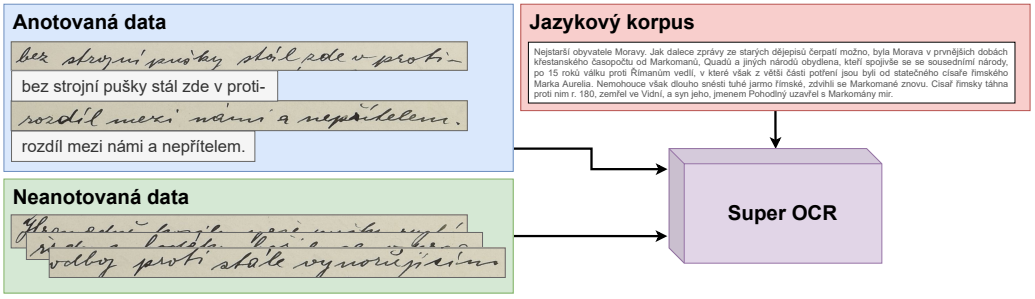
BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2021
http://excel.fit.vutbr.cz

Využití neanotovaných dat pro trénování OCR

Petr Buchal*



Abstrakt
Vytvoření kvalitního systému rozpoznání textu (OCR) vyžaduje velké množství anotovaných dat. Získání, potažmo vytvoření anotací je nákladný proces. Tato práce se zabývá využitím neanotovaných dat pro trénování OCR neuronové sítě. Navržená metoda využívající neanotovaná data spadá do kategorie self-training algoritmů. Nejprve je na omezeném množství anotovaných dat natrénován počáteční model neuronové sítě. Ten je následně spolu s jazykovým modelem použit k vygenerování několika nejpravděpodobnějších variant pseudo-štítků neanotovaných dat. Takto strojově anotovaná data jsou smíchána s trénovacími daty, která byla použita k vytvoření počátečního modelu a následně jsou využita k natrénování cílového modelu. Na ručně psaném ICFHR 2014 Bentham datasetu dosahuje nejlepší takto trénovaný model 1.99 % CER, což je relativní zlepšení o 23 % oproti počátečnímu modelu trénovaném pouze na anotovaných datech. Za pomocí navržené metody lze zvýšit úspěšnost OCR pomocí neanotovaných dat.

Klíčová slova: neuronová síť — rozpoznání textu — self-training — neanotovaná data — jazykový model

Přiložené materiály: N/A

*xbucha02@stud.fit.vutbr.cz, *Fakulta informačních technologií, Vysoké učení technické v Brně*

1. Úvod

Dnešní doba je bezpochyby dobou digitalizace. Instituce po celém světě se snaží digitalizovat své dokumenty, aby k nim umožnili přístup budoucím generacím. Toto platí obzvláště pak o historických dokumentech, které jsou náročné na údržbu a do budoucna hrozí ztráta cenných informací. Digitalizace starých archivů vytvořila poptávku po vysoce spolehlivých OCR systémech, které jsou potřebné k indexování textu např. pro možnost vyhledávání. Vytváření OCR systémů ovšem vyžaduje velké množství ručně anotovaných dokumentů, což je finančně nákladné. U mnohých dokumentů s málo užívanými jazyky jsou pak k přepisování potřeba dokonce odborníci, kteří konečnou cenu ještě zvyšují. Tato práce se zabývá vytvořením kvalitního OCR s pomocí neanotovaných dokumentů, a tak minimalizováním potřebných člověkohodin strávených anotováním.

Navržená metoda generuje pomocí OCR neuronové sítě a jazykového modelu strojové přepisy pro neanotovaná data tzv. pseudo-štítky. Metoda tak předpokládá dostupnost určitého množství anotovaných dat pro natrénování počáteční OCR neuronové sítě a data potřebná k vytvoření jazykového modelu, který koresponduje s neanotovanými daty. Cílová OCR neuronová síť je pak trénovaná na kombinaci anotovaných a stro-

jově anotovaných dat. Tento způsob učení se nazývá self-training.

2. OCR systém

OCR systém použitý v této práci je shodný se systémem v AT-ST [1], který je tvořen dvěma moduly - OCR neuronovou sítí a jazykovým modelem. Vstupem OCR systému je lokalizovaný řádek obsahující text. OCR neuronová síť se v principu skládá ze dvou částí, které postupně zpracovávají vstupní obrazovou informaci. V první části je vstupní obraz zpracován pomocí konvolučních vrstev, které jsou určené ke zpracování obrazu a dokáží z něj extrahovat příznaky reprezentující klíčové informace pro další zpracování. Takto získané příznaky jsou následně zpracovány rekurentními vrstvami neuronové sítě. Rekurentní vrstvy pracují s příznaky sekvenčně a jejich výstupem jsou pravděpodobnosti jednotlivých znaků na jednotlivých pozicích v obraze.

2.1 Connectionist temporal classification

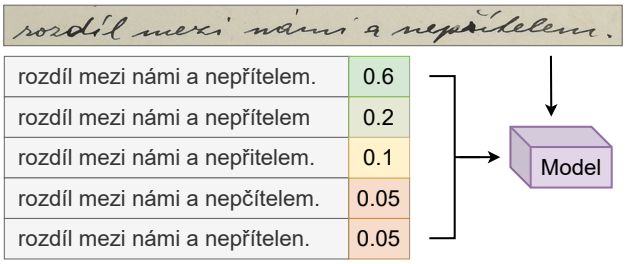
Výstupem OCR systému je pravděpodobnost jednotlivých znaků pro jednotlivé úseky vstupního obrazu. Connectionist temporal classification (CTC) [2] je chybová funkce využívaná k trénování OCR neuronových sítí. Její použití je výhodné, protože pracuje pouze s textovým přepisem řádku a nevyžaduje informace o přesných úsecích, kde se nacházejí jednotlivá písmena reference [3].

2.2 Jazykový model

Jazykový model je funkce, která zachycuje pravděpodobnostní rozložení sekvence slov (písmen) v přirozeném jazyce. Nejmodernější jazykové modely jsou tvořeny LSTM neuronovými sítěmi [4], které využívají schopnosti naučit se distribuované reprezentace. Distribuovaná reprezentace slova je vektor hodnot reprezentující vlastnosti, které charakterizují význam slova a vzájemně se nevylučují. Neuronová síť se tyto vlastnosti učí v podobě spojitých hodnot [5].

3. Self-training

Práce [6] definuje self-training následovně. Mějme m anotovaných dat (L) a n neanotovaných dat (U) tak, že $n \gg m$. Model M_0 je natrénován na anotovaných datech L. Následně je M_0 využit k vytváření pseudo-štítků pro neanotované data U. Neanotovaná data U s nejjistějšími pseudo-štítky, jejichž počet je p, jsou přimíchány k anotovaným datům L. Následně je na $m + p$ trénován model M_1. Tento proces je opakován tak, že na konci každého cyklu vznikne model M_{i+1}, který je pak využit k vygenerování $n - p$ pseudo-štítků



Obrázek 1. Trénování OCR neuronové sítě pomocí CTC s měkkými pseudo-štítky.

dosud neanotovaných dat U. Self-training končí v okamžiku, kdy byla k trénování použita všechna neanotovaná data nebo se model trénováním dále nezlepšuje.

4. Návrh metody

Self-training klade důraz na výběr p dat s nejjistějšími pseudo-štítky. Pseudo-štítky s nižší jistotou obsahují častěji chyby a jejich přítomnost v trénovací sadě snižuje úspěšnost neuronové sítě [6]. Pokud mají všechny pseudo-štítky při učení stejnou váhu, jedná se o tzv. tvrdé pseudo-štítky. Jejich použití může snížit úspěšnosti neuronové sítě, protože jisté pseudo-štítky s nízkým počtem chyb mají při učení stejnou váhu jako nejisté pseudo-štítky s vyšším počtem chyb [7].

Tento jev může pomoci odstranit trénování neuronové sítě s měkkými pseudo-štítky, tedy pseudo-štítky váhovanými podle jejich jistot, viz obrázek 1. K jednomu trénovacímu řádku v tomto případě náleží několik nejpravděpodobnějších pseudo-štítků namísto jednoho pseudo-štítku s pravděpodobností 1 jako v případě tvrdých pseudo-štítků. Při trénování je pak gradient pseudo-štítku vynásobený jeho váhou (jistotou). Jako efektivní metriku jistoty pseudo-štítku pak lze využít jeho posteriorní pravděpodobnost, která koreluje s chybovostí [1].

Dalším přístupem, jak snížit počet chyb v pseudo-štítcích, je jejich korekce jazykovým modelem [8]. Jazykový model na svůj vstup vezme výstup OCR neuronové sítě a pomocí algoritmu Beam Search [9] vygeneruje k nejpravděpodobnějších hypotéz, které lze využít jako pseudo-štítky. V případě tvrdých pseudo-štítků se do trénovací sady zařazuje hypotéza s největší jistotou.

Navrhovaná metoda kombinuje výše zmíněné přístupy a trénuje OCR neuronovou síť na k variantách pseudo-štítku pro jeden řádek tedy na tzv. měkkých pseudo-štítcích, viz obrázek 1. Ty mají oproti klasickým tvrdým pseudo-štítkům výhodu v tom, že tam kde má tvrdý pseudo-štítek velkou jistotu, budou mít v případě měkkých pseudo-štítků ostatní pseudo-štítky zanedbatelnou jistotu a neuronová síť se bude učit stále primárně z něj. Na druhou stranu, pokud má původní

Tabulka 1. Počty řádků v použitých datasetech.

	Trénovací	Validační	Testovací
Související anotované	189 805	—	700
Cílové anotované	9 194	1 415	860
Cílové neanotované	1 141 566	—	—

tvrdý pseudo-štítek nízkou jistotu, jistota u dalších pseudo-štítků bude více vyrovnaná a neuronová síť se tak bude učit rovnoměrněji z více variant.

5. Experimenty

Navrženou metodu jsem experimentálně otestoval a porovnal výsledky s prací AT-ST [1], která zkoumá na stejných datech využití tvrdých pseudo-štítků.

Úspěšnost modelů udávám v metrice CER (chybovost znaků), která se počítá následovně

$$\mathrm{CER} = \frac{S+I+D}{N}, \tag{1}$$

kde N je počet znaků reference a S, I, D jsou počty znaků přepisu, které jsou změněné, vložené a smazané vůči referenci. Čím nižší je hodnota CER, tím úspěšnější je zkoumaný model.

5.1 Datasety

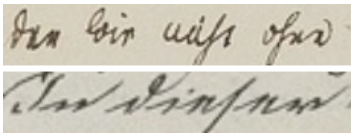
Experimenty jsem provedl na ručně psaných datech. Data jsem rozdělil na 3 množiny, viz tabulka 1. Účelem self-trainingu je zvýšení úspěšnosti modelu na cílových datech. Související data jsou vizuálně podobná datům cílovým a do trénování se přidávají kvůli malému množství anotovaných cílových dat. Související anotovaná data obsahují řádky z ICDAR 2017 READ datasetu [10], cílová anotovaná data obsahují řádky z ICFHR 2014 Bentham datasetu [11]. Cílové neanotované řádky pocházejí z datasetu získaného od Bentham projektu[1]. Ukázky řádků jednotlivých datasetů jsou na obrázku 2. READ dataset obsahuje stránky psané v němčině, italštině a francouzštině. Bentham dataset a dataset získaný od Bentham projektu obsahuje stránky psané v angličtině raného 19. století.

Jazykový model byl trénován na datasetu Wikitext-103 [12], který obsahuje přibližně 530M znaků (103 M anglických slov) a na anotovaných cílových datech pocházejících z ICFHR 2014 Bentham datasetu [11].

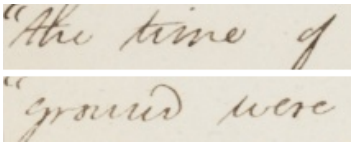
5.2 Maskovací augmentace

S cílem zlepšení implicitně modelované jazykové znalosti OCR neuronové sítě byly v experimentech na trénovací řádky aplikovány maskovací augmentace [1]. Ty nahrazují část řádku šumem generovaným z rovnoměrného rozložení pravděpodobnosti, zatímco referenční přepis zůstává stejný. Počet maskovaných oblastí

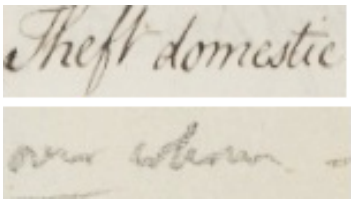
[1] https://www.ucl.ac.uk/bentham-project



(a) ICDAR 2017 READ dataset.

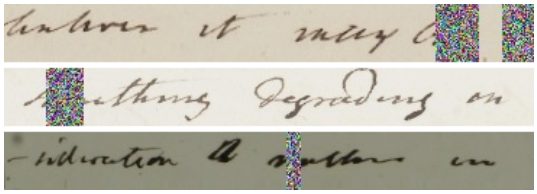


(b) ICFHR 2014 Bentham dataset.



(c) Dataset od Bentham projektu.

Obrázek 2. Ukázky krátkých řádků z použitých datasetů.



Obrázek 3. Ukázky maskovací augmentace.

jednoho řádku je získán z binomického rozložení pravděpodobnosti. Výška maskované oblasti odpovídá výšce řádku a její délka se získává z rovnoměrného rozložení pravděpodobnosti. Ukázka maskovacích augmentací je k dispozici na obrázku 3.

5.3 Generování měkkých pseudo-štítků

Po natrénování na anotovaných datech počáteční model zpracuje neanotované řádky. Jazykový model pro každý zpracovaný řádek vygeneruje pomocí algoritmu Beam Search [9] 16 nejpravděpodobnějších hypotéz. Hypotézy s největší posteriorní pravděpodobností jejichž součet tvoří 90% jsou přidány do trénovacího datasetu jako měkké pseudo-štítky. Hodnota 90 % byla zvolena experimentálně tak, aby v případě jistých řádků nebyly do trénovacího datasetu zbytečně přidávány hypotézy s pravděpodobností blížící se nule. Pokud 16 nejpravděpodobnějších hypotéz obsahuje hypotézu prázdného řetězce, tak se do trénovacího datasetu nepřidává žádná a řádek je vyřazen.

Experiment byl proveden na dvou různě velkých datových sadách, které zkoumají vliv množství dostupných cílových anotovaných dat. Malá datová sada obsahuje 10 % cílových anotovaných dat, která byla vybrána náhodně. Velká datová sada obsahuje 100 %

Tabulka 2. Počty trénovacích řádků v jednotlivých datových sadách. Ve velké datové sadě bylo použito 9 194 jedinečných cílových anotovaných řádků, které se při trénování vyskytovaly 10× častěji než ostatní řádky.

	Malá	Velká
Související anotované	189 805	189 805
Cílové anotované	894	9 194×10
Cílové strojově anotované	1 002 351	1 022 199

cílových anotovaných dat, kterým byla při trénování zvednuta váha pomocí 10× častějšího výskytu v trénovacích datech. Přehled množství užitých trénovacích dat v obou datových sadách je k dispozici v tabulce 2. Toto množství anotovaných dat bylo použito jak k natrénování jednotlivých počátečních OCR neuronových sítí, tak k natrénování jednotlivých jazykových modelů.

5.4 Self-training s měkkými pseudo-štítky

Výsledky experimentu jsou k dispozici v tabulce 3. Oba cílové modely předčily v CER jak odpovídající počáteční modely, tak nejlepší odpovídající modely trénované na tvrdých štítcích [1]. Cílový model v experimentu s malou datovou sadou dosáhl 3.98 % CER, což je relativní zlepšení o 38 % oproti počátečnímu modelu a cílový model v experimentu s velkou datovou sadou dosáhl 1.99 % CER, což je relativní zlepšení 22 % oproti počátečnímu modelu. Na obou datových sadách bylo dosaženo lepších výsledků než při shodných experimentech s tvrdými štítky [1]. Na menší datové sadě bylo dosaženo relativního zlepšení o 10 % a na velké datové sadě o 4 %.

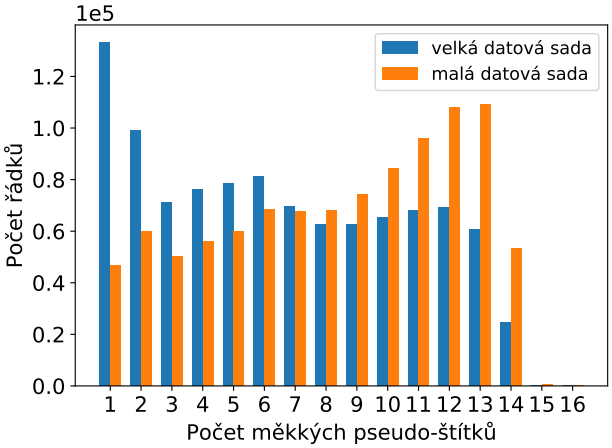
S větším množstvím anotovaných dat stoupá úspěšnost počátečního modelu, což způsobuje akceptaci menšího počtu variant pseudo-štítků s větší pravděpodobností, viz obrázek 4.

Architektury OCR neuronové sítě a jazykového modelu jsou kvůli porovnatelnosti shodné s architekturami v AT-ST [1]. Architektura OCR neuronové sítě vychází z CRNN [13], kde jsou konvoluční bloky následované rekurentními vrstvami. K inicializaci konvolučních vrstev na začátku trénování byly využity váhy z předtrénované architektury VGG16. Vstupní řádky mají konstantní výšku 40 pixelů a variabilní šířku W. Výstup neuronové sítě má rozměry $W/4 \times |V|$, kde $|V|$ je počet prvků množiny výstupních symbolů zahrnující i prázdný symbol. Jazykový model představuje LSTM neuronová síť shodná s neuronovou sítí v AT-ST [1], která má 2 vrstvy s počtem 1 500 buněk.

OCR modely byly trénovány 250 000 epizod optimalizačním algoritmem Adam a chybovou funkcí CTC. Na začátku trénování byl učící koeficient nastaven na hodnotu 2×10^{-4}, jeho hodnota se skokově snižovala na polovinu po 150 a 200 tisících epizodách. Velikost trénovacího vzorku byla nastavena na 32.

Tabulka 3. Výsledky modelů na cílových testovacích datech v CER [%].

	Malá	Velká
Počáteční model	6.43	2.58
Cílový model - tvrdé štítky	4.41	2.07
Cílový model - měkké štítky	**3.98**	**1.99**



Obrázek 4. Počty měkkých pseudo-štítků na jeden neanotovaný řádek v trénovacím datasetu. Oranžová barva reprezentuje pseudo-štítky neanotovaných dat generovaných počátečním modelem trénovaným na malé datové sadě. Modrá barva reprezentuje pseudo-štítky generované počátečním modelem trénovaným na velké datové sadě.

6. Závěr

Navrhl jsem self-training metodu pro trénování OCR pomocí měkkých pseudo-štítků generovaných jazykovým modelem. Experimentálně jsem metodu ověřil na dvou datových sadách. Datové sady se lišily v množství dostupných cílových anotovaných dat. Cílový model trénovaný na menší datové sadě dosáhl na testovacích datech 3.98% CER, což je relativní zlepšení o 38 % oproti počátečnímu modelu. Cílový model trénovaný na větší datové sadě dosáhl 1.99% CER, což je relativní zlepšení o 23 % oproti počátečnímu modelu. Na obou datových sadách dosáhla navržená metoda lepších výsledků než srovnatelná metoda využívající trénování na tvrdých pseudo-štítcích [1]. Na menší datové sadě navržená metoda dosáhla relativního zlepšení 10 % a na velké datové sadě 4 %.

Dalšího zvýšení úspěšnosti OCR by mohlo být dosaženo opakováním celého procesu trénování s cílovým modelem a korekcí výstupů cílového modelu jazykovým modelem. Problematice využití neanotovaných dat pro trénovaní OCR se budu věnovat i nadále

v rámci diplomové práce.

Poděkování

Tato práce vznikla za podpory Ministerstva kultury České republiky v rámci projektu NAKI II PERO (DG18P02OVV055).

Tímto bych chtěl poděkovat vedoucímu mé práce Ing. Michalu Hradišovi, Ph.D. za odborné vedení a cenné rady.

Literatura

[1] Karel Beneš and Martin Kišš. AT-ST: Self-Training Adaptation Strategy for OCR in Domains with Limited Transcriptions. 2021.

[2] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural 'networks. volume 2006, pages 369–376, 01 2006.

[3] Harald Scheidl. An intuitive explanation of connectionist temporal classification. blogpost (english), June 2018.

[4] Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. Lstm neural networks for language modeling. 09 2012.

[5] Y. Bengio. Neural net language models. *Scholarpedia*, 3(1):3881, 2008. revision #140963.

[6] Deepayan Das and C. Jawahar. *Adapting OCR with Limited Supervision*, pages 30–44. 08 2020.

[7] Alexander Mey and Marco Loog. A soft-labeled self-training approach. pages 2604–2609, 12 2016.

[8] G. Leifert, R. Labahn, and J. A. Sánchez. Two semi-supervised training approaches for automated text recognition. In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 145–150, 2020.

[9] Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. *CoRR*, abs/1702.01806, 2017.

[10] Joan Andreu Sánchez, Verónica Romero, Alejandor H. Toselli, Mauricio Villegas, and Enrique Vidal. Dataset for ICDAR2017 Competition on Handwritten Text Recognition on the READ Dataset (ICDAR2017 HTR). July 2017.

[11] J. A. Sánchez, V. Romero, A. H. Toselli, and E. Vidal. Icfhr2014 competition on handwritten text recognition on transcriptorium datasets (htrts). In *2014 14th International Conference on Frontiers in Handwriting Recognition*, pages 785–790, 2014.

[12] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *CoRR*, abs/1609.07843, 2016.

[13] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *CoRR*, abs/1507.05717, 2015.