Iterativní zdokonalování přepisu nahrávek s využitím zpětné vazby posluchačů

Oldřich Krůza

Univerzita Karlova v Praze
Matematicko-fyzikální fakulta
Ústav formální a aplikované lingvistiky
Malostranské náměstí 25, Praha
kruza@ufal.mff.cuni.cz

Abstrakt Disertační práce, ke které se tato teze vztahuje, si klade za cíl vytvořit systém pro přepis mluvených korpusů do textu s využitím komunitní práce. Motivována je existencí a stavem souboru nahrávek Karla Makoně. Úmysl zpřístupnit veřejnosti jeho mluvené dílo je reálným podkladem mého výzkumného snažení. Technologie vyvinutá s tímto záměrem by však měla být použitelná i pro jiná data. Tato teze čtenáři představí zmíněný úkol, dosavadní postup podniknutý k jeho dosažení a plány pro budoucí práci.

1 Úvod

Moje práce má za účel umožnit zainteresovaným osobám co možná nejsnáze a nejúplněji zužitkovat poklad, který ve formě spisů a nahraných přednášek zanechal český mystik Karel Makoň.

Aby se moje práce mohla nazývat výzkumem, měla by úlohu řešit inovativním způsobem. Pro to je největší prostor v samotném způsobu řešení – v kombinaci automatického rozpoznávání mluvené řeči a manuálních oprav na webu. Aby užitek z mojí práce byl co největší, mělo by být možné nástroje vytvořené pro zpracování Makoňových nahrávek použít i na jiné sady dat podobné formy.

2 Dílo Karla Makoně

Psané dílo Karla Makoně je již leta volně k dispozici a některé jeho spisy vyšly knižně. Díky snaze a píli několika málo jedinců byly Makoňovy spisy nejdříve přepisovány na stroji a vydávány jako samizdat, nakonec byly přepsány do digitální formy a od té doby jsou volně ke stažení na internetu.

Situace s Makoňovými přednáškami je značně odlišná. Je zdigitalizováno asi tisíc hodin nahrávek, což představuje valnou většinu záznamů, jejichž existence je mi známa. Kvůli nekonzistentnímu a nedeskriptivnímu značení nahrávek je v podstatě nemožné se v archivu orientovat. I lidé, kteří byli osobně u většiny přednášek přítomni, jsou bezradní, mají-li dohledat záznam konkrétní přednášky nebo určité téma.

Tato situace je východiskem pro moji disertační práci. Podstatnou její součástí jsou zmínění lidé, kteří Makoňovy přednášky znají nebo se o ně zajímají, poslouchají jejich záznamy a osud díla jim není lhostejný.

3 Záměr disertace

Cílem je zpracování a zpřístupnění Makoňových nahrávek v nejširším smyslu. To je jednak velice rozsáhlý a jednak velice vágní úkol. Konkrétní úkoly, které jsou součástí tohoto širokého cíle, jsou zejména:

1. separovat úseky s jiným obsahem, než je hovor Karla Makoně,
2. nahrávky akusticky vyčistit,
3. umožnit vyhledávání v nahrávkách,
4. zajistit, aby osoby, které hledají informace o tématech, jež Makoň pokrývá, mohly jeho přednášky nalézt i bez vědomosti o jejich existenci,
5. vytvořit index podle témat, popřípadě podle jiných kritérií,
6. doplnit anotace k přednáškám

a mnohé další.

Bod 1 má jistý potenciál k automatizaci, ovšem není akutní a dá se provádět postupně bez časového omezení. Nebudeme se mu tedy zde věnovat.

Bod 2 taktéž není stěžejní, ale aspoň hrubé a částečné jeho dosažení může zvýšit jednak komfort posluchačů, jednak úspěšnost automatických metod zpracování. Jeho zevrubné řešení by byla úloha pro specialistu nebo tým.

Bod 3 je prvořadý a z vyjmenovaných nejdůležitější. Ohled na prohledavatelnost archivu je jedno z hlavních vodítek při rozvrhu práce.

Bod 4 je taktéž výsostně důležitý. Současná komunita kolem odkazu Karla Makoně čítá nefundovaným odhadem desítky až stovky lidí. Přitom témata, která jsou Makoněm adresována, jsou v okruhu zájmu mnoha tisíců. Prvotní motivace mojí práce je, aby z Makoňova díla byl co největší užitek – a tedy nejen pro ty, kteří ho už znají.

Bod 5 je výhledový, míněný pro další případný výzkum.

Bod 6 je specifický tím, že jeho splnitelnost je silně časově omezena. K přednáškám neexistují skoro žádná metadata, a při tom je někdy podstatné vědět, k jaké knize se daná slova vztahují nebo v jaké situaci byl člověk, k němuž Makoň promlouvá. Tyto znalosti mají pouze očití svědkové a vzhledem k jejich věku nelze spoléhat, že je budou moci poskytnout ještě za několik let.

Kdyby existoval kompletní kvalitní synchronizovaný[1] přepis archivu, automaticky by to řešilo stěžejní body 3 a 4 a značně ulehčilo řešení bodů 5 a 6. Naopak aspoň částečné splnění bodů 1 a 2 by mohlo ulehčit tvorbu přepisu.

Konkrétním cílem mojí disertace je tedy vytvoření přepisu celého archivu. Vzhledem k jeho rozsahu a omezeným finančním prostředkům je nereálné provádět úlohu manuálně. Na druhou stranu povaha dat (spontánnost řeči, rozsáhlá slovní

[1] ve smyslu, aby existoval *alignment* mezi textem a audiem

zásoba[2], kolísající zvuková kvalita atd.) ztěžují kvalitní automatický převod do textu.

Jestliže úlohu přepisu archivu nelze dost dobře provést ani manuálně ani automaticky a třetí alternativa nám není známa, zbývá se pokusit o kombinaci obou přístupů. Základní moje představa je ta, že za pomoci minimálního potřebného množství trénovacích dat se vytvoří automatickými metodami prvotní přepis korpusu. V tom bude mnoho chyb. Přepisy se zpřístupní i s nahrávkami přes webové rozhraní a posluchačům se zobrazí aktuální přepis synchronně se zvukem. Chyby, na které posluchač narazí, bude moci opravit. Tyto opravy se budou sbírat a používat ke zlepšení automatického přepisu. Tak se bude iterativně přepis zdokonalovat.

Obrázek 1 načrtává schéma aplikace. Zvuk s přepisem se prezentují uživateli, uživatel poskytne opravu části přepisu, oprava se uloží a přidá do trénovacích dat. Po nasbírání určitého množství oprav se model znova natrénuje a neopravené části přepisu se znova rozpoznají.

3.1 Přepisovací aplikace

První krok v postupu je vytvoření aplikace, která umožní poslech nahrávek se synchronním zobrazením přepisu a přepis opravit. Jako platformu jsem zvolil web kvůli jeho jasným výhodám: Je univerzálně dostupný na všech rozšířených architekturách, aplikaci není třeba instalovat a komunikace se serverem je snadná.

Nevýhody webu jsou též jasné: architektura protokolu HTTP je omezujícím faktorem. Nutnost přenášet celou aplikaci a potřebná data pokaždé přes internet mohou působit značné zpomalení. Vhodnost webu při uvážení charakteru dané aplikace je též diskutabilní.

Pro aplikaci, jejímž primárním účelem je práce s audiozáznamem, se web zdá být platformou méně vhodnou. Zajisté tomu před několika lety ještě bylo. I dnes by bezproblémový přístup k souborovému systému, ke zvukové kartě a vůbec k hardwaru práci ulehčil. Na druhou stranu právě odstínění hardwarových interakcí a přítomnost transparentní vrstvy pro přehrávání zvuku přítomné v HTML5 též nelze zanedbat.

V posledku jsou zmíněné nevýhody nepříliš podstatné: Práce s audiem je možná a nevyžaduje extrémní úsilí na straně programátora ani uživatele. Náročnost na datové přenosy není v dnešní době nadstandardní – to dokazuje i oblíbenost webových aplikací pro přehrávání nikoliv jen zvuku, nýbrž i videa. Případy, kdy se vyskytnou problémy, se často dají řešit volbou jiného internetového prohlížeče na straně uživatele a to je srovnatelná zátěž s instalací dedikované aplikace.

Práce bez stálého připojení k internetu je zajisté možná, ale implementace by vyžadovala nemalé úsilí. I potom by zde bylo omezení kvótami pro lokální ukládání dat internetovými prohlížeči.

Přepisovací aplikace má mít, jak zmíněno výše, tři základní funkcionality:

[2] Slovní zásoba je relativně málo rozsáhlá vzhledem k tomu, že hovoří jeden mluvčí v tematicky omezené doméně. Přesto však vzhledem k vlastnostem češtiny je rozsáhlá natolik, že představuje problém.

Obrázek 1. Schéma běhu aplikace

– přehrání zvolené nahrávky,
– synchronní zobrazení přepisu
– a editaci přepisu.

Přehrávání Přehrávání nahrávek je delegováno na knihovnu jPlayer, která tuto funkcionalitu v internetových prohlížečích zpřístupňuje jednoduchým interfacem, který využívá tagu $< audio >$ ze standardu HTML5, a v případě jeho nedostupnosti nebo nepoužitelnosti se uchýlí k Flashi. Samotná zvuková data jsou uložena na externí CDN[3]. Soubory nebyly nijak štěpeny nebo slepovány po digitalizaci, čili jeden soubor odpovídá až na výjimky jedné straně kazety nebo jednomu převinutí kotouče. Běžná délka je tedy 45 až 90 minut.

Zobrazování přepisu Pro vybrání způsobu zobrazení synchronního přepisu bylo nutné zvážit několik faktorů, především ergonomii, výpočetní náročnost a náročnost na vývoj. V úvahu připadaly tři možnosti:

1. titulky jako u filmu,
2. běžící text jako u HTML tagu $< marquee >$
3. a zobrazení několika řádků, které by se posouvaly vertikálně.

Formát titulků se jeví velmi vhodný. Za prvé je člověk uvyklý takový formát sledovat. Za druhé je velice jednoduchý na implementaci a výpočetní náročnost je mizivá. Nedostatky vidím dva. Jednak je potřeba vhodně volit hranice mezi jednotlivými titulky, a to automatickými metodami nemusí být snadné. Jednak pro slova na začátku a na konci titulku chybí kontext, takže úsek, který přesahuje hranici titulku, by bylo obtížné opravit.

Formát běžícího textu, kde uprostřed je aktuálně vyřčené slovo, jsem zavrhl velmi rychle z důvodů ergonomických. Eliminuje sice problém kontextu, jímž trpí formát titulků, ale neustálý pohyb celého textu se může stát nepříjemným. Navíc by to byl pohyb o proměnlivé rychlosti, podle toho, jak rychle plynou slova v řeči. Pohyb by dokonce byl skokový, nikoliv plynulý, pokud by se text posouval při přechodu na další slovo.

Třetí varianta způsobu zobrazení spočívá v tom, že na monitoru je přítomen konstantní počet řádků textu. Aktuálně vyslovené slovo se nachází v prostředním řádku a jakmile se dojde na konec řádku, řádky se posunou o jeden kupředu. Tento způsob zobrazení působí alespoň tak přirozeným dojmem jako titulky. Je totiž podobný čtení statického textu a to je rozhodně běžnější případ než četba titulků. Navíc skýtá vždy dostatečný kontext pro aktuální slovo.

Vezmeme-li v potaz, že je potřeba zvýrazňovat aktuální slovo a tedy urdžovat informaci o přesném čase, kterému každé slovo odpovídá, vynořují se aspekty programátorské a výpočetní složitosti. Pokud bych věnoval každému slovu jeden HTML element, byla by věc snadná na naprogramování, ovšem při větším počtu zobrazených řádků by vykreslení mohlo trvat neúměrně dlouho a tím by trpěla odezva prostředí. Naopak při zobrazení textu jen tak bez obalování každého slova

[3] Content-delivery network

zvláštním tagem by rychlost vykreslení nepředstavovala problém. Zato by nebylo snadné naprogramovat zvýrazňování aktuálního slova a jednoznačně identifikovat časové rozpětí záznamu na základě označení části přepisu.

Nakonec jsem zvolil víceřádkové zobrazení, kde každé slovo má vlastní HTML element. Problém s rychlostí vykreslování řeším tím, že zobrazuji jen tři řádky. Běžné počítače se s tím vypořádávají s uspokojivou rychlostí, kontext je dostačující a dokonce to má výhodu, že nikdy netrvá dlouho okem nalézt vyznačené slovo.

Editace přepisu Aplikace je navržena tak, aby umožňovala pohodlné *opravování* přepisu záznamu, nikoliv pohodlné přepisování od začátku. Zamýšlený postup práce je 1) poslech zároveň se sledováním přepisu, 2) spatření chyby v přepisu, 3) označení chybné části (s případným přesahem) a její oprava.

V tomto schématu je ukryt předpoklad, že u jednotlivých slov přepisu je zřejmé, kterým vyřčeným slovům odpovídají.

Aplikace tedy přímočaře implementuje tento postup: Při označení části přepisu se vstoupí do editačního módu, identifikuje se výsek zvukového záznamu, jenž odpovídá označenému textu a uživateli se umožní jednak do textového pole vložit správný přepis, jednak opakovaně přehrát daný úsek, a jednak posouvat hranice úseku.

Jakmile uživatel potvrdí správnost vloženého přepisu, tento se odešle na server zároveň s údajem o pozici začátku a konce zvukového úseku. Aktuální akustický model se použije pro *forced alignment*, čímž se v případě neúspěchu identifikuje potenciálně nesprávný přepis[4] a v případě úspěchu se každému slovu přiřadí přesná časová pozice. Teprve poté server pošle odpověď zpět na klienta, který pak může nový přepis obohacený o metadata slít se zbytkem přepisu.

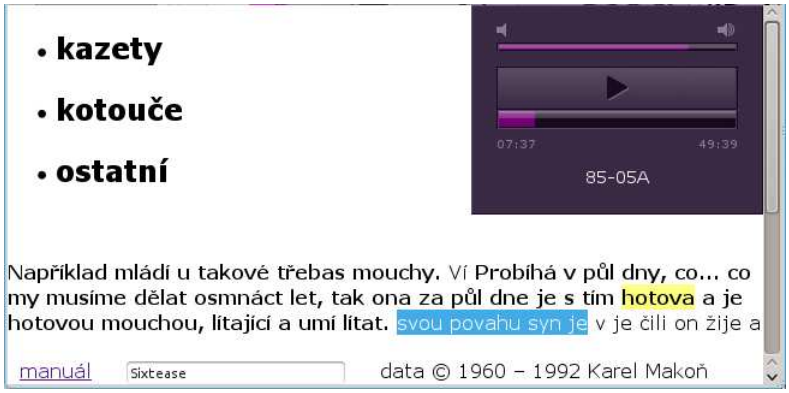
Tato funkcionalita jednak aplikaci velice obohacuje, obzvlášť ve srovnání s ostatními přepisovacími nástroji, jednak značně zvyšuje nároky na server. Kvůli nutnosti spouštět forced alignment, tedy vlastně rozpoznávač řeči, je prakticky nevyhnutelné mít vlastní server (aspoň virtuální) a nestačí běžný hosting.

Dalším velkým nárokem na server je uložení celého korpusu. Pro forced alignment je totiž nezbytně nutné, aby se dalo velmi rychle přistoupit k přepisovanému úseku. Stahování z externí CDN je prakticky vyloučené. Stačí však mít audio na serveru již parametrizované – v mém případě MFCC soubory, dokonce bez derivací a dalších dopočitatelných údajů. Nárok na úložné místo na serveru se tak podstatně snižuje.

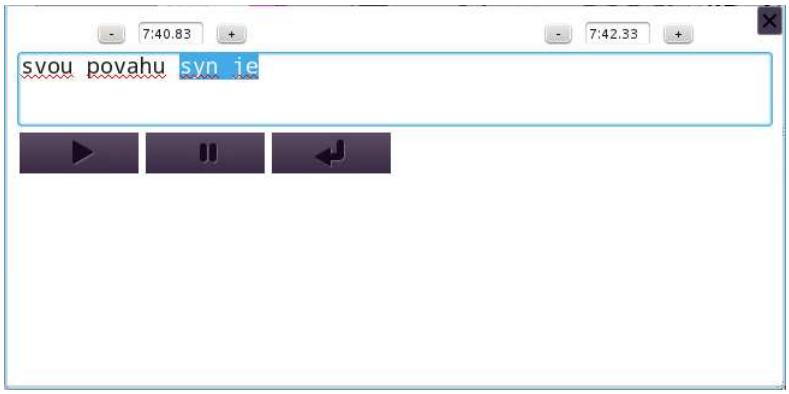
Aplikace je v popsaných rysech hotová. Dá se samozřejmě nadále vylepšovat, ale byly na ní přepsány již přes tři hodiny materiálu lidmi s minimem zaškolení. Bohužel předpoklad, že u jednotlivých slov přepisu je zřejmé, kterým vyřčeným slovům odpovídají, není se současnou kvalitou automatického rozpoznávání vždy splněn. Schéma práce se od zamýšleného liší tím, že chybná je většina slov, a proto se místo o opravu chyb vlastně jedná o kompletní přepisování. Efektivita práce tím trpí.

[4] Spolehlivost této funkcionality velice závisí na kvalitě akustického modelu a na délce úseku – čím delší, tím spíše se přijme nesprávný přepis.

Na obrázcích 2 a 3 jsou vidět screenshoty přepisovací aplikace – jednou v normálním módu, podruhé v editačním.



Obrázek 2. Přepisovací aplikace v normálním módu



Obrázek 3. Přepisovací aplikace v editačnám módu

3.2 Prvotní přepis

Druhým z prvních kroků celého postupu je získat pomocí automatického rozpoznávání řeči prvotní přepis korpusu. Že získat automaticky přepis mluvené češtiny je možné, avšak nelehké, dokládá dosavadní výzkum, např. Psutka, Hajič a Byrne 2004[1] v projektu Malach[2]. Pro první nástřel jsem vytvořil trénovací skript podle návodu dr. Peterka, který vychází z HTK Book[3] a pro didaktické účely se omezuje na jednogaussiánové monofonémy. Akustický model jsem

natrénoval na asi šesti minutách manuálního přepisu, který jsem sám udělal. Jazykový model jsem natrénoval na Makoňových psaných textech, které mají rozsah asi 26 MB čistého textu. Fonetickou abecedu jsem použil od Nouzy, Psutky a Uhlíře (1997)[4], vycházející z české fonologie, jak ji popsala Palková (1992)[5].

Jazykový model byl bigramový kvůli omezení použitého rozpoznávacího programu HVite. Pro parametrizaci zvuku byl použit formát MFCC[5] s budícím signálem, první a druhou derivací, normalizovaný. Velikost rozpoznávacího slovníku 20000 slov. Úspěšnost tohoto nastavení byla mizivá (word precision pod pět procent).

Následuje výčet experimentů podniknutých pro vylepšení jednak jazykového a jednak akustického modelu.

Jazykový model Pohled na výstup rozpoznávače odhalil přítomnost mnoha specifických slov, obzvláště vlastních jmen. Tato slova by měla být velmi nepravděpodobná v běžném textu a vůbec jejich přítomnost v rozpoznávacím slovníku dané velikosti je pochybná. Jednalo se ku příkladu o jména ,,Lao“, ,,Tchaj“ a podobná. Jak se do výstupu dostala bylo na snadě: jazykový model byl natrénovaný na Makoňových knihách – právě v naději, že jeho vlastní slovní zásoba úspěšnosti pomůže. Když se některý spis zabýval intenzivně tím nebo oním východním mudrcem, četnost jeho jména (mnohdy navíc nesklonného, což ještě přispělo k počtu výskytů jeho jediné formy) ho zařadila mezi velice silné prvky v jazykovém modelu. Navíc jejich jednoslabičnost přispěla k tomu, že mnohé akustické kontexty takovým slovům nebyly příliš vzdálené.

Pokusil jsem se tedy potlačit přítomnost těchto slov. Rozdělil jsem data pro trénink jazykového modelu na tři části a slova, která se často vyskytovala jen v jedné z nich, jsem vyloučil z rozpoznávacího slovníku. Jev pominul, úspěšnost nezaznamenala zásadní změny.

V tomto nastavení jsem vyzkoušel ještě jazykový model natrénovaný na datech z Pražského závislostního korpusu[6]. Z vyzkoušených variant se ukázal nejlepší jazykový model natrénovaný na Makoňových knihách.

Další veličinou, která souvisí s jazykovým modelováním a která má zásadní dopad na úspěšnost, je velikost slovníku. Její optimum pro dané ostatní nastavení jsem získal natrénováním na heldout datech. Zkonvergovala na 1700 slovech.

Akustický model Variace jazykového modelu neměly zásadní vliv na úspěšnost rozpoznávání. Rozšíření trénovacích dat na dvě a poté tři hodiny úspěšnost zvýšily, ale zdaleka ne podle očekávání (WP stále kolem pěti procent). Dospěl jsem k domněnce, že v trénovacím řetězci akustického modelu se vyskytuje nějaká kritická chyba. Srovnal jsem tedy svůj skript znova s postupem uvedeným v HTK Book.

Jeden ze zásadních rozdílů mezi mým postupem a postupem doporučeným byl v nakládání s trifonémy. Trénování trifonémů, čili hlásek ovlivněných kontextem zprava i zleva, vede k tomu, že v testovacích datech se mnohdy vyskytne

[5] mel-frequency cepstral coefficients

některý foném v takovém kontextu, v jakém se v trénovacích datech nikdy nevyskytl. To je typický případ řídkosti dat, vpravdě moru statistických metod.

Běžný postup řešení takové situace spočívá v tom, že se na základě lingvisticky motivovaných rysů všechny trifonémy rozdělí do shluků a při setkání s neznámým trifonémem se s ním nakládá jako se zástupcem shluku, do kterého náleží.

Když jsem prvně implementoval trifonémy, tento postup mi nebyl znám a tak jsem vyvinul vlastní, odlišnou metodu. Ta spočívá v tom, že se spočte, kolikrát se který trifoném vyskytuje v trénovacích datech, a použije se sada polyfonémů, které se vyskytují aspoň N-krát, kde N je nastavitelná přirozená konstanta. Já jsem zvolil N=3. Když se vyskytne jiný trifoném, nahradí se bifonémem a v případě, že oba bifonémy (zleva i zprava) jsou též mimo danou sadu polyfonémů, nahradí se monofonémem. Tento postup je aplikován při generování polyfonémového přepisu jak trénovacích tak testovacích dat. Tím je garantováno, že ke každému polyfonému existuje aspoň N trénovacích příkladů[6] za cenu toho, že některé fonémy pozbudou kontext.

Rozdíl v těchto přístupech je, že se méně důvěry vkládá do shlukování fonémů a více do robustnosti kontextově nezávislých modelů. Dá se předpokládat, že s výbornou sadou rysů pro shlukování a s dostatkem trénovacích dat doporučená metoda bude fungovat lépe. V mém případě, kdy dat je poskrovnu a rysy jsem nadefinoval po krátké rešerši sám, se ukázala moje metoda lepší.

Ještě podotknu, že ač shlukování trifonémů nepoužívám pro ošetření neznámých trifonémů, přesto je používám pro zvýšení robustnosti modelů.

K prvnímu znatelnému zvýšení úspěšnosti došlo po použití tzv. mixtur, čili modelu, kdy každý foném není reprezentovám jedním gaussiánem, nýbrž několika.

Mixtury se získávají štěpením modelů. Je-li natrénovaný model o N mixturách pro daný foném, vytvoří se nový model o M mixturách, kde $M > N$, obvykle $M = N + 1$. Nové gaussiány jsou identické tomu, ze kterého byly odvozeny. Několika trénovacími iteracemi se pak od sebe vzdálí, odhaduje-li to data lépe.

Tento proces lze činit pro každý model (model každého fonému) zvlášť nebo pro všechny najednou. Dá se předpokládat, že jemnějším prohledáváním, kdy se štěpí jen fonémy, u nichž to přinese zlepšení, musí vést k lepšímu výsledku. Tato intuice je ovšem lichá, aspoň v mém experimentu štěpení všech modelů najednou vedlo k mnohem lepšímu výsledku, a taky podstatně rychleji.

Ve většině experimentů bylo lokální optimum nalezeno u osmi mixtur. Doporučuje se též modelům pro ticho přidělit paušálně dvojnásobek mixtur oproti ostatním fonémům. To jsem zatím nevyzkoušel.

Dr. Jurčíček nalezl skript pro trénování akustického modelu na mluveném Wall Street Journalu, který také v hrubých rysech sleduje postup popsaný v HTK Book, ale v mnoha bodech se od něho odchyluje. Na svých datech s tímto skriptem dosáhl větší úspěšnosti, než s mým skriptem – asi 80% word precision oproti asi 60.

[6] Kromě případu, kdy nějaký monofoném by byl natolik řídký, že by se nevyskytl v datech ani N-krát.

Porovnání skriptu pro WSJ a mého se tedy nabízí jako zdroj námětů pro zlepšení. Podrobné porovnání jsem ještě neukončil. Detailů, v nichž se skripty liší, jsem již nalezl mnoho. Žádný však nezlepšil výkon na mých datech – naopak, pokud došlo ke změně, pak k horšímu.

Do nynějška jsem nasbíral asi tři a půl hodiny trénovacích dat. Moje původní domněnka, že výstup rozpoznávače bude natolik dobrý, aby se v něm jen ,,opravovaly chyby" a nemusel se materiál přepisovat celý, a že k tomu postačí takové množství trénovacích dat, kolik budu schopen přepsat sám, se tedy ukázala naivní. Momentální odhad podložený konzultacemi s odborníky (dr. Peterkem a dr. Motlíčkem) je, že budu potřebovat přibližně osm až dvanáct hodin trénovacích dat na vygenerování použitelného prvotního přepisu[7], jenž se bude moci nadále zdokonalovat podle načrtnutého schématu.

Výsledky Tabulka 1 ukazuje, jak se jednotlivé experimenty odrazily na úspěšnosti. Současný stav není současné optimum – adaptace mého skriptu tak, aby kopíroval recept pro WSJ není dokončena a některé zásahy způsobily lokální pokles úspěšnosti.

,,Jazykový model bez biasu" je ten, u kterého ze slovníku byla odstraněna slova vyskytující se často jen v jedné části. Oba výsledky týkající se experimentů s jazykovým modelem byly získány aplikací toho kterého jazykového modelu na současný akustický model. V době, kdy jsem tyto experimenty dělal, to jest před experimenty s akustickým modelem, byly výsledky horší.

Taktéž model ,,bez mixtur" je současný stav před začátkem štěpení gaussiánů, nikoliv jako dříve, než jsem vůbec s mixturami začal experimentovat. Všechny experimenty tedy ukazuji na současném modelu s co nejmenšími změnami. Je tak lépe vidět, jaký přímý dopad ten který experiment má. Hůře je pak vidět vývoj, jak se jevil v průběhu času mně. Všechny experimenty kromě ,,bez mixtur" používají osm mixtur – a to ne jako zvolenou konstantu, nýbrž jako počet mixtur, kde se vyskytuje první nebo druhé lokální optimum.

Pro úplnost zmíním, že ,,word precision" je podíl správně rozpoznaných slov vůči počtu slov v datech zlatého standardu. Accuracy je podíl, kde v čitateli je počet správně rozpoznaných slov minus počet ,,insertions" (slov ve výstupu, která nemají protějšek ve zlatém standardu) a ve jmenovateli opět počet slov ve zlatém standardu.

Tyto metriky jsou v rozpoznávání mluvené řeči zavedené, ačkoliv osobně bych za šťastnější považoval používání standardních ,,precision" a ,,recall". Precision by v tomto případě byla počet zásahů dělený počtem slov ve výstupu a recall by byl počet zásahů dělený počtem slov ve zlatém standardu (čili to, co se zde nazývá ,,word precision"). To, že tyto metriky chybí, je následkem pouze toho, že jsem se k jejich použití zatím nedostal – standardní metriky dostávám automaticky.

[7] Zmíněný objem trénovacích dat má být nutnou, nikoliv postačující podmínkou.

Experiment	Word precision	Accuracy
Současný stav	27.14%	-7.11
Jazykový model bez biasu	27.20%	-7.55
Jazykový model natrénovaný na datech z PDT	26.10%	-36.15
Před změnami podle WSJ	23.76%	-9.11
Standardní trifonémy	25.21%	-9.26
Bez mixtur	4.64%	-1.19

Tabulka 1. Úspěšnost rozpoznávání s různými experimenty.

4 Plán prací

4.1 Zdokonalení prvotního přepisu

Největším současným problémem je nízká kvalita prvotního přepisu. Na ni se upírá moje momentální úsilí a plány na nejbližší práci. Především sbírám další trénovací data. Krom toho je nabíledni vyzkoušení pokročilejších rozpoznávacích programů, než je HVite. Julius, Kaldi a HDecode všechny přicházejí v potaz, protože narozdíl od HVite podporují i jiné než bigramové jazykové modely a velké slovníky.

Z porovnávání mého trénovacího skriptu s tím pro Wall Street Journal si slibuji dosáhnout na datech z projektu Vystadial stejných nebo lepších výsledků. Pak budu mít jistotu, že v mém skriptu není závažných chyb.

Dr. Klusáček též vyvinul zajímavé inovace v rozpoznávání řeči. Plánujeme vyzkoušet jeho metodu redukce ozvěny na mých datech – to by mohlo pomoci, protože ozvěna je tam často znatelná.

Velice lákavé by bylo použít aposteriorní rysy při parametrizaci zvuku, jak to popisují Bourlard, Nelson (1994)[7]. Na to by ale bylo potřeba jednak zcela přepracovat postup trénování a jednak získat neuronovou síť pro odhadování aposteriorních rysů. Obojí je přinejmenším náročné.

Díky tomu, že trénovací data se získávají postupně a nejsou dána předem, mohu ovlilvnit, které nahrávky se budou přepisovat – do té míry, jak si uživatelé nechají nahrávku „doporučit". To se dá využít pro tzv. aktivní trénink, jak ho popsali Hakkani-Tür a Riccardi (2011)[8]. To jest, mohu na základě *confidence measure* určit, které nahrávky se rozpoznávají nejhůře a podle toho vybrat takové, jejichž přepis pravděpodobně nejvíce přispěje ke zlepšení úspěšnosti.

Přepisovací aplikace bude možná použita pro přepsání dalších několika hodin materiálu v současném neuspokojivém stavu automatického rozpoznání. Proto je otevřená alternativa zoptimalizovat ji pro přepis od nuly. Zde by bylo vhodné inspirovat se existujícími aplikacemi, např. *Transcriberem*. Uživatel by úsek pro přepis nedefinoval označením textu, nýbrž pozastavením přehrávání. Úsek by pak byl stanoven od konce posledního přepsaného úseku do pozice, kde bylo přehrávání pozastaveno, a opět by se přešlo do editačního módu. Tam by bylo samozřejmě jako nyní možné hranice úseku posouvat.

4.2 Zpracování příspěvků

Až se podaří uvést prvotní přepis na uspokojivou úroveň, uživatelé budou moci text podle původního plánu opravovat. Pokud bude příspěvků dost na to, aby nebylo reálné kontrolovat je manuálně, bude hlavní část mé práce spočívat v tom, jak příspěvky co nejlépe zpracovávat.

Přestože jsem se touto částí doposud dopodrobna nezabýval, jsou některé možnosti jasné již nyní. Triviální využití příspěvků je přidat je do trénovacích dat. To činím teď a dost možná je to nejlepší věc, která se dá vůbec udělat. Není to ale jisté. Příspěvky mají být opravami chyb. To je něco jiného než nový přepsaný úsek. Opravy chyb postihují místa, kde model neuspěl, a dá se proto předpokládat, že když se s nimi bude zacházet jinak než s ostatními trénovacími daty, bude možné z nich více vytěžit. Jak konkrétně by se to mělo dít, bude předmětem další rešerše... až věc bude aktuální, nejpozději však v průběhu čtvrtého roku.

Dalším problémem je kvalita příspěvků. Už nyní narážím na to, že v příspěvcích jsou chyby. Jedna z věcí, která může pomoci, a kterou do jisté míry činím už teď, je automatická normalizace. Týká se to v současnosti především interpunkce, která je při rozpoznávání tak jako tak ignorována, takže vliv na úspěšnost je vyloučen.

Jednotliví přispěvatelé mají různé osobní styly a zvyklosti. Kupříkladu někdo striktně odděluje věty tečkami a velkými písmeny, zatímco někdo nikoliv. Nabízí se tedy myšlenka použít pro příspěvky od různých lidí různé normalizační metody. Jak by se taková technologie měla vyvíjet a do jaké míry by měla být automatizovaná, je zatím nejasné.

Další možností je zavedení lidských arbitrů, kteří by místo přepisování kontrolovali příspěvky ostatních. Tím by se obětovala kvantita kvalitě.

Velkou pomocí je zde forced alignment, který odhalí mnohé chyby a automaticky je odmítne a navíc okamžitě přispěvatele upozorní tím, že jeho příspěvek není přijat. Čím bude rozpoznávací model dokonalejší, tím lépe bude tento mechanismus moci fungovat.

4.3 Nasazení na jiná data

Jedním z bodů mého disertačního zadání je nasazení systému na jiný korpus než na Makoně. Systém je vyvíjen Makoňovu korpusu na míru, ovšem nástroje jsou samozřejmě vůči datům agnostické. Při nasazení na jiný korpus je především potřeba dodat patřičný systém pro rozpoznávání řeči a forced alignment. Jeho trénování se pak musí řešit pro každý korpus zvlášť – dá se na to dívat jako na povinný plug-in celého systému. Pak zbývá jen nakonfigurovat cesty k datům a spustit aplikaci na nějakém serveru. Doposud jsem se o to však pro jiná data nepokusil.

Z konkrétních korpusů, kde nasazení přichází v úvahu, mohu jmenovat korpus Dialogy a dále záznamy z přednášek profesora Patočky a doc. Zdeňka Pince.

4.4 Plán B

Můj postup na disertační práci je krátce před rozcestím: Pokud se po podniknutí zmíněných plánovaných kroků ukáže, že kvalita automatického rozpoznávání je stále hluboko pod hranicí použitelnosti, budu muset změnit plán. V takovém případě se zaměřím na výzkum možností, jak splnit stěžejní cíle bez získání kompletního kvalitního přepisu.

Stěžejní cíle budou splněny, pokud půjde v korpusu vyhledávat. S úspěšností kolem šedesáti procent WP se mohu zaměřit na nasazení vhodných vyhledávacích nebo rešeršních algoritmů.

Ještě další možností by bylo zaměřit se na vyvinutí metody pro efektivní přepis lidmi. Tato varianta se překrývá se zmíněným plánem na konci sekce 4.1. Na rozdíl od ostatních možností je tato bezpochyby schůdná a nese jen málo rizik (snad jen to, že se nenajde dost lidí, kteří by přepisy prováděli, ovšem to neovlivňuje možnost takovou technologii vyvinout).

4.5 Časový odhad

Dostat úspěšnost automatického rozpoznávání na použitelnou úroveň pro získání prvotního přepisu chci zvládnout během tohoto akademického roku. To je mezník pro přistoupení nebo nepřistoupení k plánu B. Odladění trénovacího skriptu podle vzoru receptu pro Wall Street Journal nebude trvat déle než šest týdnů. Pro vyzkoušení dalších rozpoznávacích programů odhaduji maximálně měsíc. Otevře to ale prostor pro další experimenty s pokročilejšími jazykovými modely – další měsíc.

Míra dosažené úspěšnosti mi bude vodítkem pro další postup. Pokud dosáhnu úspěšnosti nad sedmdesát procent WP, budu pokračovat podle plánu. Pokud bude úspěšnost nad padesát procent, začnu zkoumat možnosti vyhledávání a rešeršních algoritmů nad výstupem rozpoznávání. Pokud bude úspěšnost nižší, budu se soustředit na možnosti kolaborativního lidského přepisu a jeho podpory.

Ať se uskuteční kterákoliv varianta, výzkum bych chtěl ukončit po čtvrtém roce studia. Konkrétní cíle se pak budou lišit podle toho, ke které variantě se po tomto roce výzkum stočí.

5 Závěr

Postup na mojí disertaci dosud vedl k pozitivnímu praktickému výsledku vyvinutí webové přepisovací aplikace. Získání automatického přepisu použitelné kvality se prokázalo jako úkol nad očekávání složitý. Úspěšnost v jeho řešení bude rozhodujícím prvkem pro další postup.

Reference

1. J Psutka, J Hajic, W Byrne *The development of ASR for Slavic languages in the MALACH project* in Proc. ICASSP 2004
http://svr-www.eng.cam.ac.uk/ wjb31/ppubs/icassp04-malach-final.pdf

2. W Byrne, D Doermann, M Franz, S Gustman, J Hajic, D Oard, M Picheny, J Psutka, B Ramabhadran, D Soergel, T Ward, Wei-Jing Zhu *Automatic recognition of spontaneous speech for access to multilingual oral history archives* in Proc. Eurospeech 2003, Geneva, Switzerland http://www.ee.umd.edu/ oard/pdf/tsap04.pdf
3. S Young, G Evermann, M Gales, T Hain, D Kershaw, X A Liu, G Moore, J Odell, D Ollason, D Povey, V Valtchev, P Woodland *The HTK Book* 2006 http://htk.eng.cam.ac.uk/
4. J Nouza, J Psutka, J Uhlíř *Phonetic alphabet for speech recognition of Czech* Radioengineering Vol.6, pp. 16-20, 1997 http://www.radioeng.cz/fulltexts/1997/97_04_04.pdf
5. Z Palková *Fonetika a fonologie Češtiny* Univerzita Karlova, Praha 1992
6. A Böhmová, J Hajič, E Hajičová, B Hladká *The Prague Dependency Treebank: A Three-Level Annotation Scenario* 2007 http://www.scientificcommons.org/43211198
7. H Bourlard, N Morgan *Connectionist speech recognition: a hybrid approach* Vol. 247. Springer, 1994.
8. D Hakkani-Tür, G Riccardi *Active Learning in Spoken Language Understanding: Systems for Extracting Semantic Information from Speech* (eds G. Tur and R. De Mori), John Wiley & Sons, Ltd, Chichester, UK. doi: 10.1002/9781119992691.ch8