou sa vedci v tejto oblasti zaoberajú, je štúdium diel autorov asociovaných so špecifickými náboženskými skupinami a komunitami. Porovnávaním ich textov s oficiálnymi doktrínami a učeniami ich denominácií môžu výskumníci hlbšie nahliadnuť do presvedčení, viery a uhlov pohľadu komunít, formovaných učeniami vplyvných autorov.

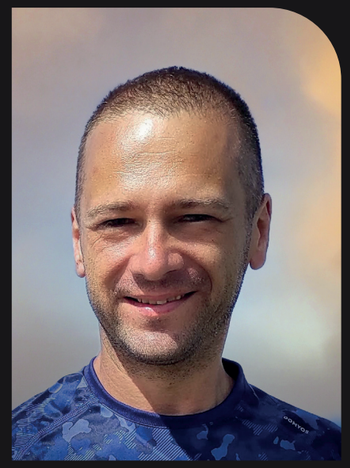
Štúdia sumarizuje prístup využívajúci embeddingové indexy a jazykové modely na efektívnu analýzu textov s náboženskými témami. Primárnym cieľom je vyvinúť nástroj na získavanie informácií, špeciálne navrhnutý na účinné lokalizovanie relevantných častí textu v dokumentoch. Identifikácia nesúladov medzi získanými časťami textu z diel náboženských komunít a oficiálnymi náukami daného náboženstva, z ktorého táto komunita pochádza, nie je cieľom tejto práce a je ponechaná na teológov.

Táto práca vznikla spojeným úsilím Národného superpočítačového centra a Teologickej fakulty Trnavskej univerzity. Na dosiahnutie riešenia, ktoré vyžadovalo numericky náročné spracovanie veľkého objemu dát, boli využité výpočtové zdroje HPC systému Devana.

Texty analyzované v tejto štúdii pochádzajú z náboženskej komunity známej ako *Hnutie Nazaret* (bežne nazývanej aj "Beňovci"), ktorá sa začala formovať v sedemdesiatych rokoch minulého storočia. Hnutie, o ktorom niektorí vedci hovoria, že má známky sekty, je stále aktívne aj v dnešnej dobe, avšak v redukovanej a zmenenej forme. Jeho zakladateľ, Ján Augustín Beňo (1921 – 2006), bol tajne vysväteným katolíckym kňazom v totalitnej dobe. Beňo nabádal členov hnutia k aktívnemu žitiu viery prostredníctvom každodenného čítania biblických textov a uplatňovania ich posolstva v pra-

Analýza a štúdium textov s náboženskými témami boli historicky doménou filozofov, teológov a ďalších špecialistov v sociálnych vedách.

ThDr. Ing. Milan Urbančok SDB, PhD. a doc. Jozef Žuffa, PhD. pôsobia ako vedeckí pracovníci a pedagógovia na Teologickej fakulte TU v Trnave. Milan Urbančok (vľavo) je vedúcim Katedry systematickej teológie. Jozef Žuffa je prodekanom pre rozvoj a pôsobí na Katedre praktickej teológie.



xi prostredníctvom konkrétnych rozhodnutí a činov. Hnutie sa rozšírilo po celom Slovensku, pričom komunity existovali takmer v každom väčšom meste. Rozšírilo sa aj do susedných krajín, ako Poľsko, Česká republika, Ukrajina a Maďarsko. V roku 2000 bolo v hnutí približne tristo manželských párov, tisíc deti a stotridsať kňazov a študentov pripravujúcich sa na kňazstvo. Hnutie malo tri hlavné ciele: radikálnu prevenciu v oblasti vzdelania, podporu kňazov, ktorí by mohli pôsobiť ako rodičovské postavy na identifikáciu a rozvoj kňazských povolaní u detí, a výrobu a distribúciu samizdatových materiálov potrebných na katechézu a evanjelizáciu.

Pre výskum bolo k dispozícii 27 dokumentov vytvorených tou-to komunitou. Tieto dokumenty, ktoré významne vplývali na formovanie ideológie komunity Beňovci, boli reprodukované a distribuované počas komunistického režimu vo forme samizdatov – literatúry zakázanej komunistickým režimom. Po politickom prevrate boli viaceré z týchto dokumentov vytlačené a distribuované verejnosti mimo hnutia. Väčšina z dokumentov pozostávala z textov určených pre "ranné úvahy" – krátke meditácie nad biblickými textami. Dokumenty taktiež obsahovali zakladateľove komentáre o učeniach Katolíckej cirkvi a vybraných témach týkajúcich sa výchovy detí, spirituálneho vedenia a katechézy pre deti.

Dokumenty obsahovali niekoľko duplicít, avšak pre úlohu získavania a vyhľadávania informácií to nepredstavuje problém. Všetky dokumenty sú napísané výhradne v slovenskom jazyku.

Jeden z dokumentov bol anotovaný pre účely testovania expertom z partnerskej fakulty, ktorý sa dlhodobo venuje Hnutiu Nazaret. Anotáciami myslíme časti textu (zvyčajne odseky, prípadne vety) označené ako patriace do jednej z piatich tried, pričom tieto triedy reprezentujú päť tém:

1. Direktívna poslušnosť
2. Hierarchická výchova
3. Radikálnosť v prevzatí modelu života
4. Ľudské potreby realizované len v spoločenstve/hnutí a v rodine
5. Divné/čudné/silné

Každá z týchto tém je doplnená o súbor otázok (dopytov/výrazov), ktoré boli navrhnuté na testovanie riešenia získavania informácií. Cieľom týchto testovacích otázok je vyhodnotiť, koľko relevantných častí textu týkajúcich sa danej témy dokáže náš nástroj získať z anotovaného dokumentu.

TABUĽKA 1

Príklad anotovaného textu patriaceho do triedy Direktívna poslušnosť.

Text	Anotácia
Veď ak milujeme svojho Boha, ako si to myslíme, alebo aj hovoríme, nemôže nám byť ľahostajný nijaký odklon od jeho svätej vôle.	Direktívna poslušnosť

Postup riešenia

Existuje viacero metód vhodných na riešenie tejto úlohy, vrátane klasifikácie textu, modelovania témy textu, RAG (z angl. *Retrieval-Augmented Generation*), alebo optimalizácie predtrénovaného jazykového modelu. Avšak, požiadavkou partnerských teológov, zaoberajúci sa analýzou týchto dokumentov, bola identifikácia konkrétnych častí textu relevantných k daným témam, a teda získanie ich presného znenia. Práve preto bola vybraná metóda získavania informácií (z angl. *information retrieval*). Tento prístup sa líši od metódy RAG, ktorá bežne obsahuje časť získavania informácií a tiež časť generovania nového textu, v tom, že sa sústredí výhradne na identifikáciu relevantných častí textu v dokumentoch a negeneruje žiadny nový text.

Metóda získavania informácií využíva jazykové modely na transformovanie komplexných dát, ako je text, do numerickej repre-

Keďže všetky analyzované dokumenty v rámci tejto štúdie sú v slovenskom jazyku, je potrebné, aby zvolený jazykový model "rozumel" slovenčine, čo značne zúžilo možnosti jeho výberu.

zentácie, ktorá zachytáva celý význam a kontext daného vstupu. Táto numerická reprezentácia, nazývaná embedding (vo zvyšku textu budeme kvôli jednoduchosti využívať už len tento termín), môže byť použitá na sémantické vyhľadávanie v dokumentoch analyzovaním pozícií a blízkosti embeddingov v multidimenzionálnom vektorovom priestore. Použitím otázok (dopytov) dokáže systém nájsť v dokumentoch relevantné časti textu meraním podobnosti medzi embeddingami otázok a embeddingami segmentovaného textu. Tento prístup nevyžaduje žiadnu optimalizáciu existujúceho jazykového modelu, takže modely môžu byť použité bez akýchkoľvek úprav a pracovaný postup zostáva pomerne jednoduchý.

Výber modelu

Keďže všetky analyzované dokumenty v rámci tejto štúdie sú v slovenskom jazyku, je potrebné, aby zvolený jazykový model "rozumel" slovenčine, čo značne zúžilo možnosti jeho výberu. K dnešnému dňu existuje len jeden verejne dostupný model, ktorý rozumie výhradne slovenskému jazyku, a niekoľko multilingválnych modelov, ktoré rozumejú slovenčine do určitej miery. Štyri predtrénované modely boli vybrané z malého množstva dostupných možností, prvým z nich je model *Slovak-BERT* [1]. *Slovak-BERT* je verejne dostupný model založený na architektúre transformerov. Ďalším vybraným modelom je *text-embedding-3-small* model. Ide o výkonný proprietárny embedding model dostupným len cez API spoločnosti OpenAI. Tretím modelom je verejne dostupný embedding model *BGE M3* [2], ktorý je výkonným multilingválnym modelom podporujúcim viac než 100 jazykov. Posledným modelom je taktiež multilingválny model z dielne Microsoftu nazývaný *E5* [3], ktorý je rovnako verejne dostupný.

Tieto štyri modely boli použité na získanie vektorových reprezentácií textu. Ich výkon bude detailne diskutovaný v nasledujúcich častiach reportu.

Predspracovanie dát

Prvým krokom predspracovania dát je segmentovanie textu (z angl. *chunking*). Hlavným dôvodom pre tento krok bolo splniť požiadavku teológov na vyhľadávanie (získavanie) krátkych častí textu. Okrem toho bolo potrebné dokumenty rozdeliť na menšie časti, aj kvôli obmedzenej dĺžke vstupu niektorých jazykových modelov. Na túto úlohu bola použitá knižnica Langchain [4]. Poskytuje hierarchické segmentovanie textu, ktoré produkuje prekrývajúce sa bloky textu definovanej dĺžky (s definovaným prekrytím) tak, aby v nich bol zachovaný kontext. Takto boli vytvorené bloky s dĺžkami 300, 400, 500 a 700 znakov. Následne spracovanie pozostávalo z odstránenia diakritiky, úprava textu na veľké/malé písmená, podľa podmienok modelov a odstránenie vylúčených slov (z angl. *stopwords*). Odstraňovanie týchto slov je bežnou praxou v úlohách spracovania prirodzeného jazyka, keďže vylúčené slová nenesú žiadnu významovú informáciu. Niektoré modely môžu profitovať z odstránenia vylúčených slov na zlepšenie relevantnosti získaných blokov textu, ale iné môžu ťažiť z ponechania týchto slov, aby bol zachovaný celý kontext nevyhnutný na pochopenie textu.

TABUĽKA 2

Príklad dvoch blokov textu s prekrytím.

Index	Anotácia
8	Podľa tohoročnej sa rozvádza už každé tretie. Tento bolestný spoločenský jav vysvetľujú niektorí skutočnosťou, že dnešní manželia sú náročnejší a od svojho manželstva viac očakávajú než tí, čo žili pred nami. Že by to bola pravda? Od manželstva môže človek čakať, len toľko, koľko doň vloží.
9	Že by to bola pravda? Od manželstva môže človek čakať, len toľko, koľko doň vloží. Ak minulosť týmto bláznovstvom rozvodovosti netrpela, tak zaiste preto, že v manželstve nevidela iba tú sentimentálnu príjemnú lásku, ale aj tú obetavú, živú z viery v Boha a z poslušnosti voči Cirkvi. Zaujímajú nás začiatky, priebeh a dôsledky rozvodov? Nie je ťažko spoznať ich.

Vektorové embeddingy

Vektorové embeddingy boli vytvorené z blokov textu s použitím vybraných predtrénovaných jazykových modelov.

V prípade modelu *Slovak-BERT*, sme pre generovanie embeddingov použili model bez pridaných predikčných vrstiev, a následne sme ukladali iba prvý embedding, ktorý obsahuje celý význam vstupného textu. Ďalšie používané modely priamo produkujú embeddingy vo vhodnej forme, preto nebolo potrebné žiadne dodatočné spracovanie výstupov.

V nasledujúcej časti s výsledkami analyzujeme výkon všetkých vybraných embedding modelov a porovnávame ich schopnosti zachytiť kontext.

Výsledky

Pred uskutočnením kvantitatívnych testov prešli všetky embeddingové indexy predbežným hodnotením, aby sa zistila úroveň porozumenia slovenského jazyka a špecifickej náboženskej terminológie evaluovaných modelov. Predbežné hodnotenie zahŕňalo subjektívne posúdenie relevantnosti získaných častí textu.

Tieto testy odhalili, že embeddingy získané pomocou modelu *E5* nie sú dostatočne efektívne pre naše dáta. Keď sme pomocou testovacej otázky hľadali informácie v dokumentoch, väčšina získaných blokov textu obsahovala kľúčové slová použité v otázke, ale neobsahovala kontext otázky. Možným vysvetlením by mohlo byť, že tento model uprednostňuje zhody na úrovni slov pred zhodami kontextu v slovenskom jazyku. Ďalším dôvodom môže byť aj to, že tento model bol natrénovaný na dátach, ktoré neobsahovali veľké množstvo textu v slovenčine, resp. výber textov nebol dostatočne rozmanitý, čo môže viesť k nižšiemu výkonu modelu *E5* v slovenčine, aj keď v iných jazykoch dosahuje výborné výsledky. Podotýkame, že tieto pozorovania nie sú definitívne závery, ale skôr hypotézy založené na súčasných, obmedzených výsledkoch. Rozhodli sme sa ďalej nevyhodnocovať výkon embeddingových indexov získaných z *E5* modelu, keďže je to irelevantné vzhľadom na neschopnosť modelu zachytiť nuansy náboženského textu. Na druhej strane, schopnosti modelu *Slovak-BERT*, ktorý je založený na architektúre RoBERTa charakteristickej jej relatívne jednoduchou topológiou, prekonali očakávania. Navyše, výkon *text-embedding-3-small* a *BGE M3* embeddingov splnil očakávania, keďže prvý, subjektívne vyhodnotený, test ukázal veľmi dobré porozumenie kontextu a nuáns v textoch s náboženskými témami a taktiež výborné porozumenie slovenského jazyka.

Preto boli kvantitatívne testy vykonané len pre vektorové databázy využívajúce *Slovak-BERT*, OpenAI *text-embedding-3-small* a *BGE M3* embeddingy.

Vzhľadom na povahu riešeného problému a charakter testovacích anotácií existuje potenciálna obava týkajúca sa ich kvality. Niektoré časti textu mohli byť nesprávne klasifikované, pretože môžu patriť do viacerých tried. Táto skutočnosť, spolu s možnosťou ľudskej chyby, mohla ovplyvniť konzistentnosť a presnosť anotácií.

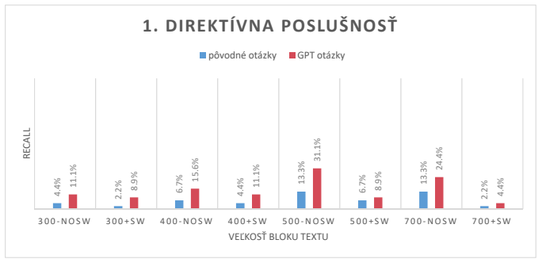
Berúc do úvahy túto skutočnosť, sme sa rozhodli zamerať výhradne na vyhodnotenie metriky zvanej *recall*. Hodnotu tejto metriky meriame ako pomer počtu získaných blokov zhodných s anotáciami, k celkovému počtu anotovaných blokov textu (bez ohľadu na podiel falošne pozitívnych blokov). Recall vyhodnocujeme pre každú tému a pre všetky vektorové databázy s rôznymi dĺžkami blokov textu.

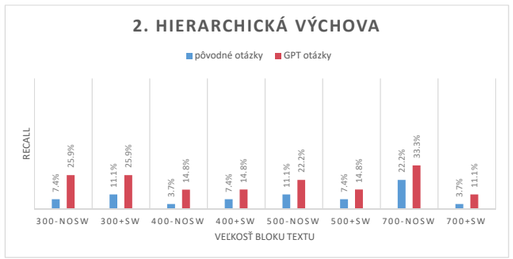
Komplexnosť a interpretačná povaha náboženských štúdií sa pravdepodobne prejavuje nielen v kvalite testovacích anotácií, ale aj v samotných testovacích otázkach. Ako príklad môžeme uviesť testovaciu otázku "Božia vôľa" pre tému *Direktívna poslušnosť*. Hoci pozorný čitateľ rozumie, ako táto otázka súvisí s danou témou, nemusí to byť očividné pre jazykový model. Preto, okrem vyhodnotenia pomocou dodaných testovacích otázok budeme vyhodnocovať výkon embeddingov aj s použitím ďalších otázok, ktoré boli získané metódou kontextovej augmentácie. Kontextová augmentácia je technika v prompt inžinieringu používaná na zlepšenie kvality textových dát a je dokumentovaná vo viacerých vedeckých článkoch [5], [6]. Táto technika spočíva v tom, že sa zvolený jazykový model použije na vytvorenie novej otázky (príp. nového textu) na základe pôvodnej otázky (textu) a doplneného kontextu s cieľom formulovania lepšej otázky. Jazykový model použitý na generovanie nových otázok pomocou tejto techniky bol *GPT 3.5* a tieto otázky budeme ďalej v texte označovať ako "GPT otázky".

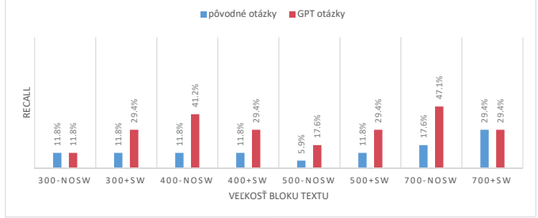
Slovak-BERT embeddingové indexy

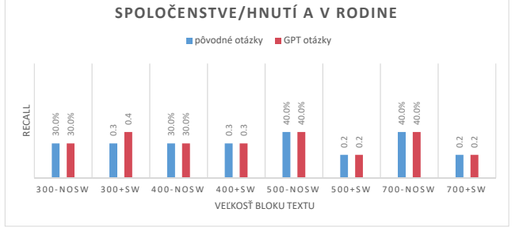
Vyhodnotenie metriky recall pre embeddingové indexy využívajúce *Slovak-BERT* embeddingy pre štyri rôzne veľkosti blokov textu s použitím a bez použitia metódy odstraňovania vylúčených slov je zobrazené na Obrázku 1. Toto vyhodnotenie zahŕňa každú z piatich tém špecifikovaných v úvode a pokrýva pôvodné aj GPT otázky.

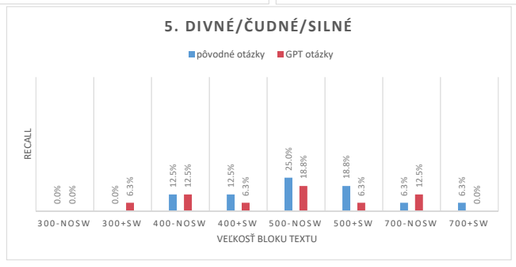
Je očividné, že GPT otázky produkujú vo všeobecnosti lepšie výsledky než pôvodné otázky, okrem prípadu posledných dvoch tém, pri ktorých obe sady otázok produkujú podobné výsledky. Je tiež zrejmé, že *Slovak-BERT* embeddingy vo väčšine prípadov profitujú z odstránenia vylúčených slov. Najvyššia hodnota recall bola dosiahnutá pre tému *Radikálnosť v prevzatí modelu života*, s veľkosťou blokov textu 700 znakov, s odstránenými vylúčenými slovami, dosahujúc viac než 47 %. Na druhej strane, najhoršie výsledky boli získané pre tému *Divné/čudné/silné*, kde ani jedna sada otázok nedokázala úspešne získať relevantné časti textu z dokumentov. Dokonca, v niektorých prípadoch neboli získane absolútne žiadne relevantné bloky textov.











OBRÁZOK 1

Hodnoty recall pre všetky témy vyhodnotené pomocou pôvodných aj GPT otázok (pre všetky skúmané veľkosti blokov textu) pre embeddingy generované modelom *Slovak-BERT*. Indexy embeddingov označené ako +SW obsahujú vylúčené slová, zatiaľ čo -NoSW znamená, že vylúčené slová boli odstránené.

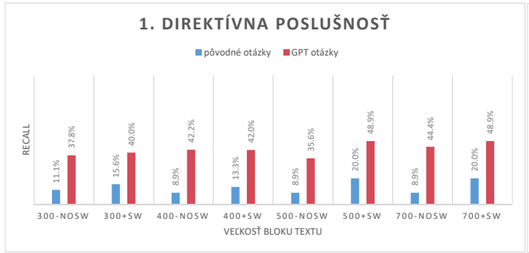
OpenAI text-embedding-3-small embeddingové indexy

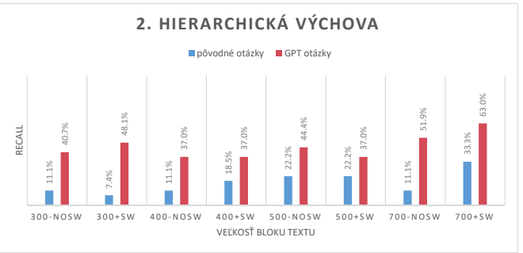
Analogicky k vyhodnoteniu *Slovak-BERT* embeddingových indexov, grafy s výsledkami pre embeddingy získané modelom *text-embedding-3-small* sú zobrazené na Obrázku 2. Hodnoty metriky recall sú všeobecne vyššie než tie získané so *Slovak-BERT* embeddingami. Podobne ako v predchádzajúcom prípade, GPT otázky produkujú lepšie výsledky. Pozorovateľný je taktiež istý trend medzi hodnotou metriky recall a veľkosťou textových blokov – dlhšie bloky textu zvyčajne vykazujú vyššie hodnoty recall.

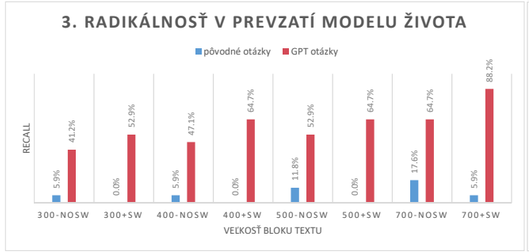
Zaujímavé zistenie sa týka témy *Radikálnosť v prevzatí modelu života*. S použitím pôvodných otázok sme nezískali takmer žiadne relevantné výsledky. Naopak, pri použití otázok generovaných pomocou GPT modelu, boli hodnoty recall metriky výrazne vyššie a dosahovali takmer 90 % pre bloky textu s veľkosťou 700 znakov.

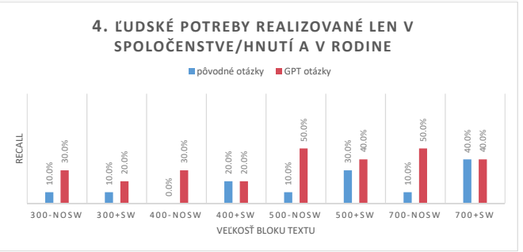
Čo sa týka odstraňovania vyradených slov, vplyv tejto techniky na embeddingy sa líši. Pre témy 4 a 5 sa ukazuje, že odstránenie vyradených slov je prospešné. Avšak, pre ostatné témy tento krok výhody neprináša.

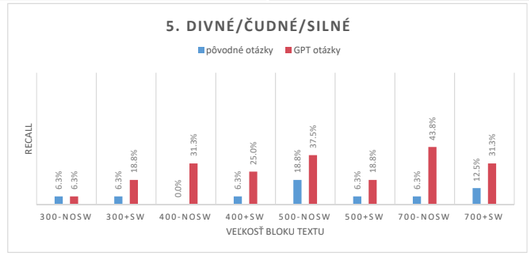
Témy 4 a 5 vykazovali najslabšie výsledky medzi všetkými témami. Môže to byť spôsobené povahou otázok pre tieto dve témy, keďže sú to citáty a celé vety, na rozdiel od otázok pre ostatné témy, ktoré sú frázy, kľúčové slová alebo výrazy. Zdá sa, že model *text-embedding-3-small* funguje lepšie s frázovitým typom otázok. Ale na druhej strane, keďže otázky pre témy 4 a 5 sú celé vety, zdá sa embeddingy profitujú z odstránenia vyradených slov, keďže v tomto prípade to môže pomôcť pri zachytení kontextu v dlhých otázkach.









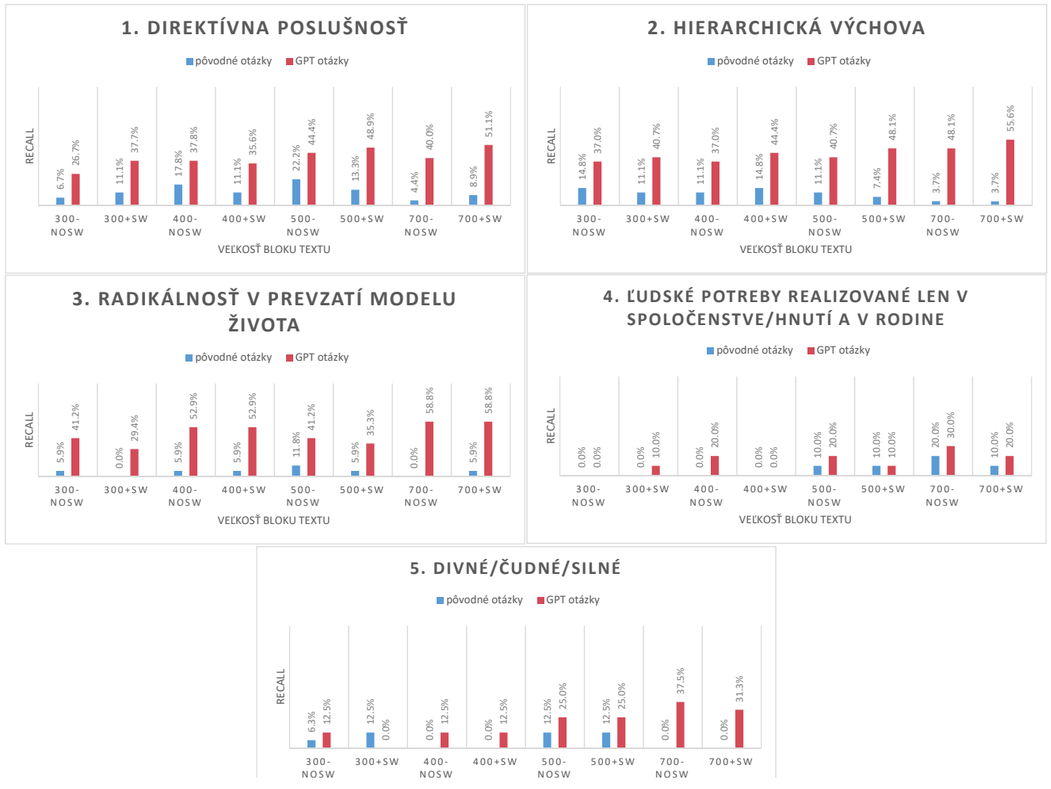


OBRÁZOK 2

Hodnoty recall vyhodnotené pre všetky témy pomocou pôvodných aj GPT otázok, pre všetky embeddingové indexy generované modelom *text-embedding-3-small*. Embeddingové indexy označené +SW obsahujú vylúčené slová, zatiaľ čo indexy označené -NoSW majú vylúčené slová odstránené.

OBRÁZOK 3

Hodnoty metriky recall pre všetky témy získané s použitím pôvodných aj GPT otázok pre embeddingy vytovrené modelom *BGE M3*. Značky +SW označujú indexy obsahujúce vylúčené slová, zatiaľčo -NoSW indikuje, že vylúčené slová boli v daných indexoch odstránené.



Téma 4 je veľmi špecifická a preto možno vyžaduje detailnejšie testovacie otázky, keďže poskytnuté otázky pravdepodobne neobsahujú všetky nuansy danej témy. Naopak, téma 5 je veľmi všeobecná, vďaka čomu je celkom pochopiteľné, prečo je zachytávanie kontextu tejto témy pomocou embeddingov náročné. Všeobecný charakter tejto témy by mohol profitovať z iného analytického prístupu. Napríklad metóda analýzy sentimentu by mohla zachytiť zvláštnu, čudnú a silnú náladu vo vzťahu k študovaným náboženským témam.

BGE M3 embeddingové indexy

Grafy s vyhodnotenou metrikou recall pre embeddingové indexy využívajúce *BGE M3* embeddingy sú zobrazené na Obrázku 3. Tieto hodnoty ukazujú výkon spadajúci medzi *Slovak-BERT* a OpenAI *text-embedding-3-small* embeddingy. V niektorých prípadoch sa nepodarilo dosiahnuť také vysoké hodnoty metriky recall ako pri OpenAI embeddingoch, avšak *BGE M3* embeddingy stále vykazujú konkurencieschopný výkon, hlavne ak prihliadneme na skutočnosť, že sa jedná o verejne dostupný model, na rozdiel od OpenAI embeddingového modelu, ku ktorému sa dá pristupovať len cez API, čo môže byť niekedy problémom kvôli zdieľaniu súkromných alebo citlivých dát a taktiež kvôli finančným nákladom.

S týmito embeddingami môžeme pozorovať rovnaký fenomén ako s *text-embedding-3-small* embeddingami: krátke, frázovité otázky sú preferované pred dlhšími otázkami podávanými formou viet a citátov. Preto sú hodnoty recall pre prvé tri témy vyššie, ako sme diskutovali už v predchádzajúcej časti.

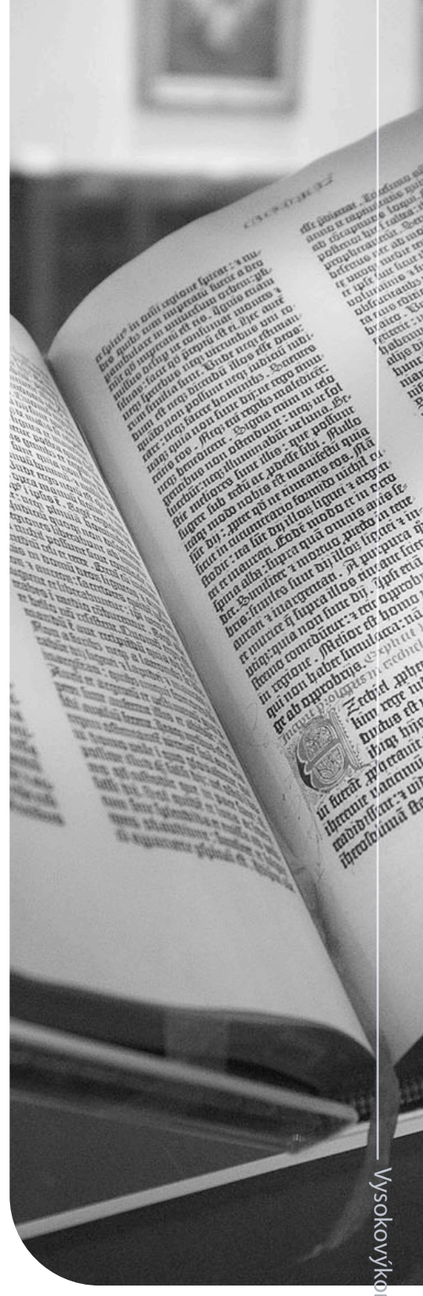
Odstránenie vylúčených slov sa zdá byť užitočné, hlavne pre posledné dve témy.

Záver

Štúdia prezentuje prístup pre analýzu textov s náboženskými témami pomocou numerických reprezentácií textu zvaných embeddingy, generovanými troma vybranými predtrénovanými jazykovými modelmi: *Slovak-BERT*, OpenAI *text-embedding-3-small* a *BGE M3* model. Výberu modelov predchádzalo posúdenie ich schopnosti "rozumieť slovenčine" a náboženskej terminológii. Pre zvolené tri modely sme konštatovali dostatočnú schopnosť, čo ich predurčilo ako vhodných kandidátov na zvládnutie úlohy získavania informácií z danej sady dokumentov.

Výzvy týkajúce sa kvality testovacích otázok boli adresované pomocou techniky kontextovej augmentácie. Tento prístup pomohol pri formulovaní vhodnejších otázok, čo viedlo k získavaniu relevantnejších častí textu, ktoré zachytávali všetky nuansy tém, ktoré teológovia v texte hľadajú.

Výsledky demonštrujú, že efektívnosť embeddingov generovaných týmito modelmi, hlavne modelom *text-embedding-3-small* od OpenAI, je dostatočná na hlboké porozumenie kontextu, aj v slovenskom jazyku. Hodnoty metriky recall pre embeddingy tohto modelu sa líšia v závislosti od témy a použitých testovacích otázok, pričom najlepšia hodnota bola dosiahnutá pre tému *Radikálnosť v prevzatí modelu života* dosahujúc takmer 90 %, s použitím GPT otázok a dĺžky textových blokov 700 znakov. Vo všeobecnosti, *text-embedding-3-small* model mal najlepšie výsledky s najväčšou analyzovanou dĺž-



Zistenia zdôrazňujú potenciál využitia veľkých jazykových modelov pre špecializované oblasti ako analýza textu s náboženskými témami.

kou blokov textu, vykazujúc mierny trend zvyšujúcej sa hodnoty recall so zväčšujúcou sa dĺžkou blokov textu. Téma *Divné/čudné/silné* mala najnižšiu hodnotu recall, čo môže byť dôsledkom neurčitosti v špecifikácii tejto témy.

Pre *Slovak-BERT* embeddingové indexy sú hodnoty recall o niečo nižšie, ale stále pomerne pôsobivé vzhľadom na jednoduchosť tohto jazykového modelu. Lepšie výsledky boli získané v použitím GPT otázok, s najlepšou hodnotou 47.1 % pre tému *Radikálnosť v prevzatí modelu života* s dĺžkou blokov 700 znakov, a s embeddingami vytvorenými z textu s odstránenými vylúčenými slovami. Celkovo, tento model najviac ťažil z odstraňovania vylúčených slov.

Čo sa týka *BGE M3* embeddingov, výsledky boli taktiež veľmi dobré, dosahujúc vysokú hodnotu recall metriky, aj keď nie až takú vysokú ako v prípade OpenAI embeddingov. Ale vzhľadom na to, že *BGE M3* je verejne dostupný model, sú tieto výsledky pozoruhodné.

Tieto zistenia zdôrazňujú potenciál využitia veľkých jazykových modelov pre špecializované oblasti ako analýza textu s náboženskými témami. Výskum by sa ďalej mohol zaoberať zhlukovaním embeddingov za účelom odhalenia asociácií a inšpirácií autorov týchto diel. Pre teológov, budúca práca spočíva v analýze získaných častí textu s cieľom identifikovať odchýlky od oficiálneho učenia Katolíckej cirkvi, čím sa objasnia interpretácie a pohľady hnutia.

POĎAKOVANIE

Výskum bol realizovaný s podporou Národného kompetenčného centra pre HPC, projektu EuroCC 2 a Národného Superpočítačového Centra na základe dohody o grante 101101903-EuroCC 2-DIGITAL-EURO-HPC-JU-2022-NCC-01.

Výskum bol realizovaný s využitím výpočtovej infraštruktúry obstaranej v projekte Národné kompetenčné centrum pre vysokovýkonné počítanie (kód projektu: 311070AKF2) financovaného z Európskeho fondu regionálneho rozvoja, Štrukturálnych fondov EU Informatizácia spoločnosti, operačného programu Integrovaná infraštruktúra 2014 – 2020.

LITERATÚRA

[1] Matúš Pikuliak, Štefan Grivalský, Martin Konôpka, Miroslav Blšták, Martin Tamajka, Viktor Bachratý, Marián Šimko, Pavol Balážik, Michal Trnka, and Filip Uhlárik. Slovakbert: Slovak masked language model, 2021.

[2] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

[3] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multil-ingual e5 text embeddings: A technical report, 2024.

[4] Harrison Chase. Langchain. https://github.com/langchain-ai/langchain, 2022. Accessed: May 2024.

[5] Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting for retrieval-augmented large language models, 2023.

[6] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. Query expansion by prompting large language models, 2023.

03

Popularizácia HPC

Krátke správy

EuroCC 2 All Hands Meeting
vo Vysokých Tatrách

V nádhernom prostredí Vysokých Tatier sa členovia Európskych národných kompetenčných centier a Centier excelentnosti stretli na konferencii EuroCC 2 & CoEs & CASTIEL 2 Intermediate Conference. Podujatie sa konalo od 22. do 24. apríla a jeho cieľom bolo diskutovať o pokroku v projekte a zdieľať dosiahnuté úspechy.

Prvý deň otvoril tím projektového manažmentu spolu s lokálnymi organizátormi zo Slovenského kompetenčného centra pre HPC. Účastníci diskutovali o nedávnych vývojoch a výsledkoch prvého hodnotenia projektu. Dôležitou časťou boli aj rozhovory o synergiách medzi kompetenčnými centrami a centrami excelentnosti, ktoré vytvorili základy pre spoločné aktivity a výmenu skúseností.

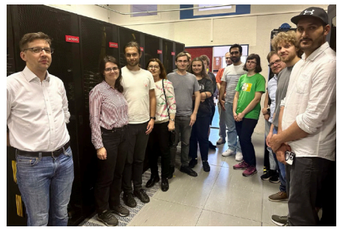
Druhý deň bol venovaný praktickým otázkam financovania a strategického plánovania. Workshopy sa sústredili na rozpočtové pravidlá a diskusie o kompetenciách NCC a CoE poskytli cenné poznatky potrebné pre udržateľný rast. Účastníci tiež mali možnosť zúčastniť sa exkurzie okolo jazera Štrbské pleso, nadväzovať kontakty s novými kolegami a v príjemnej atmosfére neformálne diskutovať o možnostiach spolupráce.

Regionálne stretnutie národných kompetenčných
centier pre HPC v Strednej Európe

Dňa 10. júna sa v Grundlsee konalo tretie stretnutie stredoeurópskej pracovnej skupiny kompetenčných centier pre HPC. Podujatie zorganizovalo NCC Rakúsko a zúčastnili sa ho zástupcovia z Poľska, Rakúska, Chorvátska, Českej republiky, Slovenska, Slovinska a Maďarska.

Úvodné slovo mal vedúci rakúskeho NCC pre HPC, Markus Stöhr, po ktorom nasledovali prezentácie jednotlivých centier. Hlavnou témou stretnutia bola spolupráca medzi jednotlivými NCC a navonok. Workshop o efektívnej komunikácii a prezentácii aktivít prostredníctvom firemných profilov na LinkedIn viedla odborníčka na obsahový marketing Natascha Trzepizur. Okrem toho účastníci diskutovali na tému zlepšenia organizácie a koordinácie tréningových programov. Posledná sekcia bola venovaná umelej inteligencii.

[redacted] eurocc.nscc.sk



NCC Slovensko & NCC Poľsko
ORCA Hands-on Workshop

V dňoch 21. a 22. mája 2024 sa v Bratislave uskutočnil ORCA hands-on workshop, organizovaný národnými kompetenčnými centrami pre HPC zo Slovenska a Poľska. Workshop bol zameraný na softvérový kvantovo-chemický balík ORCA a jeho praktické využitie.

Účastníci sa oboznámili so základmi a pokročilými technikami práce s ORCA. Lektor Klemens Noga z Cyfronetu predstavil poľský HPC ekosystém a možnosti pre používateľov v oblasti výpočtovej chémie. Súčasťou programu bola aj prehliadka superpočítača Devana vo Výpočtovom stredisku SAV. Workshop

poskytol účastníkom cenné znalosti pre prácu s ORCA v oblasti kvantovo-chemických výpočtov na HPC systémoch.

Možnosti využitia HPC pre malé a stredné podniky

Dňa 19. marca sa v Bratislave konalo podujatie "Objavte potenciál superpočítača v praxi", ktoré spojilo odborníkov z Národného kompetenčného centra HPC a Slovenskej obchodnej a priemyselnej komory. Cieľom bolo predstaviť firmám výhody a možnosti využitia vysokovýkonného počítania v podnikaní.

Účastníci mali príležitosť dozvedieť sa o moderných technológiách a ich aplikáciách vo vývoji a výrobnej technológii. Lucia Demovičová a Michal Pitoňák predstavili služby NCC pre HPC a diskutovali o možnostiach budúcej spolupráce. Firmy sa mohli oboznámiť s testovacími alokáciami HPC zdrojov a zapojením sa do bezplatných pilotných projektov, čo predstavuje atraktívnu príležitosť adopcie HPC bez veľkých investícií. Súčasťou podujatia bola aj prehliadka superpočítača Devana, ktorá poskytla unikátny pohľad do sveta vysokovýkonných výpočtov.

Popularizačné prednášky

Tím Národného kompetenčného centra pre HPC organizoval sériu popularizačných prednášok zameraných na aktuálne témy v oblasti vedy a technológie. Prednášky sa zaoberali rôznymi zaujímavými témami, ako je úloha vysoko výkonných počítačov a grafických procesorov v genómovej analýze DNA, konformačné preferencie prirodzene neusporiadaného proteínu tau, zvedavá umelá inteligencia, a atomistický pohľad na biopolyméry. Ďalšie témy zahŕňali moderné technológie a výzvy v automatickom spracovaní reči, modelovanie geovedných dát pre spoznanie geodynamických procesov Zeme, kvantové počítanie a jeho praktické aplikácie, a kvantovú zložitosť a simulácie. Prednášky poskytli účastníkom cenné poznatky a inšpiráciu, čím prispeli k širšiemu porozumeniu pokročilých vedeckých oblastí. Pripravujeme ďalšiu sériu popularizačných prednášok a tešíme sa na Vašu účasť!

Bezplatné IT kurzy

Národné kompetenčné centrum pre vysokovýkonné počítanie (HPC) poskytuje možnosť získať nové IT zručnosti prostredníctvom bezplatných online kurzov.

Na zimný semester sme pripravili rôznorodé kurzy, zamerané na oblasti ako umelá inteligencia, strojové učenie, programovanie v Pythone a Julii, interaktívne vizualizácie či správa databáz. Medzi nimi sú napríklad ***Jemný úvod do AI*** alebo ***Strojové učenie v Pythone***.

Účastníci majú možnosť využiť superpočítač Devana na niektoré praktické cvičenia. Kurzy pravidelne aktualizujeme na základe spätnej väzby, aby sme reflektovali aktuálne potreby a trendy.

Neváhajte a prihláste sa na niektorý z našich kurzov už dnes!

V nádhernom prostredí Vysokých Tatier sa členovia Európskych národných kompetenčných centier a Centier excelentnosti stretli na konferencii **EuroCC 2 & CoEs & CASTIEL 2 Intermediate Conference**. Podujatie sa konalo od 22. do 24. apríla a jeho cieľom bolo diskutovať o pokroku v projekte a zdieľať dosiahnuté úspechy.

Poľsko
SUPERPOČÍTAČ
ALTAIR

Česko
SUPERPOČÍTAČ
BARBORA

Česko
SUPERPOČÍTAČ
KAROLINA

Poľsko
SUPERPOČÍTAČ
EAGLE

Slovensko
SUPERPOČÍTAČ
DEVANA

Superhrdinovia VEDY

Visegrad Fund

Projekt Superhrdinovia vedy (Superheroes 4 Science), podporovaný Vyšehradským fondom, sa zameriava na propagáciu High Performance Computingu (HPC) a superpočítania, ktoré sú kľúčové pre rozvoj všetkých vedných odborov. HPC a superpočítanie sú často považované za tretí pilier vedy – popri teórii a experimente. Preto je nevyhnutné popularizovať význam superpočítačov a ich využitie, ktoré majú pozitívny vplyv na každodenný život človeka.

V nadväzujúcom projekte, ktorý teraz realizujeme, sa zameriavame na vysvetlenie High Performance Computingu (HPC) a jeho aplikácie v oblastiach, ako sú umelá inteligencia (AI) a kvantové výpočty (QC). HPC je zásadná oblasť technológie, ktorá využíva superpočítače na riešenie zložitých problémov a vykonávanie výpočtovo náročných úloh. Umelá inteligencia a kvantové výpočty sú obzvlášť náročné, no ich význam pre budúcnosť je obrovský.

Partneri projektu

Česká republika
IT4Innovations národné superpočítačové centrum pri VŠB – Technickej univerzite v Ostrave je popredným centrom v oblasti HPC, dátových analýz, AI a QC.

Poľsko
Poznańské superpočítačové a sieťové centrum (PSNC) poskytuje komplexné služby v oblasti cloudových výpočtov a HPC a prevádzkuje superpočítač Altair.

Slovensko
Národné superpočítačové centrum, z. z. p. o. (NSCC) podporuje aktivity v oblasti HPC na Slovensku.

Ciele a aktivity

Projekt má za cieľ vzdelávať širokú verejnosť a najmä inšpirovať mladú generáciu na štúdium vedecko-technických odborov. Súčasťou projektu sú aj interaktívne vzdelávacie materiály a komiksy, ktoré jednoduchým spôsobom vysvetľujú, ako vysokovýkonná technika a umelá inteligencia môžu zlepšiť a skvalitniť náš život. V rámci projektu vytvárame počítačovú hru, ktorá mladším generáciám priblíži tieto princípy využívania HPC technológie zábavnou formou.

Superheroes 4 Science je príkladom toho, ako medzinárodná spolupráca a podpora inovatívnych technológií môžu formovať budúcnosť vedy a technológií. Druhé pokračovanie projektu (2023 – 2025) je zamerané na rozšírenie povedomia o HPC, AI a QC a na inšpirovanie ďalšej generácie vedcov a technikov.

hpc focus

2024

ZODPOVEDNÝ REDAKTOR
Lucia Demovičová

GRAFIKA A DTP
Gabriela Obadalová

FOTOGRAFIE
Pavol Novák
Shutterstock

PERIODICITA
Raz ročne

VYDAVATEĽ
Národné superpočítačové centrum, z. z. p. o.

Dúbravská cesta 3484/9
841 04 Bratislava
Slovenská republika

Tel.: +421 904 816 609
eurocc@nscc.sk

eurocc.nscc.sk

v spolupráci s

Centrum spoločných činností SAV, v. v. i.
Výpočtové stredisko SAV

Dúbravská cesta 9
845 35 Bratislava
Slovenská republika

Tel.: +421 (0)2/ 3229 3111
vssav@savba.sk

vs.sav.sk

TLAČ
Neumarh tlačiareň, s. r. o.

Mlynská Dolina 5
841 04 Bratislava 4
Slovenská republika

Tel.: +421 903 446 414
bratislava@neumahr.sk

www.neumahr.sk

ISBN 978-80-89871-23-0
ISSN 2729-9090

Texty neprešli jazykovou korektúrou.

Národné superpočítačové
centrum, z. z. p. o.

Dúbravská cesta 3484/9
841 04 Bratislava
Slovenská republika

eurocc.nscc.sk

ISBN 978-80-89871-23-0
ISSN 2729-9090