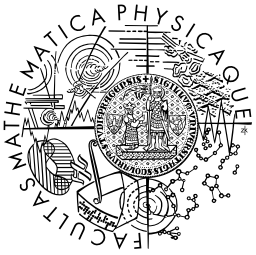

MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Jiří Balhar

Extrakce melodie pomocí hlubokého učení

Ústav formální a aplikované lingvistiky

Vedoucí bakalářské práce:	Mgr. Jan Hajič
Studijní program:	Informatika
Studijní obor:	Programování a softwarové systémy

Praha 2019

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ..... Podpis autora

Poděkování.

Název práce: Extrakce melodie pomocí hlubokého učení

Autor: Jiří Balhar

Ústav: Ústav formální a aplikované lingvistiky

Vedoucí bakalářské práce: Mgr. Jan Hajič, ústav

Abstrakt: Abstrakt.

Klíčová slova: klíčová slova

Title: Melody Extraction using Deep Learning

Author: Jiří Balhar

Institute: Institute of Formal and Applied Linguistics

Supervisor: Mgr. Jan Hajič, institute

Abstract: Abstract.

Keywords: key words

Obsah

Úvod

Spolu s harmonií a rytmem představuje melodie základní kámen většiny známé hudby. V průběhu vývoje od folklórních zpěvů přes orchestrální skladby po soudobou elektroniku si melodie téměř vždy zachovávala své dominantní postavení nositele esence jednotlivých písní. Melodie je to hlavní, co si člověk po poslechu skladby odnáší a nejsnadněji vybaví, a její důležitost je zejména v našem kulturním kontextu natolik jednoznačná, že je občas těžké si bez ní vůbec hudbu představit.

Tato práce se zabývá metodami přepisu melodických kontur ze zvukové nahrávky. Jde o jednu z nejdůležitějších a zároveň nejtěžších úloh z oboru *Music Information Retrieval*, jejíž rozsah využití v této doméně pokrývá významnou část aktivně řešených, otevřených problémů. Spolehlivý přepis melodie by usnadnil vyhledávání v hudebních datech, ať už na základě notového zápisu (*Symbolic Melodic Similarity*), pomocí nekvalitní nahrávky z rádia (*Audio Fingerprinting*), pomocí broukání (*Query by Singing/Humming*) nebo dokonce pomocí coveru hledané písně (*Audio Cover Song Identification*). Mimo vyhledávání by byl algoritmus užitečný pro další zpracování zvukového signálu, ať už pro manipulaci a úpravu melodického hlasu (například software Melodyne), nebo naopak jeho odstranění a vytvoření karaoke doprovodu (*Informed Source Separation*). V neposlední řadě by extrakce melodie pomohla při kategorizaci hudebních dat, například podle žánru (Salamon a kol. (2012)) nebo podle zpěváka (*Singer Characterization*). A konečně široké spektrum využití by se nalezlo i v muzikologii (případně etnomuzikologii) pro statistickou i kvalitativní studii hudebních motivů a postupů (V jazzu například Pfleiderer a kol.).

Příkladem použití ale může být i pomoc při transkripci. Představíme-li si začínajícího hráče na saxofon, který si chce do not přepsat svoje oblíbené jazzové sólo, aby se ho mohl naučit, výstup algoritmu pro přepis melodie mu dá užitečnou informaci o tom, jaký tón zní v jakou chvíli. Z této reprezentace už pak hráči zbývá nalezené tóny projít a zapsat je do notové osnovy ve správných délkách.

1. Související práce

1.1 Definice melodie

Jelikož z muzikologického hlediska žádná jasná a obecně přijímaná definice melodie neexistuje a ve výsledku melodie zůstává pro každého posluchače ryze subjektivním pojmem, pro extrakci melodie si výzkumné týmy volí spíše pragmaticky takové definice, se kterými se nejlépe pracuje. Příkladem může být jedna z prvních prací zabývající se extrakcí melodie. Výstupem v práci Goto a Hayamizu (1999) je kontura fundamentální frekvence sestávající se z nejsilnějších tónů hrajících v omezeném frekvenčním rozsahu. Tato definice je poměrně úzká, tóny melodie se totiž jistě mohou vyskytovat i mimo autory specifikovaný rozsah a nemusí být vždy v poměru s doprovodem nejsilnější složkou signálu. Z technického hlediska však umožnila autorům implementaci algoritmu běžícího v reálném čase, který poskytoval sémanticky bohatý popis vstupních nahrávek. Navazující články pracují s volnějšími definicemi, které lépe reflektují podstatu melodie. Mimo to se používaná definice proměňuje díky novým datasetům, jejichž autoři tvoří protipól k ryze technickým a objektivním cílům algoritmických metod. Zatímco pro tvorbu algoritmů je praktické zvolit co nejkonkrétnější cíl, při tvorbě datasetu se naopak projevuje lidská subjektivita autorů anotací.

Kompromisem mezi subjektivní a praktickou definicí se na dlouhou dobu stala „extrakce základní frekvence hlavního melodického hlasu“. Ačkoli melodii v reálném hudebním materiálu obvykle nese více hlasů, které se v hraní střídají (například píseň se zpěvem a kytarovým sólem), v letech 2005 – 2015 se v soutěži MIREX provádí evaluace pouze nad krátkými výňatky, kde tato definice není omezující. Tento pohled však otevírá také jiné přístupy, například extrakci melodie pomocí modelování hudebního záznamu jako součtu signálu jednoho hlasu a doprovodu (Durrieu a David, 2010), (Bosch a Gómez, 2016) nebo přímo omezení se na separaci lidského zpěvu a doprovodu (Ikemiya a kol., 2016). Nově se objevují práce, které „hlavní“ melodický hlas neinterpretují nutně jako „nejsilnější“. Skladatelé a hráči používají množství různých postupů, které melodii zvýrazňují — krom dynamiky ji ovlivňuje například také barva hlasu, vibrato nebo délka not. Salamon a Gomez (2012) využívá těchto rysů pro výběr mezi kandidáty na melodickou konturu.

Posunem v rámci MIR komunity bylo zveřejnění nových datasetů MedleyDB (Bittner a kol., 2014) a ORCHSET (Bosch a kol., 2016), oba přináší nová data, ve kterých již melodii nenese pouze jeden hlas po celou dobu skladby. V porovnání s do té doby dostupnými daty jde o mnohem rozmanitější kolekce. V případě MedleyDB jde o první volně dostupný dataset, ve kterém se objevují celé skladby, nikoli pouze výňatky a autoři předkládají rovnou tři verze anotací:

1. Základní frekvence nejvýraznějšího melodického hlasu, jehož zdroj zůstává po dobu nahrávky neměnný.
2. Základní frekvence nejvýraznějšího melodického hlasu, jehož zdroje se mohou měnit.
3. Základní frekvence všech melodických hlasů, potenciálně pocházejících z více zdrojů.

První formulace je v souladu s doposud používanou definicí. Zbylé dvě se snaží posouvat možné cíle budoucích metod a předložit komunitě nové výzvy, podle Salamon a kol. (2014) totiž výzkum začal v letech 2009–2012 stagnovat. Zatímco anotace s jednou melodickou linkou (1. a 2. definice) se v navazujících pracích často používají, zatím žádný článek se nepokusil představit metodu, jejímž cílem by bylo extrahovat více melodických linek (3. definice).

Bosch a kol. (2016) při práci na datasetu ORCHSET vychází z článku Poliner a kol. (2007), který definuje melodii jako „jednohlasou sekvenci tónů, kterou bude posluchač nejspíše reprodukovat, pokud jej požádáme o zapískání či zabroukání příslušné skladby“. Přestože nejde o objektivní definici, v praxi se posluchači často na jedné konkrétní sekvenci tónů shodnou, a to jak u populární hudby, kde melodii často nese lidský zpěv, tak u orchestrálních skladeb. Ačkoli se definice neujala pro metody extrakce, Bosch a kol. (2016) ji využili pro anotaci výňatků z orchestrálních skladeb, u kterých by předchozí zmíněné definice selhávaly, jelikož pojem melodie je u orchestrální hudby mnohdy komplikovanější než u jiných žánrů. Anotace tak spočívala v přezpívání orchestrálních výňatků skupinou posluchačů a následném srovnání a zpracování těchto nahrávek.

1.2 Průzkum existujících metod

Jen do soutěže MIREX se od roku 2005 přihlásilo 45 týmů s 62 různými metodami pro extrakci melodie, s různou mírou přesnosti přepisu. Mezi přístupy k tomuto problému tedy existuje veliká rozmanitost, jejíž kompletní popis přesahuje rámec této práce. Zaměříme se proto na společné rysy a celkové trendy v oboru.

Shrnující práce od Poliner a kol. (2007) a Salamon a kol. (2014) se pro charakterizaci systémů pro transkripci odkazují na příbuznou úlohu odhadu fundamentální frekvence monfonní nahrávky. Algoritmy pro monofonní tracking na základě vstupního signálu $x(t)$ počítají *funkci salience* $S_x(f_\tau, \tau)$ pro každý krátký časový okamžik (okno) τ a frekvenci f_τ. Výsledkem této funkce je relativní ohodnocení (příp. pravděpodobnost) jednotlivých frekvencí obsažených ve vstupním signálu, které značí, zda-li je daná frekvence fundamentální frekvencí znějícího hlasu.

Výstupem monofonního trackingu je posloupnost frekvencí s maximální salienci, tedy posloupnost frekvencí, které jsou nejlépe ohodnocenými kandidáty na fundamentální frekvenci. V praxi se k salienci ještě přičítají temporální závislosti, aby se zajistila kontinuita extrahovaných frekvenčních kontur a zvýšila robustnost proti šumu obsaženému v nahrávce.

Přejdeme-li k úloze extrakce melodie, obecně se vstupní polyfonní signál $x(t) = x_m(t) + x_d(t)$ skládá ze směsi melodického hlasu $x_m(t)$ a hudebního doprovodu $x_d(t)$, cílem metod pro extrakci je z pohledu přepisu fundamentální frekvence zvýšení robustnosti algoritmu vůči tomuto „melodickému šumu“ $x_d(t)$. Výstupem našeho systému tedy bude posloupnost odhadů frekvence v každém časovém okně vstupního signálu, reprezentovaná vektorem $\hat{\mathbf{f}}$:

$$\hat{\mathbf{f}} = \underset{\mathbf{f}}{\operatorname{argmax}} \left[\sum_{\tau} S'_x(f_\tau, \tau) + C(\mathbf{f})\right]$$

kde f_τ je frekvence na pozici τ ve vektoru $\mathbf{f}$. $S'_x(f_\tau, \tau)$ je upravená funkce salience, která při výpočtu zohledňuje vliv doprovodu a složka $C(\mathbf{f})$ představuje temporální vlastnosti melodie.

Spolu s odhadem frekvencí by také měl systém na výstupu určit úseky, ve kterých v nahrávce melodie zní a kdy nikoli. K výstupu tedy patří také vektor $\hat{\mathbf{v}}$, se stejným počtem složek jako $\hat{\mathbf{f}}$, který značí znělost melodie v každém časovém okně τ.

Většina existujících metod sdílí podobnou základní strukturu při řešení extrakce, která se zakládá na popsané formalizaci. Prvním krokem je transformace zvuku do frekvenční domény a následný odhad znějících výšek tónů v polyfonním signálu (výpočet *funkce salience*), druhým krokem je pak zpracování těchto odhadů a výběr melodie (tedy zpřesnění výsledné $\hat{\mathbf{f}}$ pomocí $C(\mathbf{f})$). Přístupy k řešení těchto dvou kroků již s konkrétními příklady nastíníme v dalších sekcích.

1.2.1 Odhad výšek tónů

Spektrální analýza

Zvuk hraného tónu na melodickém nástroji je z fyzikálního pohledu periodická změna tlaku vzduchu. Perioda tohoto signálu se nazývá fundamentální frekvence (označujeme F0) a zpravidla je tento signál složen ze součtu řady sinusoid, jejichž frekvence jsou celočíselným násobkem fundamentální frekvence. V čase měnící se amplitudy těchto *harmonických frekvencí* udávají hlasitost a barvu hlasu, výška první harmonické frekvence (tj. výška fundamentální frekvence) pak ve většině případů odpovídá posluchačem vnímané výšce tónu.

TODO obrázek: signál -> spektrum -> salience

Prvním krokem metod pracujících s hudebním signálem je proto provedení spektrální analýzy, jde o převod zvuku do frekvenční reprezentace, která odhaluje tyto harmonické struktury tónů a umožňuje s nimi dále pracovat.

Přístupů ke spektrální analýze je více, přímočará a podle Dressler (2016) nejčastěji používaná metoda je krátkodobá Fourierova transformace (STFT). Jejím principem je rozdělení vstupního signálu na množinu překrývajících se oken konstantní délky a výpočet Fourierovy transformace těchto krátkých zvukových úseků. Komplexní výsledek transformace umocníme a získáme tzv. výkonové spektrum signálu, které obsahuje informaci o poměrech energie frekvencí, ze kterých se signál v okně skládá.

TODO? umocněná rovnice STFT

Na libovolnou metodu převodu diskretizovaného signálu na frekvenční doménu se inherentně vztahuje Gaborův limit (související s principem neurčitosti). Volbou délky vstupního okna transformace zpřesňujeme buď frekvenční nebo časové rozlišení výsledné spektrální reprezentace. Zvolíme-li krátké vstupní okno, zvyšujeme časové rozlišení (krátké okno lépe zachycuje rychlé změny průběhu signálu), avšak ztrácíme přesnost na frekvenční ose, opačný vztah platí pro volbu delšího okna.

Tato limitace je markatní zejména pokud STFT používáme pro hudební data. Z povahy hudebních intervalů a harmonických struktur tónů platí, že téměř všechny periodické signály se v hudební skladbě vyskytují ve vzájmených relativních poměrech (v případě intervalů v poměrech $2^{\frac{n}{12}}$ a v případě harmonických frekvencí v celočíselných), u vyšších tónů jsou proto rozdíly mezi relevantními frekvencemi absolutně větší než u nižších tónů. Frekvenční rozlišení STFT je konstantní na celém výstupním frekvenčním rozsahu. V praxi proto volba jakékoli velikosti okna zajistí dobrý poměr frekvenčního a časového rozlišení jen pro část

rozsahu. Ve výsledku je pak buď pro vyšší frekvence okno příliš velké (zbytečně detailní rozlišení frekvence na úkor časového) a nebo naopak pro nižší frekvence je okno nedostačující (rozlišení frekvence nemusí být ani na úrovni půltónů).

Z tohoto důvodu existují vedle STFT i další metody, jejichž cílem je nabídnout lepší kompromis frekvenčně-časového rozlišení v kontextu melodických dat. Goto a Hayamizu (1999) používají MRFFT (Multi-Resolution Fast Fourier Transform), principem je opakovaný downsampling signálu (převzorkování na nižší vzorkovací frekvenci) a aplikace Fourierovy transformace na každý vzniklý signál; s každou iterací spektrum obsahuje čím dál podrobnější informace o nižších frekvencích, protože vyšší frekvence se při downsamplingu ztratí. Brown (1990) popsala metodu Constant-Q Transform (CQT), která spočívá v použití proměnné délky okna Fourierovy transformace pro výstupní frekvenční pásma, která rovnoměrně pokrývají logaritmickou osu frekvence. Paiva a kol. (2004) napodobují mechanismy lidského sluchu pomocí banky pásmových filtrů (Cochleagram) s logaritmicky rozmístěnými mezními frekvencemi a sumy autokorelací na jednotlivých frekvenčních pásmech signálu (Summary correlogram).

I přes uvedené důvody se Salamon a kol. (2014) a Dressler (2016) domnívají, že metoda zpracování signálu příliš neovlivňuje výslednou přesnost algoritmů pro přepis melodie. Tvrzení dokládají jednak celkovým srovnáním výsledků metod ze všech ročníků soutěže MIREX a jednak neochvějnou převahou využití krátkodobé Fourierovy transformace, jakožto efektivní a dostačující metody pro spektrální analýzu.

Postprocessing spektrogramu

Po převodu signálu na frekvenční reprezentaci následuje u většiny metod některý druh úpravy celého spektrogramu, předcházející samotnému výpočtu *funkce salience*. Výsledkem tohoto kroku může být potlačení šumu a nemelodických částí signálu, zpřesnění informace o výšce znějících frekvencí nebo normalizace či jiná úprava amplitud.

Nejčastější úpravou je nalezení lokálních maxim; supresí nemaximálních oblastí se zbavíme velkého množství nemelodických složek signálu, přitom informaci o těch melodických neztratíme. Výhodou práce s množinou maxim je, že jejich frekvenci lze na základě spektra dále zpřesnit pomocí parabolické interpolace (Rao a Rao (2010)) a nebo využitím úhlové frekvence (Salamon a Gomez (2012), Dressler (2009)).

Jinou úpravou jsou různé druhy normalizace, ať jednoduché aplikace logaritmu na jednotlivé hodnoty spektrogramu (Cancela (2008), Bittner a kol. (2017)) nebo složitější strategie normalizace, které závisí na hodnotách celého výsledku Fourierovy transformace jednoho okna (Ryynänen a Klapuri (2008)), či které používají pohyblivé průměry nebo jinou metodu, beroucí v potaz širší zvukový kontext. Cílem normalizace je zvýraznění slabších harmonických frekvencí a potlačení celopásmových zvuků (například perkusí). Principielně podobným krokem je aplikace pásmového filtru (Goto a Hayamizu (1999)) pro zvýraznění frekvencí obsahující melodii. Případně využití psychoakustických filtrů modelující lidské vnímání hlasitosti (Salamon a Gomez (2012)). Ze signálu lze také oddělit melodické nástroje a perkusivní doprovod pomocí metod separace signálů (*source separation*). Používanými metodami jsou například Harmonic/Percussive Sound Separation (HPSS)

(Tachibana a kol. (2010)) Robust principal component analysis (RPCA) (Ikemiya a kol. (2016)).

Funkce salience

2. Datasety

Nedostupnost dostatečného množství dat pro automatickou transkripci melodie představuje zejména pro metody strojového učení otevřený problém. Zatímco pro vzdáleně příbuznou úlohu automatického přepisu mluveného slova existuje tisíce hodin nahrávek (například dataset LibriSpeech, který vznikl na základě audioknih), největší dataset s přepsanou melodickou linkou MedleyDB má celkovou délku pod osm hodin. Do roku 2014, kdy MedleyDB vznikl, existovaly datasety, které byly buď rozmanité, ale příliš krátké (ADC04, MIREX05, INDIAN08) nebo naopak celkově větší, ale žánrově a hudebně homogenní (MIREX09, MIR1K, RWC). V roce 2015 byl vydán dataset Orchset, který obsahuje 23 minut výňatků z orchestrálních skladeb různých období. Za dataset pro extrakci melodie by se také dal považovat Weimar Jazz Database, který je sice primárně zaměřený na využití v muzikologii, nicméně obsahuje přes 450 přepsaných jazzových sól. Novinkou z roku 2017 je vydání datasetu MDB-melody-synth, který byl automaticky vygenerován základě vstupní vícestopé hudby (převzaté z MedleyDB), existuje tedy naděje, že současný korpus pro přepis melodie by se mohl v budoucnu rozšířit o velkou část veřejně dostupných vícestopých nahrávek.

Co se týče blízké úlohy transkripce hudby, velikost největších datasetů se pohybuje v řádu desítek hodin, tudíž jde stále o omezené kolekce. Mezi největší se řadí MusicNet (orchestrální, 34 hodin), MAPS (klavír, 18 hodin), MDB-mf0-synth (multižánrový, 4,7 hodin), GuitarSet (kytara, 3 hodiny) a URMP (komorní orchestr, 1,3 hodiny). I když jde o úlohu, která je lépe definovaná (na rozdíl od extrakce melodie v celkovém přepisu nehraje takovou roli subjektivita při volbě hlavního hlasu), s použitím polyfonních nástrojů vyvstává problém náročné ruční anotace.

Vytváření nových datasetů je obecně velmi pracné a nákladné. Obvyklý postup totiž zahrnuje buď kompletní ruční přepis nahrávky nebo alespoň ruční opravu výstupu automatického přepisu, přičemž tuto práci odvedou nejkvalitněji pouze zaškolení hudebníci. Každá vzniklá anotace by se také měla překontrolovat, a to nejlépe jiným hudebníkem. Dalším problémem je vůbec identifikace melodie - jelikož je určení hlavní melodické linie subjektivní, musí se na výsledné anotaci shodnout co nejvíce posluchačů, ve výsledku se proto vybírají takové nahrávky, které nejsou sporné. S tím souvisí také zavedení a pečlivé dodržování anotační politky u komplexnějších skladeb (zejména orchestrálních), kde může melodii nést více hlasů zároveň (současně či střídajíc se). Také množství výchozích dat pro vznik datasetů není velké, jednak musí být skladby šiřitelné, pokud má být dataset volně dostupný a jednak by k nim měly být dostupné audiostopy, ze kterých je smíchán finální mix, jelikož ruční anotace pouze s pomocí finálního mixu je mnohem náročnější než anotace stop.

Existence dostatenčně velkého množství dat je obecně vzato zásadním předpokladem pro využití metod strojového učení pomocí hlubokých neuronových sítí, zejména pak pro netriviální úlohy, jakou je například přepis melodie, jelikož dovoluje zvětšení celkové kapacity modelu, aniž by docházelo k overfittingu. Také pro evaluaci metod, například i v soutěži MIREX, jsou potřeba takové datasety, které dobře reprezentují reálná data, přitom MedleyDB vzniklo mimo jiné z důvodu, že stávající datasety nestačily ani pro tento účel.

Možností řešení nastíněného probému nedostatku dat je více. Přímým řešením by byl návrh metody, která by celý proces vzniku datasetů výrazně ulehčila. O to se snaží článek Salamon a kol. (2017) a princip této metody popisuje kapitola XXX.

2.1 MedleyDB

Bittner a kol. (2014)

Multimodální, vícestopý dataset obsahující 122 nahrávek, k 108 z nich je dostupná anotace melodie. Kromě té obsahuje také metadata o všech písní s informacemi o žánru a instrumentalizaci. S celkovou délkou 7.3 hodiny jde o nejdelší dataset, který se zaměřuje na více hudebních žánrů. O rozmanitosti svědčí i to, že se v datasetu vyskytuje řada nástrojů mimoevropského původu, a že jen přibližně polovina písní obsahuje zpěv. Na rozdíl od ostatních datasetů jsou nahrávky ve většině případů celé písně, tedy nejde pouze o krátké výňatky, a ke každé jsou poskytnuty audiostopy, ze kterých je vytvořen výsledný mix. Na základě diskuze, kterou shrnuji v kapitole o definici melodie, autoři poskytují tři verze anotací, na základě různě obecných definic:

1. Základní frekvence nejvýraznějšího melodického hlasu, jehož zdroj zůstává po dobu nahrávky neměnný. [1]
2. Základní frekvence nejvýraznějšího melodického hlasu, jehož zdroje se mohou měnit.
3. Základní frekvence všech melodických hlasů, potenciálně pocházejících z více zdrojů.

Ačkoli třetí definice umožňuje, aby v anotaci znělo více melodických linek zároveň, v datasetu se nejedná o kompletní přepis nahrávek (použitelný pro úlohu *multif0 estimation*), ten autoři neposkytují.

Dataset vznikl obvyklou cestou ruční anotace, ze shromážděného vícestopého materiálu byly vybrány stopy s potenciálním výskytem melodie, stopy s přeslechem byly filtrovány pomocí source-separation algoritmu s ručně doladěnými parametry pro každou jednotlivou stopu, následně byl na monofonní stopy spuštěn pitch-tracker pYIN a výsledné automaticky získané anotace opravilo a vzájemně zkontrolovalo pět anotátorů s hudebním vzděláním.

2.2 MDB-synth

Hlavním přínosem práce Salamon a kol. (2017) je navržení způsobu anotace základní frekvence monofonních audiostop takovým způsobem, že výsledná dvojice zvukové stopy a anotace nevyžaduje další manuální kontrolu. Anotace stopy probíhá ve dvou krocích, nejprve získáme pomocí libovolného monopitch trackeru křivku základní frekvence a poté na základě této křivky, která může obsahovat chybné úseky, syntetizujeme novou stopu, která zachovává barvu nahrávky, ale

[1] Tato definice je shodná pro evaluační datasety používané v soutěži MIREX, s výjimkou Orchsetu

výšku tónu určuje právě tato anotace. Díky tomu je pak přesnost anotace pro tuto novou, syntetickou nahrávku stoprocentní, přitom (v ideálním případě) neztrácí charakteristiky původní nahrávky.

Pro vytváření datasetu je toto významné zjednodušení, protože tím algoritmus odstraňuje časově nejnáročnější část práce - ruční kontrolu anotací audiostop. Pokud by se ukázalo, že syntéza významně neubírá na kvalitě dat, použitím navrhované metody by mohlo vzniknout velké množství nových dat (například repozitář Open Multitrack Testbed obsahuje stovky vícestopých nahrávek, které by šlo využít). Autoři v článku provádí kvantitativní analýzu pomocí srovnání state-of-the-art algoritmů pro extrakci melodie a prokazují, že výsledky těchto metod na syntetických datech se významně neliší od výsledků na původních, tím je podle autorů potvrzená možnost použití dat jak pro trénování tak pro evaluaci nových metod.

Metoda má ale bohužel svá omezení, mezi ty zásadní patří, že se dá aplikovat pouze na stopy, které obsahují monofonní signál, vstupní data tedy nesmí obsahovat přeslech a nahrávaný nástroj může hrát pouze jednohlase, v důsledku nelze zpracovat klavír či kytara, které hrají zpravidla vícehlas. To nevadí tolik u generování datasetu pro přepis melodie, jelikož melodii často hraje jeden hlas a doprovod hrají ostatní, velkým nedostatkem je toto spíše pro generování multif0 datasetů.

Dále k článku není zveřejněná kompletní referenční implementace algoritmu, tudíž algoritmus nelze snadno spustit na nových datech. Ve výsledku je tudíž největším praktickým přínosem nová sada syntetických datasetů pro úlohy přepisu melodie, basy, monofonních stop a kompletní partitury, každý dataset obsahuje destíky nahrávek. Vícestopá data použitá pro syntézu byla převzata z MedleyDB, tudíž ve výsledku nové datasety nerozšiřují celkový hudební záběr, pouze zpřesňují ten již existující.

TODO obrázek? Porovnání spektrogramů syntetické a původní nahrávky

Z kvalitativního pohledu je na výstupních syntetických nahrávkách poznat, že jsou syntetické. Autoři sice prokazují, že současné metody na těchto datech dosahují stejných výsledků, nicméně v článku chybí diskuse o tom, zda-li v datech algoritmus nevytváří nové umělé artefakty, které by mohly zneužít metody strojového učení pro spolehlivější výsledky (které by však negeneralizovaly na reálná data). Při pohledu na spektrogram je například zřejmé, že syntetická nahrávka obsahuje mnohem více výrazných alikvótních frekvencí

2.3 Orchset

Dataset vytvořený týmem Bosch a kol. (2016) orientovaný na orchestrální repertoár pocházející z různých historických období včetně 20. století. Obsahuje 64 výňatků délky od 10 do 32 sekund. Výňatky byly vybírány tak, aby obsahovaly zřejmou melodii, dataset tedy obsahuje v porovnání málo pasáží bez melodie (6% z celkové délky). Vzhledem k komplexitě uvažovaných žánrů autoři vycházejí z kombinace rozšířené definice melodie podle Bittner a kol. (2014) a definice Poliner a kol. (2007). Melodii ve výňatcích proto zpravidla nese více hudebních nástrojů (nebo celých sekcí), které se v průběhu střídají, případně mohou části hrát společně v rozdílných oktávách (nebo jiných intervalech, tvoříce tak harmonický doprovod).

Pro zjištění melodie se v takto vrstveném materiálu autoři uchylují k úplnému základu definice melodie (Poliner) a nechávají si skupinou čtyř posluchačů výňatky přezpívávat. Tato hrubá data pak autoři sumarizují a odebírají z datasetu ty výňatky, na jejichž melodii se posluchači neshodli. Přezpívané tóny bylo nutné ručně opravit, aby načasováním přesně seděly na výňatek. Lidský hlas také samozřejmě nemá rozsah plného orchestru, proto bylo dalším krokem transponovat anotace tak, aby zněly ve správných oktávách. Zde se opět může vyskytnout problém subjektivity, pokud melodii hrají dva různé nástroje, pouze v jiných oktávách, pak je sporné, který nástroj označit jako hlavní (v některých případech taková otázka ani nedává příliš smysl.). Částečným řešením je zvolit libovolnou anotační politiku a tu konzistentně dodržovat (žádná společná v komunitě MIR neexistuje), v případě Orchsetu byla snaha minimalizovat skoky v melodické kontuře, což zároveň respektuje obecné pozorování, že v melodii se vyskytují mnohem častěji malé skoky (nejčastěji prima a malá/velká sekunda) než větší. Tedy například pokud pasáži hrané ve dvou různých oktávách předcházela pasáž hraná v jedné, anotace obou pasáží lze transponovat do společné oktávy tak, abychom na rozhraní minimalizovali skok v anotaci.

Dataset obsahuje pouze hrubé anotace tónů melodie, nikoli přesnou základní frekvenci nástroje, který v danou chvíli melodii hraje. Článek o tomto rozhodnutí příliš nediskutuje, vychází ale opět logicky z volby dat. U orchestrálních dat je tento abstraktnější pojem melodie mnohem méně sporný. Pokud hraje melodii sekce nástrojů v unisonu, přesná základní frekvence není dobře definovaná, jelikož se základní frekvence znějících hlasů vzájemně překrývají.

2.4 Weimar Jazz Database

Weimar Jazz Database Pfleiderer a kol. obsahuje přes 450 transkripcí jazzových sól ze všech období vývoje jazzu. Data původně zamýšlená pro muzikologické studie využívající statistické metody ale lze využít i pro potřeby extrakce melodie, jelikož uvažované nahrávky spadají zřejmě pod nejrestriktivnější definici melodie (definici používanou v soutěži MIREX) - melodii nese jistě právě jeden, sólový nástroj, a po celou dobu výňatku je jistě nejvýraznější. Výběr sólových nástrojů se omezuje pouze na jednohlasé, jelikož ruční anotace vícehlasých je příliš obtížná. Hlavním problémem při využívání je restriktivní licence, která platí na nahrávky, tudíž zdrojové audio, na základě kterého anotace vznikaly, není veřejně přístupné. Jelikož pro data neexistují jednotlivé stopy, ruční anotace probíhala přímo z finální nahrávky, což je obtížný úkol -

3. Evaluace metod

3.1 MIREX

Soutěž MIREX (Music Information Retrieval Evaluation eXchange) probíhá již od roku 2005 a v MIR komunitě zastává hlavní postavení jakožto každoroční událost pro nezávislé, objektivní srovnání state-of-the-art metod a algoritmů pro řešení širokého spektra úloh souvisejících se zpracováním hudebních dat. Mezi tyto úlohy patří například *rozpoznání žánru*, *odhad tempa*, *odhad akordů*, *identifikace coveru* a samozřejmě také *extrakce melodie*.

Na rozdíl od jiných úloh, kde debata o zvolení nejvhodnějších objektivních metrik pro porovnávání stále probíhá, metriky pro extrakci melodie se ustanovily již v prvním ročníku (na základě dřívějších zkušeností) a zůstaly neměnné dodnes Raffel a kol. (2014). Naopak data použitá pro testování se postupně kumulují a dnes soutěž probíhá již s řadou datasetů (ADC04, MIREX05, MIREX08, MIREX09, ORCHSET), které blíže popisuje kapitola o dostupných datech.

Downie a kol. (2010)

3.2 Trénovací, validační a testovací množina

Z dostupných dat, které pro úlohu máme k dispozici, musíme vyhradit množiny pro trénování, validaci a testování, aby byly metody porovnatelné jak mezi sebou, tak se stávajícími state-of-the-art metodami. Pro trénování se jeví jako nejvhodnější dataset MedleyDB, jednak pro svou délku a jednak pro žánrovou rozmanitost, proto je použit pro většinu popsaných experimentů. Rozdělení na tři části vychází z práce Bittner a kol. (2017) a D Basaran, S Essid (2018), aby byly metriky přímo porovnatelné s výsledky v uvedených článcích. Další výhodou použití stejného *splitu* je možnost replikace výsledků, za použití popisované architektury, a tím pádem minimalizování možnosti nějaké velké implementační chyby v kódu. Pokud by se totiž výsledky nepodařilo replikovat se stejnými daty i architekturou, musela by být chyba jinde - tedy s největší určitostí v vyvinutém frameworku.

Dalším užitečným zdrojem dat je dataset *MDB-melody-synth*, který je přesyntetizován z vícestopých nahrávek *MedleyDB*, proto se nabízí použít stejné rozdělení dat, jaké se používá pro *MedleyDB*, ze stejných důvodů uvedených v předchozím odstavci. Jelikož dataset neobsahuje veškerá data, ale pouze jejich podmnožinu, i v experimentech používaný *split* obsahuje pouze podmnožinu z původního *splitu* datasetu *MedleyDB*.

Posledním velkým datasetem, používaným pro trénování, je *Weimar Jazz Database*. Zde žádný doporučený postup ani výběr rozdělení dataestu v relevatní literatuře neexistuje, proto jsem dataset rozdělil podle metody Bittner a kol. (2017) na tři části (v celkové délce nahrávek na části v poměrech 63%, 14% a 23%). Skladby jsou rozděleny do částí podle interpretů tak, aby se každý interpret vyskytoval právě v jedné části datasetu. Toto omezení na podmnožiny Bittner a kol. (2017) nediskutuje, lze však doložit (práce Sturm (2013)), že pro úlohu *rozpoznání žánru* metody založené na strojovém učení vykazují po trénování a validaci na

datech bez tohoto filtru výrazně lepší výsledky než stejné metody spuštěné na roztříděných datech, takové zlepšení výkonu je ale jistě umělým důsledkem špatné volby trénovací množiny.

Ostatní datasety (ADC04, MIREX05, ORCHSET) jsou v práci použity pouze jako testovací data, díky tomu lze korektně výsledky přímo srovnávat s žebříčky úlohy Melody Extraction v soutěži MIREX.

3.3 Kvalitativní příklady

Pro lepší porozumění hranic testovaných metod je vhodné studovat také výsledky na kvalitativních ukázkách. Modely byly při práci vyhodnocovány na několikaminutových množinách výňatků z validačních a testovacích dat. Metodika výběru spočívala v poslechu nahrávek a ručním hledáním zajímavých hudebních jevů a také v seřazení nahrávek podle úspěšnosti přepisu stávajícími metodami a výběrem výňatků právě z těchto nejproblematičtějších příkladů.

Omezení plynoucí z potřeby zkrátit výňatky na minimum,

3.4 Metriky

Celkovou kvalitu metody pro extrakci melodie určuje její schopnost určit výšku tónu hrající melodie (*odhad výšky melodie*) a také rozpoznat části skladby, které melodii neobsahují (*detekce melodie*). Jelikož jsou tyto podúlohy na sobě nezávislé, standardní sada metrik zahrnuje jak celkové vyhodnocení přesnosti, tak dílčí vyhodnocení pro *odhad výšky* a *detekci melodie*.

3.4.1 Formát výstupu

Obvyklý formát výstupu algoritmů je CSV soubor se dvěma sloupci. První sloupec obsahuje pravidelné časové značky, druhý sloupec pak odhad základní frekvence melodie. Některé algoritmy uvádí i odhady výšky základní frekvence mimo detekovanou melodii (může jít například o doprovod, který zní i po hlavním melodickém hlasu). Aby tyto odhady byly odlišené od odhadů hlavní melodie, jsou uvedeny v záporných hodnotách. Díky tomu pak lze nezávisle vyhodnotit přesnost *odhadu výšky* a *detekce melodie*. Odhad výšky se vyhodnocuje podle absolutní hodnoty frekvence ve všech časových oknech, ke kterým existuje anotace, detekce melodie pak na všech hodnotách vyšších než 0.

3.4.2 Definice metrik

Většina metrik je definována na základě porovnávání jednotlivých anotačních oken - tedy typicky srovnáním odhadovaných a pravdivých výšek melodie po konstantních časových skocích. Datasety používané pro vyhodnocování v soutěži MIREX používají časový skok délky 10 ms. V definicích budu vycházet ze značení v práci Salamon a kol. (2014).

Označme vektor odhadovaných základních frekvencí $\mathbf{f}$ a cílový vektor $\mathbf{f}^*$, složka f_τ je buď rovna hodnotě f_0 melodie nebo 0, pokud v daném čase melodie nezní. Obdobně zaveďme vektor indikátorů $\mathbf{v}$, jehož prvek na pozici τ je

roven $v_\tau = 1$, pokud je v daném časovém okamžiku detekována melodie a $v_\tau = 0$ v opačném případě. Podobným způsobem zavedeme i vektor cílových indikátorů melodického hlasu $\mathbf{v}^*$ a také vektor indikátorů absence melodie $\bar{v}_\tau = 1 - v_\tau$.

Voicing Recall rate (Úplnost detekce)

Poměr počtu časových oken, které byly správně označené jakožto obsahující melodii, a počtu časových oken doopravdy obsahujících melodii podle anotace.

$$\mathrm{VR}(\mathbf{v}, \mathbf{v}^*) = \frac{\sum_\tau v_\tau v_\tau^*}{\sum_\tau v_\tau^*}$$

Voicing False Alarm rate (Nesprávné detekce)

Poměr počtu časových oken, které byly nesprávně označené jako melodické, k počtu doopravdy nemelodických oken.

$$\mathrm{FA}(\mathbf{v}, \mathbf{v}^*) = \frac{\sum_\tau v_\tau \bar{v}_\tau^*}{\sum_\tau \bar{v}_\tau^*}$$

Raw Pitch Accuracy (Přesnost odhadu výšky)

Poměr správně odhadnutých tónů k celkovému počtu melodických oken. Výška správně určeného tónu se může lišit až o jeden půltón.

$$\mathrm{RPA}(\mathbf{f}, \mathbf{f}^*) = \frac{\sum_\tau v_\tau^* v_\tau \mathcal{T}[\mathcal{M}(f_\tau) - \mathcal{M}(f_\tau^*)]}{\sum_\tau v_\tau^*}$$

kde $\mathcal{T}$ je prahová funkce

$$\mathcal{T}[a] = \begin{cases} 1 & \text{pro} |a| \leq 0.5 \\ 0 & \text{jinak} \end{cases}$$

a $\mathcal{M}$ je funkce zobrazující frekvenci f na reálné číslo počtu půltónů od nějakého referenčního tónu f_{ref} (například od 440 Hz, tedy komorního A4).

$$\mathcal{M}(f) = 12 \log_2 (\frac{f}{f_{\mathrm{ref}}})$$

Raw Chroma Accuracy (Přesnost odhadu výšky nezávisle na oktávě)

Počítá se podobně jako *Přesnost odhadu tónu*, výstupní a cílové tóny jsou však mapovány na společnou oktávu. Metrika tedy ignoruje chyby odhadu způsobené špatným určením oktávy tónu.

$$\mathrm{RCA}(\mathbf{f}, \mathbf{f}^*) = \frac{\sum_\tau v_\tau^* v_\tau \mathcal{T}[\langle \mathcal{M}(f_\tau) - \mathcal{M}(f_\tau^*) \rangle_{12}]}{\sum_\tau v_\tau^*}$$

Nezávislost na oktávě zajistíme pomocí zobrazení rozdílu cílového a výstupního tónu na společnou oktávu.

$$\langle a \rangle_{12} = a - 12 \lfloor \frac{a}{12} + 0.5 \rfloor$$

Overall Accuracy (Celková přesnost)

Celková přesnost měří výkon algoritmu jak v odhadu melodie tak v detekci melodie. Počítá se jako podíl správně odhadnutých oken a celkového počtu oken.

$$\mathrm{OA}(\mathbf{f}, \mathbf{f}^*) = \frac{\sum_\tau v_\tau^* v_\tau \mathcal{T}[\mathcal{M}(f_\tau) - \mathcal{M}(f_\tau^*)] + \bar{v}_\tau^* \bar{v}_\tau}{L}$$

Poznámka k definicím metrik

Definice RPA, RCA a OA zde uvedené se mírně liší od výchozích v práci Salamon a kol. (2014), jejich přímá implementace podle vzorce totiž vede kvůli nedostatečně dobře zadefinovanému vektoru frekvencí $\mathbf{f}$ k chybě, která byla přítomna i v nejpoužívanější, veřejné implementaci MIR metrik *mir_eval*. Tato chyba se týká zejména metriky RCA, která v původní definici chybně zahrnovala jako správné tóny ty, které algoritmus odhadl jako nulové (tedy neznějící) a zároveň jejich pravdivá hodnota byla po zobrazení na jednu společnou oktávu blízká nule (tedy původní tón byl blízký nějakému násobku referenční frekvence). Kvůli zobrazení na společnou oktávu se stanou „neznělé nulové odhady“ a tóny blízké referenčním frekvencím nerozlišitelné a byly nesprávně považované za korektní.

V praxi chyba této metriky na datasetu MedleyDB mohla dosahovat až sedmi procentních bodů, na repozitáři hostovaném na serveru Github jsme již spolu s autory chybu odstranili [1]. Opravný patch bude zahrnut do další verze balíku.

3.4.3 Další metriky

Protože princip vnitřního fungování neuronových sítí často není zřejmý, je užitečné mít co nejvíce různých indikátorů, abychom měli při porovnávání jednotlivých modelů alespoň podrobnou informaci, v jakých ohledech se síť zlepšuje nebo zhoršuje. Pro tento účel jsem při práci implementoval další metriky, které při hledání architektur sítí pomáhaly.

Chroma Overall Accuracy

Počítá se obdobně jako Overall Accuracy, ale tóny jsou mapovány na společnou oktávu.

Raw Harmonic Accuracy

Metrika počítá odhadovaný tón jako správný, pokud se trefil do některé z harmonických frekvencí tónu. Protože je harmonických frekvencí teoreticky nekonečné množství, parametrem metriky je do jakého celočíselného násobku se ještě odhad počítá.

$$\mathrm{RHA}(\mathbf{f}, \mathbf{f}^*, n) = \frac{\sum_{k=1}^{n} \sum_\tau v_\tau^* v_\tau \mathcal{T}[\mathcal{M}(f_\tau) - \mathcal{M}(k f_\tau^*)]}{\sum_\tau v_\tau^*}$$

[1] odkaz na Github issue: `https://github.com/craffel/mir_eval/issues/311`

Matice záměn not

Pro podrobnější souhrnný přehled četností chyb se pro klasifikační úlohy používá matice záměn. Sloupce označují správné noty, řádky odhadované. Buňka na pozici (x,y) má pak hodnotu podle četnosti odhadu noty y místo správné noty x.

4. Experimenty

Práce obsahuje souhrnné výsledky experimentů zejména nad datasetem MedleyDB, aby modely byly dobře porovnatelné se state-of-the-art výsledky a výhoda prezentovaných metod netkvěla pouze v použití více dat. U vybraných experimentů došlo k přetrénování na větší trénovací množině, aby bylo možné posoudit vliv množství dat na výsledný výkon.

V první části se zabývám zejména odhadováním *výšky tónů*. U úspěšných architektur pak implementuji i *detekci melodie*.

4.1 CREPE Architektura

První sada experimentů se zakládá na architektuře popsané v článku od Kim a kol. (2018) použité pro *monopitch tracking*. Přestože se nejedná o úlohu extrakce melodie, cílem monopitch trackingu je určit konturu základní frekvence melodického nástroje v monofonní nahrávce, která se skládá ze součtu čistého signálu a šumu v pozadí. Pokud rozšíříme pojem šumu v pozadí tak, aby zahrnoval i melodický doprovod, pak dostáváme formální definici signálu zpracovávaného algoritmy pro přepis melodie Salamon a kol. (2014).

Jinými slovy - *monopitch tracking* je speciálním případem extrakce melodie a tudíž přinejmenším stojí za zkoušku pokusit se tuto architekturu pro extrakci využít. Mimo to monofonní stopy často obsahují přeslech ostatních nástrojů, pokud nahrávka vznikala při společném hraní, tudíž by model trénovaný na výsledných mixech mohl být robustní vůči tomuto druhu rušení.

Architektura CREPE se sestává ze šesti konvolučních a pooling vrstev, pro regularizaci používá batch normalization a dropout po každé konvoluční vrstvě, jako aktivační funkce používá ReLU. Po konvolucích následuje výstupní plně propojená vrstva se sigmoid aktivací. Vstupem modelu je okno o velikosti 1024 samplů, audio je převzorkováno na 16 kHz. Před první konvolucí je vstup normalizován tak, aby každé jednotlivé okno se vzorky mělo střední hodnotu 0 a směrodatnou odchylku 1. Přesná podoba modelu je naznačena na obrázku.

Výsledný vektor o 640 složkách aproximuje pravděpodobnostní rozdělení výšky základní frekvence uprostřed vstupního okna, přičemž tento vektor pokrývá rozsah od noty C_{-1} po G_9, mezi dvěma sousedními predikovanými tóny je vzdálenost 20 centů. Výšky tónů v centech označíme $¢_1, ¢_2, \ldots, ¢_{640}$. Rozsah tedy bezpečně pokrývá obvyklé hudební nástroje a na jednu notu připadá 5 složek (tónů) výsledného vektoru.

$$¢(f) = 1200 \log_2 \frac{f}{f_{\text{ref}}}$$

Pro trénování modelu potřebujeme také cílové diskrétní pravděpodobnostní rozdělení základní frekvence tónu. Jako cílovou pravděpodobnostní funkci použijeme normální rozdělení se střední hodnotou v bodě cílové základní frekvence $¢(f_{\text{ref}})$ a se směrodatnou odchylkou 25 centů. Toto rozdělení dikretizujeme, aby měl cílový vektor stejné dimenze jako odhadovaný.

$$y_i = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\c{c}_i - \c{c}_{\text{ref}})^2}{2\sigma^2}\right)$$

Převod z výstupního vektoru na výšky not provedeme pomocí střední hodnoty výstupního vektoru. Jelikož by ale výšku tónu ovlivňoval i další melodický šum, který se na výstupním vektoru také objevuje, spočítáme střední hodnotu pouze z okolí maxima výstupu.

$$\hat{\c{c}} = \sum_{i,|\c{c}_i - \c{c}_m|<50} \hat{y}_i \c{c}_i \Big/ \sum_{i,|\c{c}_i - \c{c}_m|<50} \hat{y}_i \ , m = \text{argmax}_i(\hat{y}_i)$$

Optimalizovaná loss funkce modelu $\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}})$ se počítá jako binární vzájemná korelace mezi vektorem cílových pravděpodobností y a výstupním vektorem $\hat{y}$.

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{i=1}^{640} \left(-y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i)\right)$$

Optimalizace probíhá pomocí algoritmu Adam (Kingma a Ba, 2014) s learning rate 0.0002.

4.1.1 Replikace výsledků CREPE

Pro ověření správnosti implementace architektury *monopitch trackeru CREPE* spustíme model na syntetických, monofonních datech používaných v článku Salamon a kol. (2017). Na rozdíl od článku Kim a kol. (2018) jsem model netrénoval na všech datech pomocí postupu *5 fold cross validation*, jiné zásadní rozdíly mezi implementacemi jsem však na základě článku a veřejně dostupného kódu neidentifikoval.

Po jedné epoše trénování model dosáhl vyšší přesnosti, než je uváděná v literatuře, tento rozdíl přičítám zejména zmiňované odlišné evaluační strategii.

Metrika	Práh	Průměrná hodnota	Hodnota Kim a kol. (2018)
RCA	50 centů	0.988	0.970
RPA	50 centů	0.986	0.967
RPA	25 centů	0.975	0.953
RPA	10 centů	0.937	0.909

Při replikaci experimentu jsem narazil na důležitost správného promíchání dat. Framework Tensorflow použitý pro trénování promíchává data vždy pomocí bufferu pevné velikosti pro dvojice vstupů a cílových výstupů. V praxi je však potřeba buď nastavit buffer na velikost větší než je celková velikost datasetu, a nebo implementovat vlastní míchání přes všechna dostupná data. Při nedostatečně promíchaných datech totiž trénovací dávky (batch) nejsou reprezentativní pro celý dataset, ale pouze pro jeho podmnožinu, což se negativně projevuje kolísající validační přesností modelu.

4.1.2 CREPE pro extrakci melodie

Jako první experiment nad melodickými daty spustíme nezměněnou architekturu CREPE, v následujících experimentech se tuto baseline pokusíme překonat. Abychom urychlili trénování následujících experimentů, přesnost určíme pro sítě

s různou kapacitou, pokud se výsledky při různých kapacitách příliš neliší, můžeme experimenty provádět s architekturou s nižší kapacitou. Kapacity upravíme pomocí multiplikátoru počtu filtrů u všech konvolučních vrstev, počty filtrů jsou uvedeny v tabulce.

Vrstva	1.	2.	3.	4.	5.	6.	Celkový počet parametrů
CREPE 4x	128	16	16	16	32	64	558240
CREPE 8x	256	32	32	32	64	128	1771200
CREPE 16x	512	64	64	64	128	256	6163200

Model	RPA	RCA
CREPE 4x	0.634	0.753
CREPE 8x	0.661	0.766
CREPE 16x	0.666	0.771
Salamon	0.547	0.608
Bittner	0.735	0.791
Basaran	0.737	0.803

Z výsledků na validačních datech po 200k iteracích (přibližně 6 epoch) je zřejmé, že překonání state-of-the-art metod založených na pravidlovém zpracování zvuku (**?**) není obtížné. Zároveň také vidíme, že se výsledek modelů CREPE 8x a CREPE 16x liší řádově o desetiny procentních bodů a přitom model s větší kapacitou se trénuje o 35

4.1.3 Vliv rozlišení diskretizace výšky noty

Otestujeme nastavení granularity výstupního vektoru. V článku Kim a kol. (2018) se totiž důvod volby pěti frekvencí na notu nediskutuje. Intuitivně by však mělo vyšší rozlišení spíše pomáhat, důvodem je, že nástroje a zejména lidský hlas se často při hraní odchylují od přesných frekvencí hraných not a vyšší rozlišení tyto odchylky může lépe zachytit.

Kapacita	Diskretizace	RPA	RCA
4x	hrubá	0.606	0.708
4x	jemná	0.634	0.753
8x	hrubá	0.614	0.724
8x	jemná	0.661	0.766
16x	hrubá	0.612	0.711
16x	jemná	0.666	0.771

Jak je vidět z tabulky a grafů, jemná granularita výstupu jednoznačně zlepšuje přesnost sítě. Abychom potvrdili hypotézu, že vyšší rozlišení pomáhá zmenšit počet chyb o půltón, můžeme vytvořit histogram vzdáleností cílového a odhadovaného tónu, v tomto histogramu by pak měl být vidět pokles v příslušných třídách.

Podle histogramu se počet chyb o půltón mezi zkoumanými modely liší téměř o polovinu, zlepšení tohoto druhu chyb je tedy podstatné.

4.1.4 Vliv rozptylu cílové pravděpodobnostní distribuce výšky noty

Podle Bittner a kol. (2017) pomáhá cílová distribuce s vyšším rozptylem snížit penalizaci sítě za téměř korektní odhady výšek tónů. Mimo to u dostupných dat často nejsou anotace naprosto perfektní, jisté rozostření hranice anotace tudíž pomáhá i v případě nepřesné cílové anotace, síť pak není tolik penalizována za svou případnou správnou odpověď.

V článku se však nediskutuje nastavení směrodatné odchylky na 20 centů, Kim a kol. (2018) používá odchylku 25 centů a není na první pohled zřejmé, jaká je optimální hodnota. Příliš vysoký rozptyl způsobí, že síť bude tolerovat více chyb o půltón, příliš nízký rozptyl naopak penalizuje i téměř správné odhady. Intuitivně se nejlepší nastavení pravděpodobně bude pohybovat kolem používaných 25 centů, jelikož to je hranice chybné klasifikace, na druhou stranu optimální hodnota jistě bude závislá na nastavení rozlišení výstupního vektoru, jelikož nižší rozlišení bude jistě vyžadovat vyšší hodnotu rozptylu (v extrémním případě rozptylu blížícího se k nule a cílové frekvence mimo kvantizační hladiny by vzniklý cílový vektor nemusel obsahovat žádné ostré maximum).

Poznamenám také technický detail, který je důležitý při samotné implementaci. Přestože jsem cílový výstup sítě zadefinoval jako diskrétní pravděpodobnostní rozdělení, při trénování je tento vektor hodnot pronásoben koeficientem tak, aby $\max(\mathbf{y}) = 1.0$ a tedy součet prvků vektoru není roven jedné (a o pravděpodobnostní rozdělení se doopravdy nejedná). Důvodem je použití aktivační funkce *sigmoid* u výstupní vrstvy, která nezaručuje výstup korektního rozdělení. Díky tomu se na výstupu může objevit různé množství stejně pravděpodobných kandidátů na melodii.

Testovaná síť má vstupní okno široké 4096 vzorků, používá multiplikátor kapacity 16x a vstup zpracovává 6 různě širokými konvolučními vrstvami (viz experiment *Vliv násobného rozlišení první konvoluční vrstvy*).

Směrod.	Raw Pitch Accuracy	Raw Chroma Accuracy
0.000	0.657	0.759
0.088	0.672	0.775
0.177	0.689	0.784
0.354	0.669	0.773
0.707	0.654	0.757

Z experimentů vyplývá, že optimální směrodatná odchylka se pohybuje kolem hodnoty 0.177, tedy níže než v porovnávaných pracích.

4.1.5 Vliv šířky vstupního okna

Architektura CREPE byla navržena pro monopitch tracking, dá se předpokládat, že jelikož je v monofonních nahrávkách oproti polyfonním daleko méně (melodického) šumu, není pro určení výšky tónu potřeba větší kontext než použitých 1024 vzorků (při vzorkovací frekvenci 16kHz toto odpovídá 64 milisekundám audia). To ale nemusí platit pro složitější signály, kde by síť mohla z delšího kontextu těžit. Otestujeme tedy vliv většího vstupního okna na výslednou přesnost.

Šířka vstupního okna	Raw Pitch Accuracy	Raw Chroma Accuracy
512 (32 ms)	0.634	0.748
1024 (64 ms)	0.645	0.763
2048 (128 ms)	0.648	0.760
4096 (256 ms)	0.650	0.762
8192 (512 ms)	0.675	0.775

4.1.6 Vliv násobného rozlišení první konvoluční vrstvy

Podle Kim a kol. (2018) se přesnost CREPE snižuje s výškou tónu. Autoři si tuto skutečnost vysvětlují neschopností modelu generalizovat na barvy a výšky tónů neobsažených v trénovací množině, generalizaci by ale mohla pomoci také úprava modelu. Protože k rozpoznání vyšších frekvencí stačí méně vzorků než pro rozpoznání nižších, mohli bychom se pokusit upravit první konvoluční vrstvu sítě, která tento úkol zastává, a rozdělit ji na množiny různě širokých konvolucí, jejichž kanály následně sloučíme zpět do jednotné vrstvy. To by mělo mít za následek, že rozpoznávání vysokých tónů budou zastávat užší konvoluce a jejich kernel bude jednodušší než široké kernely s vysokou mírou redundance.

První vrstvu s kernelem s 256 filtry (tj. počet filtrů první vrstvy s multiplikátorem 8x, viz první experiment) jsem rozdělil na vícero různě širokých kernelů s menším počtem filtrů, tak aby kapacita sítě zůstala přibližně stejná a sítě byly porovnatelné.

Počet/šířka kernelů	512	256	128	64	32	16	8	4	Celkový počet parametrů
1	256								2098880
2	128	128							2066112
3	85	85	85						2041918
4	64	64	64	64					2029248
5	51	51	51	51	51				2016350
6	42	42	42	42	42	42			2001944
7	36	36	36	36	36	36	36		1996184
8	32	32	32	32	32	32	32	32	2000448

Experiment jsem provedl na síti se vstupním oknem 978 vzorků, multiplikátorem kapacity 8,

Počet konvolučních vrstev	Raw Pitch Accuracy	Raw Chroma Accuracy
1	0.629	0.734
2	0.628	0.732
3	0.632	0.734
4	0.636	0.739
5	0.643	0.740
6	0.638	0.737
7	0.636	0.736
8	0.640	0.737

Zlepšení výsledků se pohybuje v řádu desetin procentních bodů, tedy není příliš vysoké. Zlepšení je nejvíce patrné v případě pěti různě širokých konvolučních vrstev, kde dosahuje 1.3 procentního bodu. Analýzou výsledků přesnosti podle výšky noty se mi nepodařilo prokázat hypotézu, že by konvoluce s více rozlišeními

pomáhala u odhadu not vyšších frekvencí. Její přínos je drobný a projevuje se na většině frekvenčních pásem.

4.2 Wavenet Architektura

Generativní model WaveNet popsaný týmem van den Oord a kol. (2016) je architektura navržená pro generování zvukového signálu, autoři však síť testovali i pro převod mluvené řeči na text (dataset TIMIT) a dosáhli výsledků srovnatelných se state-of-the-art. Síť se však pro *Music Information Retrieval* od svého zveřejnění příliš neuchytila. Její použití se v oblasti hudby se omezuje na generativní úlohy (Hawthorne a kol. (2018), Yang a kol. (2017), Engel a kol. (2017) a další), případně *source-separation* (Stoller a kol., 2018). Jediný publikovaný pokus s použitím architektury WaveNet pro automatický přepis podnikli Martak a kol. (2018) nad datasetem MusicNet. Jejich model však netestovali na standardních evaluačních datasetech ze soutěže MIREX, tudíž není zřejmé, jakých výsledků v porovnání s existujícími metodami autoři dosáhli.

Architektura spočívá v důmyslném vrstvení dilatovaných konvolucí. Díky exponenciálně rostoucím dilatacím se také exponenciálně zvětšuje receptivní pole jednotlivých konvolučních vrstev. Díky této vlastnosti pak například stačí pro pokrytí 1024 vzorků vstupu pouze 9 vrstev s šířkou kernelu 2 a dilatacemi 1,2,4,8 ... 512. Pokud bychom stejného receptivního pole chtěli dosáhnout pomocí obvyklých konvolucí počet potřebných vrstev by byl lineární vzhledem k šířce pole. Vrstvení konvolucí je porovnáno na schématu.

4.2.1 Baseline na základě Martak a kol. (2018)

Pro srovnání spustíme architekturu popsanou ve zmíněném článku pro úlohu extrakce melodie. Jelikož byla architektura zamýšlena pro dataset MusicNet, který obsahuje celý přepis skladeb do MIDI not, výstupem jsou diskrétní noty. Jak jsme zjistili v předchozím experimentu na architektuře CREPE, hrubá diskretizace výrazně zhoršuje přesnost výsledků, upravíme proto architekturu tak, aby měla výstupní distribuce jemnější rozlišení.

5. Výsledky

Závěr

Seznam použité literatury

Bittner, R., Salamon, J., Tierney, M., Mauch, M., Cannam, C. a Bello, J. (2014). MedleyDB: A multitrack dataset for annotation - intensive mir research. *International Society for Music Information Retrieval Conference*.

Bittner, R. M., Mcfee, B., Salamon, J., Li, P. a Bello, J. P. (2017). Deep Salience Representations for F0 Estimation in Polyphonic Music. *Ismir*, pages 23–27. URL https://bmcfee.github.io/papers/ismir2017{_}salience.pdf.

Bosch, J. J. a Gómez, E. (2016). Melody Extraction Based on a Source-Filter Model Using Pitch Contour Selection. *13th Conference on Sound and Music Computing*, pages 67–74.

Bosch, J. J., Marxer, R. a Gómez, E. (2016). Evaluation and combination of pitch estimation methods for melody extraction in symphonic classical music. *Journal of New Music Research*, **45** (2), 101–117. ISSN 17445027. doi: 10.1080/09298215.2016.1182191. URL https://repositori.upf.edu/bitstream/handle/10230/26985/Bosch{_}NewMusic{_}Eval.pdf?sequence=1{&}isAllowed=y.

Brown, J. C. (1990). Calculation of a constant Q spectral transform. *The Journal of the Acoustical Society of America*, **89**(1), 425–434. ISSN 0001-4966. doi: 10.1121/1.400476. URL http://academics.wellesley.edu/Physics/brown/pubs/cq1stPaper.pdf.

Cancela, P. (2008). Tracking melody in polyphonic audio. *4th Music Information Retrieval Evaluation eXchange*. URL https://pdfs.semanticscholar.org/226e/a7b870fd229149fae2e6c8b15d8a3d4f9bb8.pdf.

D Basaran, S Essid, G. P. (2018). Main melody extraction with source-filter nmf and crnn. *Ismir*, pages 82–89. URL http://ismir2018.ircam.fr/doc/pdfs/273{_}Paper.pdf.

Downie, J. S., Ehmann, A. F., Bay, M. a Jones, M. C. (2010). eXchange : Some Observations and Insights. *Advances in Music Information Retrieval*, pages 93–115. URL https://pdfs.semanticscholar.org/3836/15607f345a8cfbc5e884eaa5ca04f3c4b139.pdf.

Dressler, K. (2009). Audio melody extraction for mirex 2009. *Evaluation eXchange (MIREX)*, pages 1–3. URL http://www.idmt.fraunhofer.de/content/dam/idmt/de/Dokumente/Produktflyer/QbH/white{_}paper{_}fraunhofer{_}idmt{_}audio{_}melody{_}extraction{_}mirex{_}200
pdf.

Dressler, K. (2016). Automatic Transcription of the Melody from Polyphonic Music. URL https://www.db-thueringen.de/servlets/MCRFileNodeServlet/dbt{_}derivate{_}00038847/ilm1-2017000136.pdf.

Durrieu, J.-l. a David, B. (2010). Source / Filter Model for Unsupervised Main Melody. **18**(3), 1–12. URL https://www.irit.fr/{~}Cedric.Fevotte/publications/journals/ieee{_}asl{_}voice{_}extrac.pdf.

Engel, J., Resnick, C., Roberts, A., Dieleman, S., Eck, D., Simonyan, K. a Norouzi, M. (2017). Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders. URL http://arxiv.org/abs/1704.01279.

Goto, M. a Hayamizu, S. (1999). A real-time music scene description system: Detecting melody and bass lines in audio signals. *IJCAI-99 Workshop on Computational Auditory Scene Analysis*, (August), 31–40. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.33.1085{&}rep=rep1{&}type=pdf.

Hawthorne, C., Stasyuk, A., Roberts, A., Simon, I., Huang, C.-Z. A., Dieleman, S., Elsen, E., Engel, J. a Eck, D. (2018). Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset. pages 1–12. URL http://arxiv.org/abs/1810.12247.

Ikemiya, Y., Itoyama, K. a Yoshii, K. (2016). Singing Voice Separation and Vocal F0 Estimation Based on Mutual Combination of Robust Principal Component Analysis and Subharmonic Summation. *IEEE/ACM Transactions on Audio Speech and Language Processing*, **24**(11), 2084–2095. ISSN 23299290. doi: 10.1109/TASLP.2016.2577879.

Kim, J. W., Salamon, J., Li, P. a Bello, J. P. (2018). Crepe: A Convolutional Representation for Pitch Estimation. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, **2018-April**, 161–165. ISSN 15206149. doi: 10.1109/ICASSP.2018.8461329. URL https://arxiv.org/pdf/1802.06182.pdf.

Kingma, D. P. a Ba, J. (2014). Adam: A Method for Stochastic Optimization. pages 1–15. URL http://arxiv.org/abs/1412.6980.

Martak, L. S., Sajgalik, M. a Benesova, W. (2018). Polyphonic Note Transcription of Time-Domain Audio Signal with Deep WaveNet Architecture. *International Conference on Systems, Signals, and Image Processing*, **2018-June**, 2–6. ISSN 21578702. doi: 10.1109/IWSSIP.2018.8439708. URL https://vgg.fiit.stuba.sk/wp-uploads/2018/09/iwssip2018{_}wavenet.pdf.

Paiva, R. P., Mendes, T. a Cardoso, A. (2004). An Auditory Model Based Approach for Melody Detection in Polyphonic Musical Recordings. (May 2014), 21–40. doi: 10.1007/978-3-540-31807-1_2.

Pfleiderer, M., Frieler, K., Abesser, J., Zaddach, W.-G. a Burkhart, B. *Inside the Jazzomat*. ISBN 9783959831246. URL http://schott-campus.com/wp-content/uploads/2017/11/inside{_}the{_}jazzomat{_}final{_}rev{_}oa4.pdf.

Poliner, G. E., Member, S., Ellis, D. P. W., Member, S., Ehmann, A. F., Gómez, E., Streich, S. a Ong, B. (2007). Melody Transcription

From Music Audio : Approaches and Evaluation. *Ieee Transactions on Audio, Speech, and Language Processing*, **15**(4), 1247–1256. doi: 10.1109/TASL.2006.889797. URL https://academiccommons.columbia.edu/doi/10.7916/D8NC69RK/download.

Raffel, C., Mcfee, B., Humphrey, E. J., Salamon, J., Nieto, O., Liang, D. a Ellis, D. P. W. (2014). mir_eval: A Transparent Implementation of Common MIR Metrics. *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014)*, pages 367–372. URL https://bmcfee.github.io/papers/ismir2014{_}mireval.pdf.

Rao, V. a Rao, P. (2010). Vocal melody extraction in the presence of pitched accompaniment in polyphonic music. *IEEE Transactions on Audio, Speech and Language Processing*, **18**(8), 2145–2154. ISSN 15587916. doi: 10.1109/TASL.2010.2042124. URL https://www.ee.iitb.ac.in/course/{~}daplab/publications/PrePrintForWebsite.pdf.

Ryynänen, M. P. a Klapuri, A. P. (2008). Automatic Transcription of Melody, Bass Line, and Chords in Polyphonic Music. *Computer Music Journal*, **32**(3), 72–86. ISSN 0148-9267. doi: 10.1162/comj.2008.32.3.72.

Salamon, J. a Gomez, E. (2012). Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Transactions on Audio, Speech and Language Processing*, **20**(6), 1759–1770. ISSN 15587916. doi: 10.1109/TASL.2012.2188515.

Salamon, J., Rocha, B. a Gomez, E. (2012). Musical genre classification using melody features extracted from polyphonic music signals. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 81–84. ISSN 15206149. doi: 10.1109/ICASSP.2012.6287822. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.296.5021{&}rep=rep1{&}type=pdf.

Salamon, J., Gómez, E., Ellis, D. P. a Richard, G. (2014). Melody extraction from polyphonic music signals: Approaches, applications, and challenges. *IEEE Signal Processing Magazine*, **31**(2), 118–134. ISSN 10535888. doi: 10.1109/MSP.2013.2271648. URL http://www.justinsalamon.com/uploads/4/3/9/4/4394963/salamon{_}gomez{_}ellis{_}richard{_}melodyextractionreview{_}ieeespm{_}2013.pdf.

Salamon, J., Bittner, R. M., Bonada, J., Bosch, J. J., Gomez, E. a juan pablo Bello (2017). An Analysis/Synthesis Framework for Automatic F0 Annotation of Multitrack Datasets. *Proceedings of the International Society for Music Information Retrieval {(ISMIR)} Conference*, pages 71–78.

Stoller, D., Ewert, S. a Dixon, S. (2018). Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation. pages 334–340. ISSN 13514180. doi: arXiv:1806.03185v1. URL http://arxiv.org/abs/1806.03185.

Sturm, B. L. (2013). Classification accuracy is not enough. *Journal of Intelligent Information Systems*, **41**(3), 371–406. ISSN 0925-9902. doi: 10.1007/s10844-013-0250-y. URL https://link.springer.com/content/pdf/10.1007{%}2Fs10844-013-0250-y.pdf.

Tachibana, H., Ono, T., Ono, N. a Sagayama, S. (2010). Melody line estimation in homophonic music audio signals based on temporal-variability of melodic source. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (May 2014), 425–428. ISSN 15206149. doi: 10.1109/ICASSP.2010.5495764.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. a Kavukcuoglu, K. (2016). WaveNet: A Generative Model for Raw Audio. pages 1–15. URL http://arxiv.org/abs/1609.03499.

Yang, L.-C., Chou, S.-Y. a Yang, Y.-H. (2017). MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation. URL http://arxiv.org/abs/1703.10847.

Seznam obrázků

Seznam tabulek

Seznam použitých zkratek

Přílohy