Univerzita Pavla Jozefa Šafárika v Košiciach

Prírodovedecká fakulta

PREPIS HOVORENEJ REČI, PODPORA SLOVENSKÉHO JAZYKA

ČLÁNOK

Študijný odbor: **Informatika**
Školiace pracovisko: **Ústav informatiky**
Vedúci záverečnej práce: **RNDr. Erik Bruoth, PhD.**

Košice 2021

Samuel Baran

Obsah

Úvod **2**

1 Teoretické východiská práce **3**

1.1 Rozpoznávanie reči 3

1.1.1 Architektúra ASR systémov 3

1.2 DeepSpeech 4

1.2.1 Výskum spoločnosti *Baidu Research* 5

1.2.2 Základné súčasti modelu *DeepSpeech* 5

1.2.3 Architektúra modelu *DeepSpeech* 7

2 Príprava datasetu **10**

2.1 Common voice 10

2.2 Spracovanie audiokníh a e-kníh 11

2.2.1 Predspracovanie audiokníh 11

2.2.2 Predspracovanie e-kníh 12

2.2.3 Forced alignment 12

Záver **13**

Úvod

Prepis hovorenej reči do textu (STT z angl. Speech To Text) je vo svetových jazykoch značne rozšírený a je bežnou súčasťou mnohých moderných technológií. Jazyky, ako je napr. anglický jazyk, sú zvýhodnené oproti minoritným jazykom, kde patrí aj slovenský jazyk, v dostupnosti vstupných dát určených na tréning v podobe rôznych zvukových nahrávok a im prislúchajúcich prepisov. To umožňuje spoľahlivé trénovanie modelov STT a tým pádom aj rýchlejšiu a efektívnejšiu implementáciu STT v praxi. Keďže táto oblasť poskytuje nedostatočnú podporu slovenského jazyka, rozhodli sme sa danú problematiku preskúmať a hľadať spôsob trénovania modelov rozpoznávania reči pre slovenský jazyk.

Kapitola 1

Teoretické východiská práce

1.1 Rozpoznávanie reči

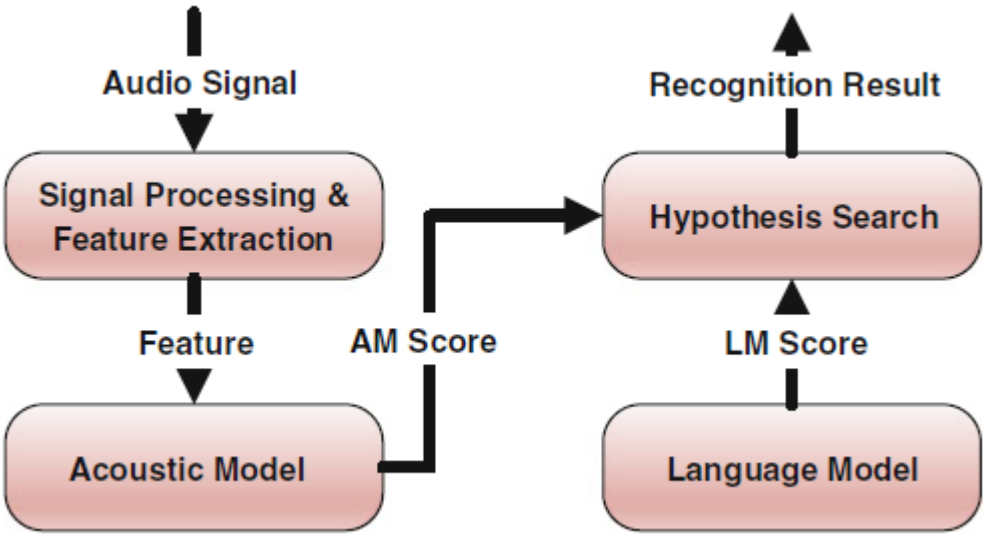
Rozpoznávanie reči, tiež známe ako Automatic speech recognition (ASR) alebo Speech to text (STT), je jednou z mnohých podoblastí vedného odboru NLP (Natural language processing) zaoberajúceho sa výskumom spracovania textovej alebo zvukovej podoby prirodzeného jazyka za pomoci informačných technológií [1]. Úlohou samotného ASR je vyvíjať techniky a systémy umožňujúce interakciu medzi používateľom a počítačom prostredníctvom zvuku [2].

1.1.1 Architektúra ASR systémov

ASR systémy pozostávajú zo štyroch základných komponentov: komponent na spracovanie signálov a extrahovanie vlastností, akustický model, jazykový model a komponent zodpovedný za zlučovanie výsledkov akustického a jazykového modelu [2] ako je popísané na obrázku 1.1.

Komponent spracovania signálov a extrakcie vlastností

Komponent spracovania signálov a extrakcie vlastností má na vstupe audio signál, ktorý následne očisťuje od šumu a skreslenia, ďalej signál prevádza zo spojitého na diskrétny a nakoniec z neho extrahuje vektory uchovávajúce vlastnosti signálu (feature vectors) slúžiace ako vstupné dáta pre za ním nasledujúci akustický model.



Obr. 1.1: Architektúra ASR systémov [2]

Akustický model

Samotný akustický model spája znalosti o akustike a fonetike a zo vstupných dát generuje skóre akustického modelu pre rôzne dlhú postupnosť feature vektorov.

Jazykový model

Jazykový model odhaduje pravdepodobnosť korektnosti predikovanej postupnosti slov, tiež nazývanej ako skóre jazykového modelu, na základe korelácií slov získaných z textových dátových korpusov.

Preľadávanie priestoru hypotéz

Posledný komponent, ako sme už vyššie zmienili, kombinuje skóre jazykového modelu a skóre akustického a ako výstup generuje postupnosť slov s najvyšším skóre, ktorá je zároveň výsledkom rozpoznávania reči.

1.2 DeepSpeech

DeepSpeech je open source nástroj rozpoznávania reči vyvinutý firmou *Mozilla*. Implementácia tohto nástroja využíva model trénovaný pomocou techník strojového učenia, inšpirovaný výskumom *Deep Speech: Scaling up end-to-end speech recognition* [3] spoločnosti *Baidu Research - Silicon Valley AI Lab*.

1.2.1 Výskum spoločnosti *Baidu Research*

Najúspešnejšie systémy rozpozávania reči sa spoliehajú na sofistikované pipeliney pozostávajúce z viacerých algoritmov a manuálne vyvíjaných komponentov, akými sú napríklad akustické modely alebo skryté Markovove modely (HMM).

Výskum spoločnosti *Baidu Research* uvádza [3], že zlepšenie výsledkov bežných systémov je najčastejšie dosahované ladením práve týchto súčastí, ktorý si však vyžaduje značné úsilie odborníkov.

Preto autori výskumu predstavujú end-to-end systém rozpoznávania reči nazývaný *Deep Speech*, ktorý tieto komplikované metódy nahrádza síce robustnou, ale jednoduchou rekurentnou neurónovou sieťou(RNN). Kombináciou s jazykovým modelom tento nástroj dosahuje na náročných úlohách rozpoznávania reči lepšie výsledky ako tradičné metódy, za čo vďačí paralelnému tréningu veľkej neurónovej siete na viacerých jednotkách GPU s použitím tisícok hodín dát.

Využívanie výhod end-to-end strojového učenia ale prináša viacero výziev. Prvou je budovanie veľkých olabelovaných datasetov, čo môže byť problém pri minoritných jazykoch, akými su napríklad aj slovenský jazyk, ktorých rozšírenosť je znevýhodňujúcim faktorom. Ďalšou, nie menej dôležitou výzvou, je vývoj inovatívnych metód tréningu sietí dostatočne veľkých na to, aby boli schopné efektívne využiť potenciál vyššie zmienených datasetov.

1.2.2 Základné súčasti modelu *DeepSpeech*

Predtým ako si popíšeme architektúru nástroja DeepSpeech, potrebujeme zadefinovať a priblížiť potrebné pojmy akými sú: *MFCC*, *LSTM* sieť a *CTC* funkcia a *Adam optimizer* .

MFCC

Výpočet koeficientov *MFCC*, z anglického Mel-Frequency Cepstral Cofficient, je jednou z najpopulárejších techník extrahovania vlastností zvuku, využívaná v rozpoznávaní reči založenom na frekvenciách zvukových nahrávok.

Zvukový signál je naprv rozdelený na za sebou idúce časové rámce voliteľnej dĺžky, ktoré sú vo väčšine systémov vytvárané, tak aby sa rámce prekrývali, čo zabezpečuje plynulejší prechod medzi susednými rámcami. Každý takýto rámec sa ďalej delí za pomoci *Hammingových okien* za účelom eliminácie nespojitosti na okrajoch rámcov. Následne sa pomocou *rýchlej Fourierovej transformácie (FFT)* extrahujú frekvenčné

zložky každého zvukového rámca, na ktoré sú následne aplikované logaritmické *Mel-Scaled* filtre, kde filtre pre vyššie frekvencie majú väčší rozsah a filtre pre nižšie frekvencie nižší, ale šírka ich pásma ostáva rovnaká, čím je zabezpečený prevod frekvencií s logaritmickým rastom na takmer lineárne *Mel* frekvencie, ktoré lepšie modelujú spôsob vnímania zvuku ľudským uchom. Posledným krokom je spočítanie *diskrétnej kosínusovej transformácie (DCT)* pre výstupy filtrov.

Týmto postupom dostávame množinu koeficientov nazývanú aj akustický vektor, ktorý uchováva foneticky dôležité charakteristiky reči, čím sa stáva vhodným vstupom pre modely rozpoznávania reči.

LSTM

Bežné rekurentné neurónové siete *RNN* majú oproti klasickým dopredným neurónovým sieťam doležitý benefit, a to schopnosť uchovávať kontext predchádzajúcich vstupov a využívať ho na predikciu výsledku aktuálneho vstupu. Aj táto výhoda má svoje úskalie, ktorým je *vanishing gradient problem* poukazujúci na fakt, že klasické *RNN* sú schopné využívať len obmedzenú šírku kontextu.

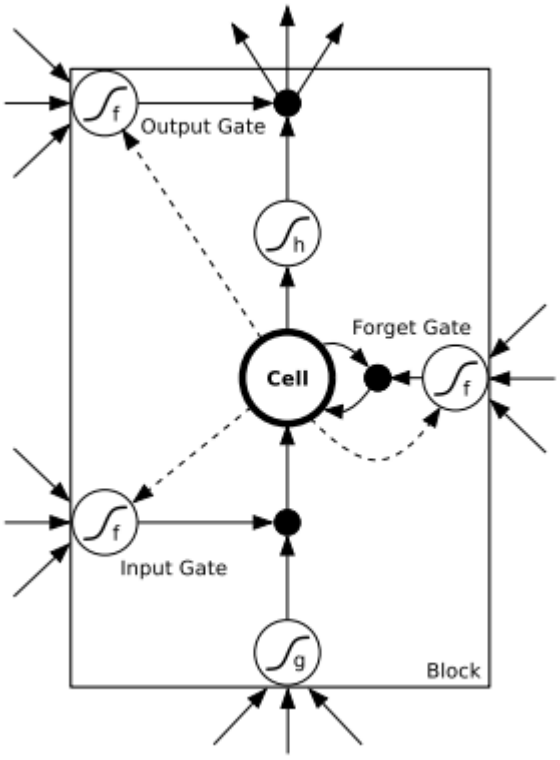
Tento problém sa podarilo vyriešit *LSTM (Long Short-Term Memory)* architektúrou rekurentných neurónových sietí [5] popísanou na obrázku 1.2, ktorá pozostáva z množiny rekurentne pospájaných sietí tiež známych ako pamäťové bloky, ktoré možno prirovnať k diferencovateľnej verzii pamäťových čipov. Každý takýto blok pozostáva z niekoľkých navzájom prepojených pamäťových jednotiek a troch multiplikativnych jednotiek (vstupná, výstupná a premazávacia brána).

Takto navrhnutá sieť vie zaznamenávať široký kontext, ktorý nie je ohraničený, a preto je schopná riešiť mnoho úloh, ktoré ostávajú pre iné architektúry *RNN* neprekonané.

CTC

CTC (connectionist temporal classification) je výstupná vrstva *RNN*[5]. Ako vyplýva z názvu, CTC bola špeciálne nadizajnovaná na úlohy klasifikácie na časových radoch, teda na olabelovanie sekvencií, pre ktoré zarovnanie vstupov a výstupov nie je známe.

Narozdiel od iných prístupov, táto architektúra nevyžaduje kombináciu neurónových sietí a *skrytého Markovovho modelu (HMM)* alebo segmentáciu trénigových dát, prípadne dodatočné externé spracovanie výstupov siete za účelom extrahovania predikcie z výstupov siete.



Obr. 1.2: Architektúra pamäťového bloku LSTM siete s jednou pamäťovou jednotku [5]

Na výstup siete je aplikovaná *Softmax* aktivačná funkcia, čo umožňuje interpretovať výstup siete ako distribúciu pravdepodobností naprieč všetkými triedami, do ktorých klasifikujeme. Na základe takéhoto výstupu potom možno určiť *loss funkciu* a tiež aj predikovanú triedu.

1.2.3 Architektúra modelu *DeepSpeech*

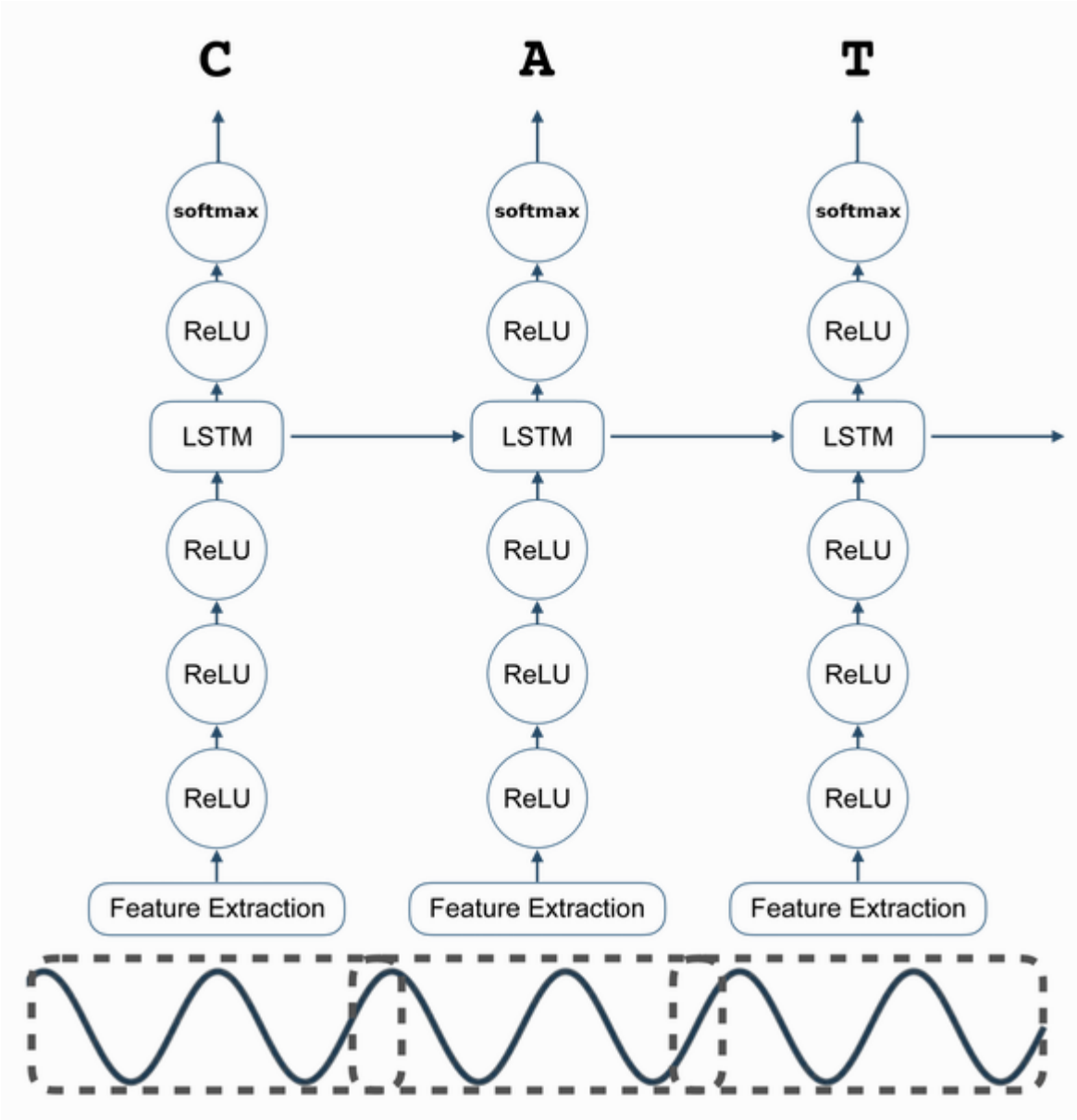
V tejto kapitole si bližšie popíšeme architektúru modelu DeepSpeech firmy Mozilla [6] zobrazenú na obrázku 1.3.

Jadrom tohto nástroja je rekurentná neurónová sieť, trénovaná na generovanie textovej transkripcie vstupných zvukových nahrávok. Architektúru tohto je možné popísať nasledovnými vzťahmi:

Majme nejakú zvukovú vzorku reči x a jej prepis y z trénovacej množiny:

$$S = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ...\}$$

Každú takúto vzorku reči $x^{(i)}$ je možné rozdeliť na súvislú postupnosť dĺžky $T^{(i)}$ pozostávajúcu z okien fixnej dĺžky. Tieto okná sú následne prevádzané na *feature*



Obr. 1.3: Architektúra DeepSpeech modelu [6]

vektory dĺžky *n_input*, ktoré sú tvorené *MFCC* konštantami, kde *n_input* je počet konštánt definovaný v konfigurácii modelu.

Na dosiahnutie lepších výsledkov a možnosti efektívnejšie predikovať prepisy týchto okien, sú samotné *feature* vektory charakterizujúce dané okná rozšírené o kontext, a teda každý jeden *feature* vektor prislúchajúci jednému oknu je rozšírený pridaním *n_context* okien zľava aj sprava, kde konštanta *n_context* je definovaná v konfiguácii modelu. Týmto sa veľkosť *feature* vektoru charakterizujúceho jedno okno zmenila z *n_input* na $n\_input * (2 * n\_context + 1)$.

Takto rozšírené vektory, sú vstupom rekurentnej neurónovej siete pozostávajúcej z piatich skrytých vrstiev, kde $h^{(l)}$ je označenie l-tej skrytej vrstvy a $h^{(0)}$ označuje

vstup siete.

Prvé tri nerekurentné vrstvy pracujú s dátami nezávisle od poradia v akom sa vyskytovali v pôvodnom zázname reči x a výpočet každého časového úseku t na týchto vrstvách vyzerá nasledovne:

$$h_t^{(l)} = g(W^{(l)} h_t^{(l-1)} + b^{(l)})$$

kde $g(z) = min\{max\{0, z\}, 20\}$ je orezan=a *ReLu* aktivačná funkcia a $W^{(l)}$, $b^{(l)}$ sú váhy a bias l-tej skrytej vrstvy.

Štvrtá vrstva je rekurentná LSTM vrstva obsahujúca n_hidden skrytých jednotiek s doprednou rekurenciou $h_t^{(f)}$:

$$h_t^{(f)} = g(W^{(4)} h_t^{(3)} + W_r^{(f)} h_{t-1}^{(f)} + b^{(4)})$$

Musíme tu však upozorniť na fakt, že hodnoty $h_t^{(f)}$ musia byť počítané postupne od $t = 1$ po $t = T^{(i)}$.

Piata nerekurentná vrstva počíta hodnoty, ktoré korešpondujú s distribúciou pravdepodobností naprieč znakmi abecedy pre každý časový úsek t a každé písmeno abecedy k nasledovne:

$$h_{t,k}^{(6)} = \hat{y}_{t,k} = (W^{(6)} h_t^{(6)})_k + b_k^{(l)}$$

Pomocou predikcie $\hat{y}_{t,k}$ model vypočíta *CTC loss* $\mathcal{L}(\hat{y}, y)$, čím zistí chybu predikcie $\hat{y}$ vzhľadom na prepis y. Pri tréningu potom vieme určiť hodnotu gradientu $\nabla\mathcal{L}(\hat{y}, y)$ vzhľadom na výstupy siete porovnávané s prepismi a ten následne šíriť celou architektúrou modelu s použitím stochastického optimalizačného algoritmu *Adam*.

Kapitola 2

Príprava datasetu

Jednou z najdôležitejších prvkov systémov využívajúcich neurónové siete je množstvo kvalitných dát, ktorých príprava vyžaduje značné úsilie a zaberá pomerne veľkú časť vývoja a trénovania modelov.

Väčšina známych problémov umelej inteligencie má voľne prístupné datasety určené práve na tréning a ohodnocovanie efektivity riešení daných úloh. Ak by sme sa však zamerali na menej rozšírené úlohy, zistíme, že dostupnosť datasetov nie je pravidlom, čo je aj náš prípad.

V tejto kapitole si preto priblížime viaceré možnosti, ako získať potrebné dáta.

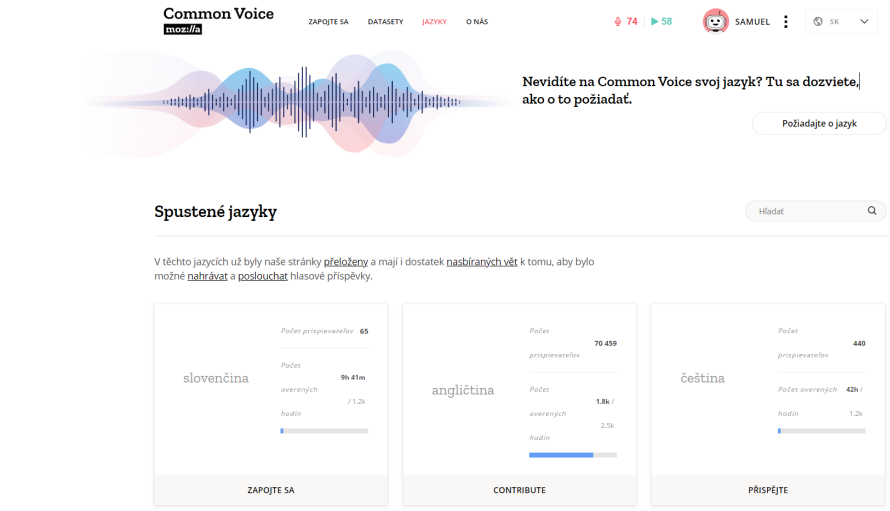
2.1 Common voice

Prvou možnosťou získania datasetu na tréning modelu bola iniciatíva Mozilly *Common vioce*, vytvárajúca open source hlasovú databázu.

Tento projekt vznikol popri projekte DeepSpeech za účelom zhromažďovania dát rozličných jazykov. Projekt funguje na princípe dobrovoľných prispievateľov, ktorí môžu prispevať buď nahrávaním reči ekvivalentnej poskytnutým textom alebo verifikáciou už nahratých nahrávok.

Momentálne je sprístupnených vyše 40 datasetov pre rôzne jazyky, ku ktorým patria napríklad angličtina, nemčina, ale aj menej rožšírené jazyky ako čeština, slovinčina alebo gruzínština.

Ďalšie jazyky sú vo fáze zberu dát, a teda ešte nedosiahli veľkosť potrebnú na sprístupnenie. Medzi nimi sa nachádza aj slovenčina, ktorá bola v čase tvorby nášho datasetu len vo fáze zbierania potrebných textov vhodných na ďalšie spracovanie a preto sme hľadali inú alternatívu.



Obr. 2.4: Screenshot stránký iniciatívy Common Voice s ukážkou stavu zberu dát pre jazyky slovenčina, čeština a angličtina.

2.2 Spracovanie audiokníh a e-kníh

Ďalším zdrojom nahrávok a im prislúchajúcich prepisov sú audioknihy a k nim kompatibilné e-knihy. Problémom týchto zdrojov je ich dĺžka, nejednotný formát alebo prípadné reklamy zakomponované v nahrávkach.

Mali sme k dispozícii 3 diela rôznych autorov s celkovou dĺžkou približne 50 hodín. Audoioknihy boli nahrávané po častiach a e-sme mali prístupné v epub formáte, takže bolo potrebné ich ďalšie spracovanie, ktoré si popíšeme v nasledujúcich podkapitolách.

Predtým by sme však radi načrtli myšlienku celého procesu spracovania kníh. Z vlastností datasetu potrebného na tréning modelu popísaných v dokumentácii vyplýva, že sme potrebovali vytvoriť množinu nahrávok a ich prepisov, kde dĺžka nahrávok kvôli obmedzeniam modelu nepresahuje 10 sekúnd. Mali sme však knihu rozdelenú na menšie časti trvajúce desiatky minút a prepis celej knihy, ktorý bolo potrebné rozdeliť na kratšie prepisy prislúcajúce jednotlivým nahrávkam a následne nájsť nástroj na segmentáciu týchto dát.

2.2.1 Predspracovanie audiokníh

Väčšina audiokníh bola nahrávaná po kapitolách a uložená v samostatných súboroch nesúcich názvy daných kapitol, ktoré neobsahovali žiadne reklamy alebo iné prídavky umiestnené, na začiatku alebo na konci a preto nebolo potrebné tieto knihy ďalej upravovať. Výnimkou bola jedna kniha, ktorá mala omnoho viac nahrávok ako kapi-

tol a teda nebolo možné jednoducho nájsť prislúchajúcu časť textu. Tieto nahrávky obsahovali aj reklamy ktorých prepisy neboli zaznamenané, a preto sme sa rozhodli danú knihu vylúčiť z množiny dát, ktoré chceme ďalej spracovávať.

2.2.2 Predspracovanie e-kníh

E-knihy sme mali prístupné v epub formáte, z ktorého bolo potrebné extrahovať text, na čo sme použili online konvertér. Túto možnosť spracovania sme si zvolili pre nízky počet epub súborov, ktoré bolo potrebné previesť na text. Pri väčšom počte zdrojov, by za účelom automatizácie bolo potrebné vyhľadať knižničné implementácie tohto prevodníka.

V ďalšej fáze sme rozdeľovali text na kapitoly pomocou oddeľovačov nachadzajúcich sa v texte, ktorými boli jednotné názvvy kapitol jednotného formátu vrámci jednej knihy, avšak na každú knihu bolo potrebné identifikovať vhodný oddeľovač, čo bránilo plne automatizovanému prístupu predspracovania týchto e-kníh.

Následne sme text každej kapitoly uložili tak, aby bola každá veta na samostatnom riadku.

2.2.3 Forced alignment

Výstupy predspracovania audiokníh a e-kníh bolo následne potrebné segmentovať za účelom skrátenia jednotlivých nahrávok. Na túto úlohu sme využili implementáciu *Forced alignment-u* s názvom *Aeneas*, ktorej výstupom bolo zarovnanie ortografických prepisov jednotlivých viet a ich zvukových záznamov, na základe ktorého sme rozstrihali narávky na časti prislúchajúce jednotlivým vetám, čím sme dosiahli potrebný stav datasetu.

Záver

V tomto článku sme popísali teoretické východiská práce počnúc všeobecnou architektútou ASR sytémov, ďalej opisom modelu DeepSpeech, pri ktorom sme spomenuli výskum spoločnosti Baidu research, z ktorého model vychádzal, priblížili sme architektúru modelu a vysvetlili sme technológie využité pri implementácii modelu.

Tiež sme zadefinovali formát dát potrebných na tréning a skúmali a analyzovali sme spôsoby získavania takýchto dát. Aj napriek viacerým postupom sme dokázali získať len obmedzený dataset, čo ponecháva priestor pre skúmanie ďalších možností rozšírenia množiny trénovacích vzoriek.

Máme rozpracovný tréning modelu na týchto dátach a technológiu, ktorá by mohla pri malých datasetoch dosahovať lepšie výsledky. Vývoj a overovanie spomenutej technológie zatiaľ nie sú uzavreté, preto finálne závery týchto fáz v článku neuvádzame.

Zoznam použitej literatúry

[1] Chowdhury, G. G., 2003. Natural language processing. In: *Annual review of information science and technology* [online]. Vol. 37, no. 1, p. 51–89. ISSN 0066-4200. Dostupné na: https://asistdl.onlinelibrary.wiley.com/doi/epdf/10.1002/aris.1440370103

[2] Dong, Y. a Li, D., 2015. *Automatic Speech Recognition:A Deep Learning Approach.* Londýn: Springer-Verlag. ISBN 978-1-4471-5778-6.

[3] Hannun, A. a kol., 2014. *Deep speech: Scaling up end-to-end speech recognition*: výskumná práca [online]. Dostupné na: https://arxiv.org/abs/1412.5567

[4] Dave, N., 2013. Feature extraction methods LPC, PLP and MFCC in speech recognition. In: *International journal for advance research in engineering and technology* [online]. Vol. 1, no. 6, p. 1–4. ISSN 2393-9877. Dostupné na: https://d1wqtxts1xzle7.cloudfront.net/40023802/Feature_Extraction_Methods_LPC__PLP_and_MFCC.pdf?1447603451=&response-content-disposition=inline\%3B+filename\%3DFeature_Extraction_Methods_LPC_PLP_and_M.pdf&Expires=1618495674&Signature=emZT24S1a-J-GAcZzgVxP-N6O~PvQUhOAda0pfvFlemt1cDGrRPZfs-ujSW~230MDTI75TaD51yj5OZwvAHQEkbw5atYafBwJt2-FgHNCJ0rVbEy6qVtK1jq-GzBI2fV3jTbsFOrpN8vl0G6YJXetnbWhUVu0tjXD3lYf46d0zbYED4raY2F-ogV4POWsZN1TQzdd4b16~aKYN-XVHUTgh2wlSWpLkfW27DW4XkaNx~eK5nsI4at1RvCklCBhxtg7zWOhg4G2ClR5WzKKmGPg5w7yveNkZE5bkjsmUwEnVz19c5tbD8-Q__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

[5] Kawakami, K., 2008. *Supervised sequence labelling with recurrent neural networks.* [online]: diplomová práca. Toronto: CS [cit. 2021-04-15]. Dostupné na: http://www.cs.toronto.edu/~graves/preprint.pdf

[6] Mozilla Corporation, 2020. *DeepSpeech's documentation*: DeepSpeech Model [online]. Dostupné na: `https://deepspeech.readthedocs.io/en/r0.9/DeepSpeech.html`