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

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

[1]Národné superpočítačové centrum, Dúbravská cesta 3484/9, 84104 Bratislava, Slovenská republika
{bibiana.lajcinova,patrik.valabek}@nscc.sk
[3]nettle, s.r.o., Matúšova 56/A, 811 04 Bratislava - Staré Mesto, Slovenská republika
michal.spisiak@nettle.ai
[2]Ústav informatizácie, automatizácie a matematiky, Slovenská technická univerzita v Bratislave, Radlinského 9, 81237 Bratislava, Slovenská republika

## 1 Úvod

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).

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.

## 2 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 [1]. 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 2.1 opisujeme náš dátový súbor, vrátane procesu tvorby týchto dát. Výrazný nedostatok dostupných, reálnych pozorovaní nás prinútil vytvoriť "syntetické dáta". V časti 2.2 navrhujeme úpravy SlovakBERT-u s cieľom natrénovať a doladiť ho pre našu úlohu. V časti 2.3 skúmame iteračné zlepšenia nášho prístupu generovania syntetických dát. Záverom, v časti 3, uvádzame výsledky trénovania a diskutujeme výkonnosť modelu.

## 2.1 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 tag-y 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 „Ďalši" 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 [2]. 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 2.3. Konečný súbor trénovacích dát pozostával z viac ako $10^4$ 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 [3].

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.

## 2.2 Vývoj a trénovanie modelov

V práci boli boli použité a porovnané dva predtrénované, všeobecné modely pre slovenský jazyk: SlovakBERT [1] a destilovaná verzia tohto modelu [4]. 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 [1]. 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.

Oba modely boli upravané 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.

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

Tabuľka 2: Počet parametrov v použitých NER modeloch a ich príslušné počty parametrov pre základný model a klasifikačnú vrstvu.

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.

### 2.3 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 finalného 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 novoidentifikovanými vzormi. Dáta doplňané 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.

## 3 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.

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

| Iteration | 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. | Všatky 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).

| | | **Predikcia** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | O | B-Ulica | I-Ulica | B-ČísloDomu | I-ČísloDomu | B-Obec | I-Obec | B-PSČ | I-PSČ |
| **Skutočnosť** | 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 %.

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.

| | | **Predikcia** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | O | B-Ulica | I-Ulica | B-ČísloDomu | I-ČísloDomu | B-Obec | I-Obec | B-PSČ | I-PSČ |
| **Skutočnosť** | 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 | 0 |

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%.

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** | **Tagy** | **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 |

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.

# 4 Záver

V tejto štúdii bol natrénovaný NER model postavený na predtrénovanom LLM modeli SlovakBERT. Model bol natrénovaný výlučne na umelo vygenerovanom súbore dát. Finálne syntentické trénovacie dáta boli reprezentatívne a kvalitné, vďaka ich iteratívnemu rozširovaniu. Spolu s dolaďovaním hyperparametrov tento iteratívny prístup umožnuje 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

# 5 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-EUROHPC-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.

# Literatúra

[1] Matús Pikuliak, Stefan Grivalsky, Martin Konopka, Miroslav Blsták, Martin Tamajka, Viktor Bachratý, Marián Simko, Pavol Balázik, 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.