Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Jan Jug

Transkripcija klavirske glasbe z globokim učenjem

diplomsko delo

univerzitetni študijski program prve stopnje
računalništvo in informatika

Mentor: doc. dr. Matija Marolt

Ljubljana 2015

Besedilo je oblikovano z urejevalnikom besedil LaTeX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

V delu preučite področje transkripcije glasbenih posnetkov z nevronskimi mrežami. Osredotočite se na klavirsko glasbo in preizkusite kako se za transkripcijo obnesejo različni modeli mrež: večnivojski perceptron, konvolucijska nevronska mreža in globoka verjetnostna mreža. Modele ovrednotite in primerjajte na standardni zbirki za transkripcijo klavirske glasbe MAPS.

Izjava o avtorstvu diplomskega dela

Spodaj podpisani Jan Jug z vpisno številko 63110241 sem avtor diplomskega dela z naslovom:

Transkripcija klavirske glasbe z globokim učenjem

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Matije Marolta,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 26. avgusta 2015 Podpis avtorja:

Zahvaljujem se predvsem svoji punci Maši, ker mi je vso pot stala ob strani in mi pomagala po svojih najboljših močeh.

Zahvaljujem se tudi mentorju za vso strokovno pomoč, sestri Katji za lingvistične popravke ter staršem in šefu, ker so razumeli mojo zaposlenost in odsotnost.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
AMT	automatic music transcription	avtomatična transkripcija glasbe
DFT	Discrete Fourier Transform	diskretna Fourierova transformacija
FFT	Fast Fourier Transform	hitra Fourierova transformacija
CQT	Constant Q Transform	transformacija s konstantnim Q
MLP	multilayer perceptron	večnivojski perceptron
CNN	convolutional neural network	konvolucijska nevronska mreža
RBM	restricted Boltzmann machine	omejeni Boltzmannov stroj
DBN	deep belief network	globoka verjetnostna mreža

Povzetek

Transkripcija glasbe je zahteven postopek simboličnega zapisa glasbenega posnetka. Cilj tega diplomskega dela je bila preučitev transkripcije klavirske glasbe z metodami globokega učenja, za kar so bili implementirani in preizkušeni trije modeli globokih nevronskih mrež: večnivojski perceptron, konvolucijska nevronska mreža in globoka verjetnostna mreža. Z modelom globoke verjetnostne mreže je bilo preizkušeno nenadzorovano predučenje, katerega namen je izluščenje glasbenih značilnosti iz zvočnega signala. Učenje modelov in preverjanje končne uspešnosti transkripcije je bilo izvedeno na zbirki za transkripcijo klavirske glasbe MAPS. Izvedena je bila tudi primerjava predpriprave podatkov s transformacijama hitre Fourierove transformacije in transformacije s konstantnim Q. Končni rezultati so pokazali, da je globoko učenje s pravim učnim načrtom lahko močno orodje za transkripcijo glasbe.

Ključne besede: avtomatična transkripcija glasbe, globoke nevronske mreže, klavirska glasba, globoko učenje, večnivojski perceptron, konvolucijska nevronska mreža, globoka verjetnostna mreža, hitra Fourierova transformacija, transformacija s konstantnim Q.

Abstract

Transcription of music is a complex process of transcribing an audio recording into a symbolic notation. The goal of this thesis was to examine transcription of piano music with deep learning, for which three models of deep neural networks were implemented: multilayer perceptron, convolutional neural network and deep belief network. Through the use of deep belief network, unsupervised pretraining for automatic extraction of musical features from audio signals was also tested. Learning of these models and evaluation of transcription was performed with MAPS database for piano music transcription. A comparison between Fast Fourier Transform and Constant Q Transform for data pre-processing was also carried out. Final results show that deep learning with an appropriate learning schedule is potentially a powerful tool for automatic transcription of music.

Keywords: automatic music transcription, deep neural networks, piano music, deep learning, multilayer perceptron, convolutional neural network, deep belief network, Fast Fourier Transform, Constant Q Transform.

Poglavje 1

Uvod

Največja uganka današnje znanosti so še vedni človeški možgani[1]. Njihova procesna moč je neprimerljivo večja od najzmogljivejših računalnikov, kljub temu, da so v sami osnovi počasnejši, saj računalniki operirajo na nivoju nanosekund (10^{-9} sekunde), medtem ko so možgani nekje na nivoju milisekund (10^{-3} sekunde). A ta zaostanek nadoknadijo s svojo kompleksno, nelinearno in visoko paralelno zgradbo, ki nam omogoča, da različne življenjske naloge rešujemo veliko hitreje in učinkoviteje kot računalnik. Eden izmed mnogih takih primerov je naš slušni sistem, ki v realnem času odlično procesira in dojema informacije, prejete iz okolice preko vibracij v zraku. Najboljši računalniški algoritmi, ki se ukvarjajo z nalogami slušnega sistema, ne pridejo niti blizu zmogljivosti izučenega ušesa. Razlog za to je popolnoma drugačna struktura in namen računalnika v primerjavi z možgani; računalnik je bil izumljen v namene reševanja matematičnih operacij, možgani pa so se razvili za preživetje v naravi, zaradi česar so morali razviti predvsem sposobnosti prepoznavanja vzorcev, dojemanja, motorike in učenja. Učenje je izredno pomemben mehanizem, saj se preko učenja možgani prilagajajo zunanjemu svetu in pridejo do novih spoznanj in novega znanja.

Ljudje smo iz nepojasnenih razlogov že od časa pred začetki civilizacij

[1]Seveda tudi možgani ostalih živalskih vrst - v nadaljevanju se bomo nanašali predvsem na človeške možgane, saj so bolj relevantni tematiki tega dela.

navdušeni nad glasbo. To dejstvo je izjemno zanimivo, saj glasba s fizikalnega vidika ni nič drugega kot hrup, ki je ravno tako urejen, da je všeč našim ušesom. Ko je človek izumil pisavo, da je lahko zapisal svoje misli in besede, je eventuelno izumil tudi način zapisa glasbe, da je ohranil, kar mu je bilo všeč. Najstarejši najdeni zapis glasbe je bil napisan v klinopisu v asirskem mestu Ugarit in sega nazaj v 15.-14. stoletje pred našim štetjem [16], kar kaže na dejstvo, da je že človek zgodnjih civilizacij gojil zanimanje za glasbo. Postopek zapisa poslušane glasbe se imenuje transkripcija in je izredno kompleksen, a možgani se lahko naučijo prepoznavati različne tone in to uporabiti pri poustvarjanju glasbe. Popolna transkripcija seveda ni vedno mogoča in je odvisna od števila instrumentov, stopnje polifonije[2], glasnosti, prisotnosti šuma in še mnogih podobnih dejavnikov, ki se združujejo v to, čemur pravimo zvok. Kako sposoben je naš slušni sistem, dobro ponazori analogija jezera: predstavljajmo si, da bi pravokotno na obalo nekega jezera izkopali dva kratka kanala in čez površino vode položili v vsakega en robček. Človeški sluh je potem podoben določanju števila plovil na jezeru, njihovih smeri in hitrosti, katero je bolj in katero manj oddaljeno in še marsičesa samo iz opazovanja vibracij na teh dveh robčkih [3]. Kljub kompleksnosti te naloge naši možgani nimajo popolnoma nobene težave dojemanja vsega tega v realnem času. Zelo priročno bi bilo, če bi to znal tudi računalnik, saj bi lahko preko različnih tipal zaznal še več in tako prekoračil omejitve biološkega slušnega sistema. Človek je omejen že z dejstvom, da so ušesa le kratki kanali, v katerih se mnogo dimenzij zvoka izgubi. Omejeni smo tudi s svojim slušnim in bolečinskim pragom, saj lahko dojemamo le frekvence v razponu od 20 Hz do okoli 20 kHz, previsoke glasnosti pa nam povzročajo bolečino. Računalnik, po drugi strani, pa lahko preko različnih senzorjev zaznava veliko več raznolikih signalov, težava je le v tem, da iz vsega tega ne zna potegniti uporabnih informacij; z drugimi besedami - zvoka ne dojema. Potreben je človek, da iz zbranih podatkov razbere vsebino, kar pa ni priročno, saj je človek prepočasen za ogromne količine zajetih podatkov.

[2]Število sočasnih tonov.

Od tu izvira motivacija, da naučimo računalnik, da sam avtomatično razbere vsebino in pomen zvoka, ki ga zaznava (če se lahko tako izrazimo). Temu postopku pravimo *avtomatična transkripcija glasbe* (angleško *automatic music transcription* ali AMT).

Avtomatično transkripcijo glasbe bi tako lahko razumeli kot neko osnovno dojemanje glasbe. Računalnik iz nabranih zvočnih podatkov razbere njihove značilnosti in se na podlagi le-teh nauči razlikovati med toni. Tako glasbi že doda nek pomen in ni več le golo zaporedje bitov, temveč zaporedje tonov. To zaporedje tonov pa se lahko potem uporabi za rekonstrukcijo glasbenega dela, bodisi v obliki notnega črtovja za namene človeka bodisi preko uporabe sintetizatorja zvoka. Ker lahko transkripcijo opišemo tudi kot postopek klasifikacije tonov, se v ta namen med drugim uporablja tudi različne metode razvrščanja v razrede, ki se veliko uporabljajo na mnogih drugih področjih raziskovanja.

Postopek transkripcije na splošno sestavljajo trije deli:

- *Izračun frekvenčnega spektra*: ker nam informacija o amplitudi zvoka, ki je računalniku najbolj pomembna pri predvajanju zvoka in tako uporabljena za digitalni zapis glasbe, ne pove ničesar o tonalni sestavi zvoka, moramo signal najprej prestaviti v frekvenčni prostor, z drugimi besedami izračunati njegov frekvenčni spekter. V ta namen se največ uporablja metode na podlagi Fourierove transformacije.

- *Zaznavanje in razlikovanje med toni*: tu se večinoma uporablja različne metode klasifikacije, kjer na podlagi značilnosti posameznega tona določimo, v kateri razred spada.

- *Post-procesiranje in končni zapis*: po določitvi tonov je navadno potrebno rezultat še naknadno obdelati z uporabo znanja glasbene teorije, da izločimo napake, ki niso logične in se ne povežejo z znanimi glasbenimi vzorci (primer: zaznan hiter ton izven glasbenega ključa, ki se pojavi v bolj počasnem delu skladbe). Na koncu je potrebno rezultat celotnega postopka še vrniti v neki razumljivi obliki za nadaljnjo

uporabo, recimo v obliki notnega črtovja ali v MIDI zapisu.

To diplomsko delo se osredotoča na drugi korak, zaznavanje tonov, transkripcije klavirske glasbe. Izračun frekvenčnega spektra je bil vseeno potreben, za kar sta bili uporabljeni dve različni transformaciji, *hitra Fourierova transformacija* (angleško *Fast Fourier Transform* ali FFT) in *transformacija s konstantnim Q* (angleško *Constant Q Transform* ali CQT). Za razlikovanje med toni so bili uporabljeni različni modeli globokih nevronskih mrež. Nevronska mreža je visoko vzporeden in porazdeljen procesor, sestavljen iz preprostih osnovnih enot — *nevronov* — v katerem je znanje pridobljeno preko postopka učenja in shranjeno v povezavah med posameznimi enotami [6]. Ideja za to je prišla iz bioloških nevronskih mrež živčnih sistemov živali, kjer se znanje prav tako pridobiva preko učenja in je shranjeno v sinaptičnih povezavah med nevroni. Zaradi te podobnosti z biološkimi modeli dojemanja so nevronske mreže zelo primerne za naloge, ki jih človek brez težav opravlja vsak dan, računalniku pa predstavljajo velike težave, kot recimo razbiranje vsebine iz slike ali zvoka preko prepoznavanja vzorcev. Obstaja veliko različnih modelov nevronskih mrež, ki so uporabni na različnih področjih, v tem diplomskem delu pa smo osredotočili na globoke modele, ki se od navadnih razlikujejo po tem, da so zgrajeni v več nivojih in so tako "globlji". Dodatni nivoji omogočajo prepoznavanje bolj abstraktnih značilnosti, a tudi povečajo težavnost učenja. Učenje globokih nevronskih mrež imenujemo *globoko učenje*. Uporabili smo tri modele globokega učenja:

- *Večnivojski perceptron* (angleško *multilayer perceptron* ali MLP): najbolj osnoven model globoke arhitekture, na katerega lahko gledamo kot na logistično regresijo z dodatnim nivojem, ki nelinearno transformira vhod.

- *Konvolucijska nevronska mreža* (angleško *convolutional neural network* ali CNN): model nevronske mreže, ki izvira iz študij biološkega vidnega sistema in uporablja matematično operacijo konvolucije za zmanjševanje kompleksnosti vhoda in samega modela.

- *Globoka verjetnostna mreža* (angleško *deep belief network* ali DBN): model mreže, ki se preko statističnih metod iz podatkov najprej nenadzorovano nauči njihovih značilnosti, nato pa te značilnosti uporabi pri nadaljnjem učenju.

Naknadni obdelavi se v tem delu nismo posvečali, saj nas je zanimala predvsem uporabnost globokih mrež pri zaznavanju tonov in ne toliko doseg najboljšega rezultata.

Zanimalo nas je, ali so globoke nevronske mreže primeren model za transkripcijo klavirske glasbe in pričakovali smo, da bosta naprednejša modela konvolucijske nevronske mreže in globoke verjetnostne mreže pri tej nalogi uspešnejša od osnovnega modela večnivojskega perceptrona. Poskusi so te predpostavke potrdili in pokazali, da dodatni nivoji konvolucije in nenadzorovanega učenja v nevronskih mrežah omogočijo globlje razumevanje zvočnih podatkov. Zanimalo nas je tudi, katera predpriprava podatkov (transformacija FFT ali CQT) je primernejša za globoko učenje, in ugotovili, da imata obe svoje prednosti in slabosti, a se na splošno bolje obnese FFT.

Poglavje 2

Ozadje

2.1 Transkripcija glasbe

Transkripcija glasbe je postopek zapisa poslušanega ali posnetega glasbenega dela. Transkripcija je tako prenos glasbe iz začasne, zvočne predstavitve v trajno, simbolično predstavitev, preko katere je mogoče zvok glasbenega dela ponoviti. A transkripcije ne smemo mešati z zajemom zvoka, kjer zvok zajamemo z mikrofonom, zapišemo na nek medij, nato pa ga lahko poslušamo na zvočnem sistemu. Tudi tako je možno zvok glasbenega dela ponoviti, vendar pa to deluje na najbolj osnovni ravni reprodukcije zajetih zvočnih valov in je za človeka brezvsebinsko. Zato mora biti simbolična predstavitev na višji ravni abstrakcije od golih zvočnih valov in mora zapisati informacije o tempu, ritmu, tonih, njihovemu trajanju in tako naprej. Tak zapis je človeku razumljiv in mu omogoča poustvarjanje danega glasbenega dela.

Raziskovanje avtomatične transkripcije glasbe se je začelo v 70-ih letih 20. stoletja, ko so računalniki končno postali dovolj zmogljivi, da so lahko predelali ogromne količine zvočnih podatkov. Izkazalo se je, da je transkripcija monofonične glasbe dokaj trivialen problem, na resnejše težave pa naletimo ko se lotimo polifonične glasbe, saj se več tonov naenkrat zlije skupaj in tako popači zvočni signal. Prvi sistemi transkripcije so bili tako omejeni na dvoglasno glasbo z mnogimi drugimi omejitvami. Skozi 90. leta je

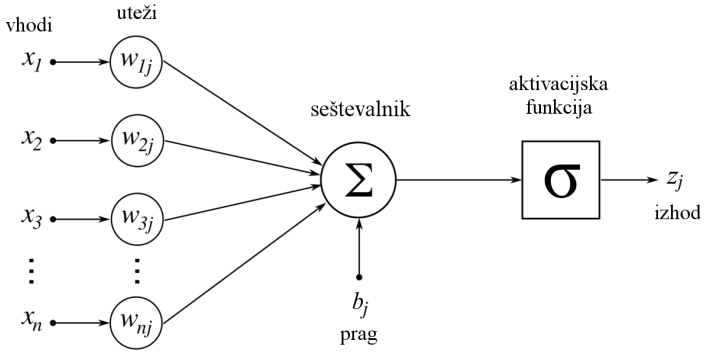
zanimanje za transkripcijo še raslo in nastalo je mnogo različnih sistemov le-te. Poleg statističnih metod [5, 2] so se za uporabne izkazale tudi metode, ki računsko modelirajo slušni sistem [7], in metode nenadzorovanega učenja [13], ki značilnosti v zvoku iščejo same. V zadnjem času se raziskuje tudi uporabo nevronskih mrež v ta namen [1, 11]. Kljub napredku pa so najboljši sistemi še vedno veliko slabši od izučenega ušesa, tako pri natančnosti kot pri prilagodljivosti. Današnji transkripcijski sistemi so namreč še vedno omejeni v svoji kompleksnosti, kot recimo omejeno število sočasnih tonov ali usmerjenost na posamezen tip instrumenta. To pa ni presenetljivo, saj je transkripcija zelo težka naloga že za človeškega poznavalca, ki potrebuje več poslušanj in se zanaša na leta izkušenj, da samo približno zapiše potek glasbenega dela, saj popolna transkripcija ni vedno mogoča. Transkripcija glasbe tako nikakor ni trivialna naloga in kljub stalnemu napredku in iskanju novih metod še vedno ni nobene splošno uporabne metode, ampak razvoj še vedno poteka znotraj omejitev, kot je usmerjenost na določene žanre ali instrumente.

2.2 Nevronske mreže

Umetne nevronske mreže so rezultat raziskovanja na mnogih področjih znanosti in tako združujejo vede nevroznanosti, matematike, statistike, fizike in računalništva. So vsestransko uporabne, saj jih najdemo na področjih modeliranja, prepoznave vzorcev, digitalnega procesiranja signalov, časovne analize (*time series analysis*) in tako naprej. Njihova glavna lastnost je zmožnost učenja, kar jih približa biološkemu modelu možganov. Osnovna procesna enota nevronskih mrež je *nevron* — matematični konstrukt, ki se obnaša podobno kot prave nevronske celice. Nevron si lahko predstavljamo kot preprost element, ki na podlagi vhodov (sinaps) ali ostane neaktiven ali pa se aktivira in odda nek odziv in tako pridoda svoj delež dela v nevronski mreži. Posamezen nevron je preveč preprost, da bi bil kakor koli uporaben, več nevronov povezanih v mrežo pa se izkaže za zelo močan sistem za proce-

siranje podatkov. Leta 1943 sta psihiater Warren McCulloch in matematik Walter Pitts dokazala, da je mogoče z dovolj medsebojno povezanimi nevroni izračunati vsako izračunljivo funkcijo [6]. Vseeno pa nevronske mreže niso niti približen nadomestek pravim možganom; so samo model, ki lahko posnema opravljanje posameznih preprostejših nalog.

Kot pri možganih je tudi pri nevronskih mrežah njihova glavna lastnost sposobnost učenja, zaradi česar nevronske mreže uvrščamo med metode strojnega učenja. Učenje lahko poteka nadzorovano ali nenadzorovano. Pri nadzorovanem učenju ob vsakem učnem primeru tudi povemo, kaj naj bi ta primer predstavljal, torej pustimo, da mreža nekaj predpostavi, nato pa ji povemo, ali se je zmotila ali ne. V vsakem primeru se mreža posodobi in tako v primeru napake kaznuje parametre, ki so vodili do odločitve, v primeru pravilne napovedi pa jih nagradi. Pri nenadzorovanem učenju pa mreži pustimo, da značilnosti in zakonitosti v učnem primeru odkrije sama, skozi različne statistične metode. Ta način se ponavadi uporablja kot predučenje ali pa kot zmanjševanje dimenzionalnosti, po katerem potem pride še nadzorovano učenje.



Slika 2.1: Primer nevrona. Vhodi in uteži predstavljajo njegove sinapse, seštevalnik in aktivacijska funkcija pa na padlagi le-teh določita njegovo aktivacijo.

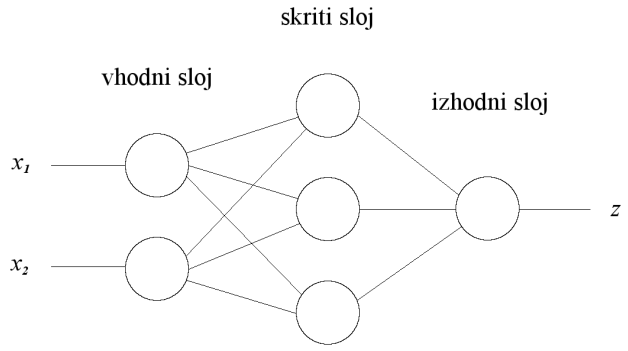
Nevron je osnovna procesna enota nevronskih mrež, sestavljena iz vhodnih povezav, seštevalnika, praga in aktivacijske funkcije. Enačba splošnega

nevrona je

$$z = \sigma(\mathbf{x} \cdot \mathbf{w} + b), \tag{2.1}$$

kjer σ predstavlja aktivacijsko funkcijo, $\mathbf{x}$ je vhodni vektor, $\mathbf{w}$ je vektor uteži tega nevrona in b njegov prag. Namen aktivacijske funkcije je omejitev izhoda na določen interval, recimo [0, 1] v primeru sigmoidne funkcije, in tako preprečimo, da bi nekateri nevroni z zelo visokimi izhodi popolnoma zasenčili ostale. Na sliki 2.1 je grafična prestavitev nevrona, implementiranega po enačbi (2.1). Zadostno število takih preprostih procesnih enot, povezanih v mrežo lahko izračuna vsako izračunljivo funkcijo.

2.2.1 Globoke arhitekture



Slika 2.2: Primer preproste nevronske mreže z dvema vhodoma, enim skritim nivojem in enim izhodom.

Globoke nevronske mreže se od navadnih razlikujejo le po tem, da imajo več skritih nivojev. *Nivo* je množica vzporednih nevronov, ki med seboj nimajo povezav, povezani pa so različni nivoji med seboj, kot lahko vidimo na sliki 2.2. Prve nevronske mreže skritih nivojev niso imele, saj še ni bilo poznano pravilo vzvratnega razširjanja napake (angleško *backward propagation of error* ali krajše *backpropagation*), ki nam omogoča učenje preko več nivojev. Dodatni nivoji nam omogočajo reševanje nelinearnih problemov [8], poleg

tega pa nam zvišajo abstrakcijo zaznavanja, saj se lahko prvi nivo nevronske mreže za prepoznavanje rokopisanih števk nauči na primer odkrivanja ostrih osnovnih potez, drugi nivo nato prepoznavanja zank in nadaljnji nivoji še kaj kompleksnejšega. A več nivojev je težje učiti, in po navadi zadostujeta dva ali trije, čeprav imajo nekatere večje kompleksne mreže lahko tudi do 11 nivojev [4]. Učenje je težje, ker je treba imeti zadosti podatkov, da lahko mreža zajame splošno predstavitev v tako velikem številu parametrov. V postopku učenja je namreč potrebno prilagajati uteži in prage vseh nevronov v mreži, kar pri globokih mrežah lahko pomeni izredno veliko število parametrov. Če je parametrov preveč, podatkov pa premalo, se bo mreža naučila le predstavitve teh podatkov, na drugih podobnih vzorcih pa bo veliko slabša; takrat pravimo da pride do prevelikega prileganja (angleško *overfitting*). Ker se učenje vedno izvaja z omejeno količino podatkov, bo do prevelikega prileganja vedno prišlo, zato je pomembno, da ga karseda zmanjšamo.

Poglavje 3

Postopki in metode dela

3.1 Podatki

Za učenje kateregakoli modela strojnega učenja potrebujemo veliko zbirko uravnoteženih podatkov, da lahko model razbere pravo reprezentacijo le-teh. V tem delu smo uporabili zbirko anotirane klavirske glasbe MAPS. V njej so glasbene datoteke v WAVE formatu ter poravnane MIDI in tekstovne datoteke različnih posnetkov klavirskih zvokov. Glasbene datoteke vsebujejo dvokanalne 16-bitne posnetke pri hitrosti vzorčenja 44,1 kHz. V tekstovnih datotekah so zapisani časi začetkov in koncev posameznih not, kar smo uporabili za določitev osnovne resnice pri nadzorovanem učenju.

Zbirka je razdeljena na 4 podzbirke: posamezni toni, pogosti akordi v zahodni glasbi, naključni akordi in glasbena dela. Posamezni toni in akordi so še dalje razdeljeni na različne variacije, kot je različno število tonov v sozvočju in različni slogi igranja (*staccato*, lestvice, ...). V teh posnetkih so vsi toni predstavljeni približno enakomerno in tako zelo primerni v namene strojnega učenja. Podatke smo naključno razdelili na 2 podmnožici, učna množica z 80 % podatkov in testna množica s preostalimi 20 %. Da lahko objektivno ocenimo učinkovitost učnega modela, moramo za preverjanje uporabiti še nevidene podatke, zato morata biti množici strogo ločeni.

3.2 Priprava podatkov

Transkripcija glasbe je sestavljena iz več delov, tako kot ima glasba več lastnosti. Določiti je potrebno tempo, ritem ter tone in njihovo trajanje, da lahko razberemo melodijo. Ker so si te lastnosti različne, obstajajo različni načini njihovega odkrivanja. V tem diplomskem delu smo se ukvarjali z določanjem tona, z ostalimi lastnostmi pa se nismo ukvarjali. Ton je lastnost glasbe, ki je tesno povezana s frekvenco zvočnega signala. Ko posnamemo nek zvočni posnetek in ga shranimo na računalnik, zajamemo zvočne valove signala in jih shranimo kot binarni tok podatkov o amplitudah teh valov, urejenih po času. Amplituda sama pa nima veliko informacij o tonu, zato potrebujemo informacijo o frekvenci, torej o nihanju amplitude skozi čas, ki jo pridobimo tako, da zvočni signal prestavimo iz časovnega v frekvenčni prostor preko ene izmed mnogih Fourierovih transformacij. Uporabili smo najbolj splošno uporabljeno *diskretno Fourierovo transformacijo* (v nadaljevanju DFT) oziroma njeni različici *hitro Fourierovo transformacijo* (v nadaljevanju FFT) in *transformacijo s konstantnim Q* (v nadaljevanju CQT).

3.2.1 DFT in algoritem FFT

Diskretna Fourierova transformacija je diskretna oblika izračuna Fourierove vrste. Ker so podatki v računalniku digitalni, je DFT logična izbira za obdelavo le-teh, poleg tega pa jo je mogoče tudi učinkovito implementirati v algoritmu *hitre Fourierove transformacije* (v nadaljevanju FFT), katerega računska kompleksnost je $O(n \log n)$. Zaradi teh dveh pomembnih lastnosti je FFT daleč najbolj splošno uporabljen algoritem na področju analize zvoka.

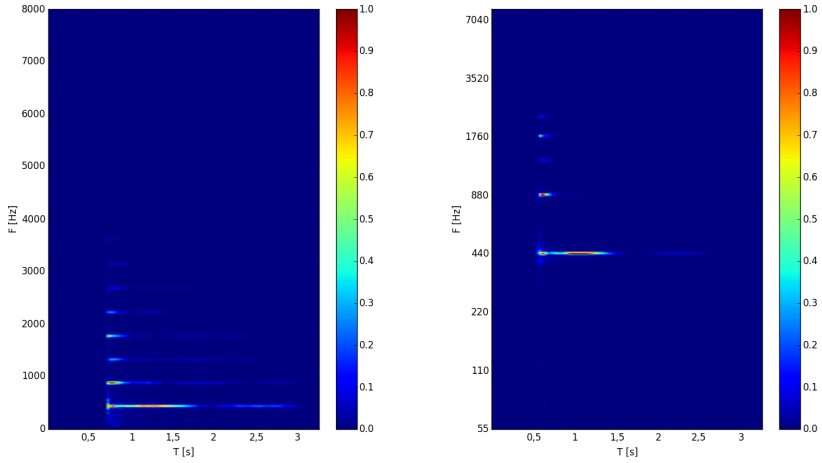
$$F_n \equiv \sum_{k=0}^{N-1} f_k e^{-2\pi i n k/N} \tag{3.1}$$

Iz definicije DFT v enačbi (3.1) lahko razberemo, da so njena zaloga vrednosti kompleksna števila (Eulerjeva oblika zapisa kompleksnih števil $e^{i\theta}$). Da lahko signal natančno predstavimo v frekvenčnem prostoru potrebujemo informacijo o amplitudi in fazi, kar lahko predstavimo s kompleksnim številom,

v katerem realna komponenta predstavlja amplitudo, imaginarna pa fazo. V namene zaznave tonov informacija o fazi ni pomembna, zato rezultat prestavimo nazaj v realna števila tako, da izračunamo moč po standardni formuli $|z| = \sqrt{(\mathrm{Im}(z) + \mathrm{Re}(z))^2}$.

DFT prenese signal v frekvenčni spekter tako, da ga primerja s sinusoidami različnih frekvenc in izračuna koeficiente prileganja signala s temi sinusoidami. Razpon frekvenc je omejen, kajti po Nyquist-Shannonovemu izreku o vzorčenju ne moremo zaznati frekvenc, ki so večje od polovice hitrosti vzorčenja. Pri vzorčenju s 44100 vzorci na sekundo oziroma 44,1 kHz tako ni mogoče shraniti frekvenc višjih od 22050 Hz. Druga omejitev pa je resolucija frekvenčnega spektra, ki je odvisna od tega, kako dolgo okno vzamemo za transformacijo. Pretvorbe iz časovnega v frekvenčni prostor se izvajajo po oknih, ki predstavljajo kratek časovni okvir signala. Velikost tega okvirja določa frekvenčno resolucijo, saj po transformaciji dobimo enako število koeficientov, kot je bilo vzorcev v vhodu. Od teh koeficientov pa jih je polovica odvečnih, saj po prej omenjenemu izreku o vzorčenju ne moremo izračunati koeficientov frekvenc višjih od polovice dolžine okna. Če iz posnetka s hitrostjo vzorčenja 44,1 kHz vzamemo okno dolžine 2048 vzorcev, bomo po transformaciji dobili 1025 ($2048/2 + 1$[1]) koeficientov frekvenc. Te frekvence so enakomerno porazdeljene čez celo domeno frekvenc, v tem primeru do 22050 Hz. En koeficient torej pokriva frekvence v razponu približno 21,5 HZ ($\approx$ 22050 Hz/1025 koeficientov). Za polovico boljšo resolucijo lahko dosežemo, če uporabimo dvakrat daljša okna. Daljša okna pa nam poslabšajo časovno resolucijo, saj zajamejo večji časovni razpon, zato je potrebno doseči kompromis med frekvenčno in časovno natančnostjo. Če ta okna zložimo enega zraven drugega, dobimo 2D sliko frekvenčnega spektra (primer na sliki 3.1).

[1]Sredinski koeficient Nyquistove frekvence, ki leži na polovici hitrosti vzorčenja.



Slika 3.1: Primerjava transformacij FFT in CQT na tonu A4. Levo: frekvenčni spekter transformacije FFT na linearni lestvici. Desno: frekvenčni spekter transformacije CQT na logaritmični lestvici.

3.2.2 CQT

CQT je kratica za angleški izraz *Constant Q Transform*, kar bi lahko nekako prevedli v *transformacija s konstantnim Q*. V sami osnovi je še vedno DFT, ki pa frekvence predstavi na logaritmični lestvici. Ideja prihaja iz biološkega zaznavnega sistema, ki signale dojema na logaritmični lestvici. Tako je frekvenca f_1, ki je eno oktavo nad frekvenco f_0, dvakrat višja od slednje, $f_1 = 2f_0$. Zaradi tega imajo višje oktave veliko večje razpone frekvenc kot nižje in so absolutne razlike med toni večje kot v nižjih oktavah, kljub dejstvu da človek razlike med toni dojema enako, ne glede na oktavo. DFT vrne frekvenčno predstavitev na linearni lestvici in tako je večina informacije strnjena v spodnjem koncu spektra, kjer so si toni bližje, v zgornjem delu pa je porazdelitev vedno redkejša. CQT je prilagoditev DFT-ja, ki odpravi to neenakost in frekvence predstavi enakomerno. To enakomerno razporeditev doseže tako, da izračuna DFT signala različnih dolžin in iz rezultatov prebere logaritmično razporejene podatke o frekvencah. Končni rezultat je frekvenčni spekter na logaritmični lestvici.

CQT odpravi redčenje frekvenčnega spektra z naraščanjem frekvence, a ima še vedno težavo z dolžinami okna za zajem nižjih frekvenc. Le-te imajo namreč veliko večji nihajni čas in tako potrebujemo veliko daljša okna, da zajamemo njihovo valovanje, s tem pa poslabšamo časovno natančnost.

Predstavitev signala na logaritmični lestvici je zelo koristna tudi zato, ker nam zmanjša dimenzionalnost signala, saj odstrani odvečne koeficiente frekvenc med posameznimi toni, ki niso pomembne. Zaradi tega pa niso več tako opazne relacije med toni in tudi višji harmoniki se lahko izgubijo. Harmoniki tona so frekvence, ki so večkratniki frekvence tega tona, in na sliki 3.1 opazimo, da DFT bolje predstavi to relacijo, medtem ko je pri CQT že bolj zabrisana.

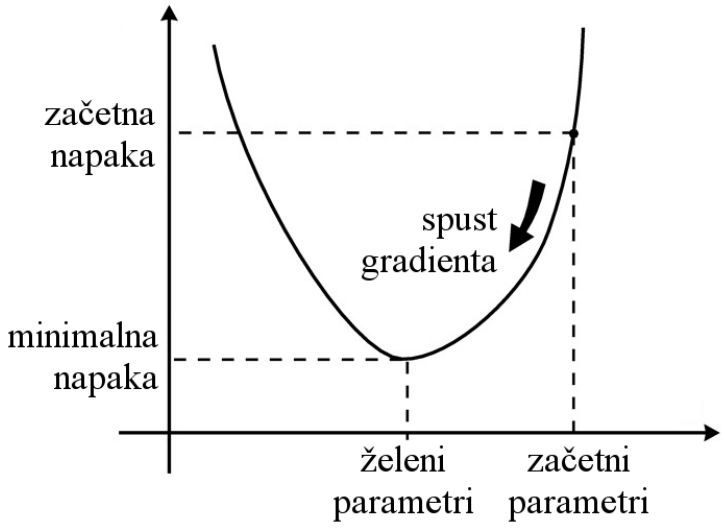
3.3 Modeli učenja

Za zaznavanje tonov smo implementirali in uporabili 3 modele globokega učenja. Za osnovne rezultate in primerjavo smo najprej naučili najbolj osnoven model globoke arhitekture, večnivojski perceptron. Drugi model je bila konvolucijska nevronska mreža, model zasnovan predvsem za prepoznavo vzorcev v slikah, a se je izkazal za uporabnega tudi pri zvočni analizi [9, 10, 12, 15]. Tretji model pa je globoka verjetnostna mreža, kot bi lahko nekako prevedli angleški izraz *deep belief networks*, ki predstavitev podatkov pridobi preko nenadzorovanega predučenja in nadzorovanega učenja po njem.

3.3.1 Večnivojski perceptron

Večnivojski perceptron (v nadaljevanju MLP) je klasifikator, ki znanje pridobiva preko nadzorovanega strojnega učenja. Sestavljen je iz vhodnega, poljubnega števila skritih in izhodnega nivoja, začetni parametri (uteži in pragi nevronov) so inicializirani naključno. Učenje poteka preko prilagajanja teh parametrov glede na napako med izhodom modela in pravo oznako učnega vzorca, oznako ki bi jo dobro naučeni model moral vrniti. Najbolj

osnoven in splošno uporabljan algoritem učenja je stohastični spust po gradientu (angleško *stochastic gradient descent* ali SGD).



Slika 3.2: Grafična predstavitev spusta po gradientu. Na ordinatni osi imamo napako, na abscisni pa parametre učnega modela.

Na nadzorovano učenje lahko gledamo kot na manjšanje napake modela in tako lahko nanj gledamo kot na optimizacijski problem. Model bo najučinkovitejši, ko bo njegova napaka najmanjša, kar pomeni, da moramo najti globalni minimum neke cenilne funkcije C, ki nam predstavlja razliko med izhodom modela in pričakovanim izhodom. Pri funkcijah ene spremenljivke minimum preprosto najdemo tako, da izračunamo vrednost funkcije pri ničlah prvega odvoda, a parametrov nevronske mreže je veliko in cenilna funkcija C je funkcija več spremenljivk. Posplošitev odvoda za funkcije večih parametrov se imenuje gradient in nam poleg naklona funkcije pove tudi smer le-tega. Če si minimum cenilne funkcije predstavljamo kot dno doline, si lahko spust po gradientu predstavljamo kot spust po steni te doline (glej grafični primer na sliki 3.2). Pri naključni inicializaciji začetnih parametrov smo postavljeni na naključno mesto na površini cenilne funkcije v njenem funkcijskem prostoru in na tem mestu lahko izračunamo smer naklona ter se

nato za majhen delež premaknemo v nasprotni smeri le-tega, podobno kot da bi se spustili v dolino. Temu majhnemu deležu pravimo *stopnja učenja* (angleško *learning rate*) in ga določimo kot hiperparameter[2] pred začetkom učenja, tipične vrednosti pa so med 10^{-6} in 10^{-3}. Posodabljanja ni potrebno početi pri vsakem učnem primeru, ampak lahko izračunamo gradient na večih primerih skupaj v mini-seriji (angleško *minibatch*) in nato uporabimo povprečje za dejansko posodobitev parametrov. Tako dobimo *stohastični* spust po gradientu.

SGD je iterativna metoda učenja. V vsaki iteraciji posodobimo parametre, ki izboljšajo učinkovitost modela. Ko zaključimo ponavljanje nad celotno zbirko podatkov, pravimo da smo naredili en prehod (angleško *epoch*) čez učne podatke. Samo ena ponovitev navadno ne zadošča, zato takrat učenje traja več prehodov.

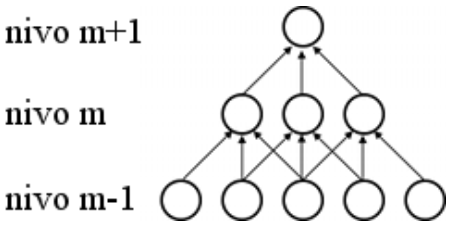
Poleg SGD-ja obstajajo tudi druge optimizacijske metode, ki so uporabne za učenje, a v tem diplomskem delu smo uporabili le SGD ali SGD, nadgrajen z mehaniko *Nesterovovega momentuma*. Momentum je prilagoditev posodabljanja parametrov, ki pri posodobitvi upošteva še dodatno spremenljivko hitrosti, v kateri imamo shranjene pretekle posodobitve. Tako mehanika momentuma deluje kot zagon učenja in pomaga k hitrejši konvergenci, kadar je minimum funkcije na položnem delu grafa funkcije, kjer gradienta skoraj ni več, poleg tega pa tudi preprečuje, da bi se učenje ustavilo v lokalnem minimumu cenilne funkcije. Nesterovov momentum je rahlo spremenjena različica momentuma, ki malenkost izboljša njegovo delovanje in učinek.

3.3.2 Konvolucijska nevronska mreža

Model konvolucijske nevronske mreže (v nadaljevanju CNN) ima osnovo v študijah vidnega korteksa sesalcev, v katerem so zaznavne celice urejene tako, da so odzivne na majhna podobmočja vidnega polja, ki so razporejena čez

[2]Z besedo *hiperparameter* označujemo parametre, ki se tičejo učnega algoritma in jih ne smemo mešati z utežmi in pragovi, ki so parametri učnega modela in predstavljajo naučeno znanje.

celotno vidno polje. Te celice delujejo kot lokalni filtri in so dobro prilagojene za zaznavo lokalno soodvisnih elementov v naravnih slikah. CNN lokalno soodvisnost odkriva preko omejenih povezav med nevroni; vsak nevron ima na vhod povezano le majhno podmnožico nevronov prejšnjega nivoja ali vhoda. Na sliki 3.3 lahko vidimo preprost primer, v katerem ima vsak nevron nivoja m na vhodu le tri nevrone prejšnjega nivoja in je tako odziven le na podobmočje celotnega vhoda. Če na isti princip povežemo več nivojev zaporedno (zgradimo globoko mrežo), postanejo ta podobmočja vedno večja in manj lokalna, kot lahko vidimo na sliki 3.3. Nivo $m + 1$ ima še vedno samo tri vhode, a vseeno zajema celotni nivo $m - 1$. V posameznem nivoju je princip lokalne soodvisnosti implementiran s konvolucijo in CNN se nauči le parametre konvolucijskih jeder ter tako drastično zmanjša število parametrov. Vsak konvolucijski nivo ima lahko več jeder in tako proizvede več različnih filtriranih izhodov, tako imenovanih *slik značilk* (angleško *feature maps*), ki zajemajo različne značilnosti slike.



Slika 3.3: Primer lokalne odzivnosti preko omejenih povezav med nevroni v konvolucijski nevronski mreži.

Tipična arhitektura CNN vključuje enega ali več konvolucijskih nivojev za izluščenje značilnosti, le-tem lahko sledijo še nivoji podvzorčenja ter skritega in izhodnega nivoja kakor pri MLPju. Učenje poteka nadzorovano preko spusta po gradientu, posodabljajo pa se parametri vseh nivojev (razen nivojev podvzorčenja, če so uporabljeni).

Kljub dejstvu, da so bile konvolucijske mreže razvite na osnovi vidnega sistema, se uporabljajo tudi na področju zvočne analize [9, 10, 12, 15]. Fre-

kvenčne spektre večih zaporednih časovnih okvirjev lahko zložimo enega za drugim in tako dobimo dvodimenzionalno sliko frekvenčnega spektra po času, nad katero lahko nato izvajamo operacijo konvolucije.

3.3.3 Omejeni Bolzmannov stroj in globoke verjetnostne mreže

Osnovni sestavni del globoke verjetnostne mreže (v nadaljevanju DBN) je omejeni Boltzmannov stroj (angleško *restricted Boltzmann machine* ali RBM). RBM je dvosmerno povezana nevronska mreža, sestavljena iz vidnega in skritega nivoja, in je različica Boltzmannovega stroja, pri kateri ni medsebojnih povezav med nevroni v nivoju[3]. Učenje poteka nenadzorovano preko iskanja verjetnostne porazdelitve, ki najbolje opiše vhodne podatke. Vsaki rekonstrukciji vhoda določimo energijo, izračunano po neki energijski funkciji, in nato poskusimo to energijo čim bolj zmanjšati oz. poiskati minimum te energijske funkcije.

DBN dobimo, če več RBM-jev zložimo enega na drugega tako, da kot vidni nivo naslednjega RBM-ja vzamemo skriti nivo prejšnjega. DBN najprej nenadzorovano predučimo nekaj časa, da zajame značilnosti podatkov, nato pa imamo dve možnosti: lahko uporabimo parametre DBN-ja za inicializacijo MLP-ja in nato nadzorovano učimo ta MLP, ki ima enako strukturo kot DBN in ima tako v začetnih parametrih že neko predstavo o podatkih o katerih se uči, lahko pa dobljene značilke na izhodu zadnjega RBMja uporabimo kot vhod v nadzorovano učenje kakega drugega modela strojnega učenja. V ta namen lahko uporabimo MLP in tako dobimo podobno arhitekturo kot pri CNN, kjer značilke dobimo iz konvolucijskih nivojev.

[3]Od tu pride omejenost v imenu omejeni Boltzmannov stroj.

Poglavje 4

Poskusi

Izvedli smo štiri različne poskuse, v katerih smo primerjali različne modele in načine učenja. Vsak poskus smo izvedli dvakrat, vsakič z drugo transformacijo nad podatki in tako primerjali še 2 načina predpriprave podatkov.

Modele smo učili po principu *eden proti vsem* (angleško *One-vs-All* ali OvA), kar pomeni, da smo za vsak ton naučili svoj klasifikator, ki je razlikoval samo med tem, ali je določen ton prisoten v okvirju ali ne. Pri tem principu učenja je potrebno model učiti na uravnoteženih podatkih, torej mora imeti približno polovico pozitivnih in polovico negativnih primerov, da lahko pravilno razlikuje med njimi. Za vsak ton smo zato ustvarili podatkovno množico s 40 % učnih primerov s tem tonom in 60 % učnih primerov z naključnimi ostalimi toni. Na koncu bi tako morali naučiti 88 mrež, eno za vsak ton na klavirju, a smo obseg učenja omejili na interval štirih oktav od tona C2 do tona C6 oz. v MIDI notaciji od tona 36 do 84. Učenje smo omejili, ker je časovno zelo zahtevno, poleg tega pa je večina klavirske glasbe osredotočena na srednjih oktavah.

Nevronske mreže je težko učiti in potrebno je najti ravno prave hiperparametre, kot sta stopnja učenja in momentni člen, da dosežemo najboljše rezultate. Najboljše hiperparametre smo iskali s predhodnimi poskusi na le eni nevronski mreži, preden smo zagnali učenje vseh.

4.1 Poskus z MLP

Pri prvem poskusu smo za učenje uporabili večnivojski perceptron z enim skritim slojem s 75 nevroni. Učenje je potekalo preko aloritma SGD, pri katerem smo uporabljali po 20 učnih primerov v miniseriji za vsako posodobitev. Pri predhodnih poskusih smo ugotovili, da stopnja učenja 10^{-4} pripelje do najboljše konvergence, momentum pa v teh poskusih ni vplival na rezultate, zato ga nismo uporabili.

Velikost vhoda se je razlikovala glede na uporabljeno transformacijo: 420 pri FFT in 345 pri CQT. Pri transformaciji po metodi FFT smo uporabili okna dolžine 2048 vzorcev in od pridobljenih 1025 koeficientov zavrgli vse nad določeno frekvenco, nad katero klavirski toni in njihovi harmoniki gotovo ne sežejo. Za to frekvenco smo najprej določili 9000 Hz, kar nam je dalo 420 koeficientov. Kasneje smo ugotovili, da ni razlike med rezultati pri omejitvi 8000 Hz, in zato smo za nadaljnje poskuse zmanjšali število koeficientov na 375. 2048 vzorcev pri hitrosti vzorčenja 44,1 kHz pomeni, da so naša okna dolga 46,44 ms. Skok med posameznimi okni smo določili na 70 % dolžine okna, 32,52 ms. Pri CQT smo od začetka uporabljali razpon do 8000 Hz, spodnjo frekvenco pa nastavili na 55 Hz. Na vsako oktavo smo izračunali 48 koeficientov in tako dobili vektor 345 koeficientov. Skok med okni je znašal 25 ms. Ker v učni množici ne sme biti oken, v katerih ni nič drugega kot šum, saj nimajo nobene informacije o tonih, smo ta okna odstranili tako, da smo eskperimentalno določili prag tišine, ki je izločil okna s prenizko magnitudo signala. Okna so bila pred vhodom v model še skalirana na interval [0, 1].

Za določitev trajanja učenja smo uporabili metodo zgodnjega ustavljanja. Namesto da bi učenje pustili izvajati določeno število ponovitev, ga ustavimo, ko opazimo, da ne napreduje več. Pri vsakem prehodu smo desetkrat izmerili srednjo absolutno napako na testni množici in ko se štiri prehode zapored napaka ni zmanjšala, smo učenje ustavili. Prvih dvajset prehodov smo izvedli ne glede na napako, da se učenje ne bi zaključilo prezgodaj. Trajanje učenja se je razlikovalo za vsak ton, a se je povprečno zaključilo v 30-40 prehodih.

4.2 Poskus s CNN

Za drugi poskus smo uporabili konvolucijsko nevronsko mrežo z dvema konvolucijskima nivojema in enim navadnim skritim. Velikost miniserije smo obdržali enako kot pri MLP, 20 učnih primerov, stopnja učenja pa je bila lahko dosti višja, $2 * 10^{-3}$. Tudi do konvergence je prišlo dosti prej kot pri MLP in zato smo za osnovo učenja določili 12 prehodov, nato pa po štirih prehodih brez napredka učenje ustavili. Trajanje učenja je bilo povprečno med 20 in 30 prehodov.

Vhodne podatke smo pripravili tako, da smo 5 zaporednih okvirjev zložili enega poleg drugega in tako dobili dvodimenzionalno sliko. Pri poskusu s transformacijo CQT smo uporabili enake podatke in tako dobili sliko frekvenčnega spektra dolžine 125 milisekund ($5 * 25$ms) in dimenzij 5 x 345. Pri poskusu s transformacijo FFT pa smo malo zmanjšali skok med okni na 50 % dolžine okna, da ne bi dobili predolgih frekvenčnih spektrov, in tako dobili frekvenčni spekter preko 139,32 milisekund ($4 * \frac{46{,}44 \text{ ms}}{2} + 46, 44$ ms) dimenzij 5 x 375. Vsako sliko smo tudi tokrat skalirali na interval [0, 1].

V konvolucijskih nivojih smo uporabljali jedra velikosti 3 x 9, v prvem nivoju 10 jeder in v drugem 20. Tako smo na izhodu drugega sloja dobili 20 vektorjev[1] dolžine 329 v primeru CQT podatkov oz. dolžine 359 pri FFT podatkih. Po konvolucijskih je prišel navaden skriti nivo MLPja velikosti 80 nevronov in nato izhod. Nivojev podvzorčenja nismo uporabili.

4.3 Poskusi z DBN

Pri učenju z modelom DBN smo preizkusili 2 različna načina učenja: učenje na značilkah, pridobljenih iz prednaučenega modela, in fino učenje prednaučenega modela. Razlika je v tem, da pri prvem za vhod v nadzorovano učenje uporabimo rezultat prednaučenega modela, pri drugem pa za nadzo-

[1]Konvolucija zmanjša vsako dimenzijo za $d_j - 1$, kjer je d_j velikost jedra v tej dimenziji. Tako smo prvo dimenzijo v prvem sloju zmanjšali na $5 - (3 - 1) = 3$ in v drugem na $3 - (3 - 1) = 1$ ter tako iz slike dobili vektor.

rovano učenje uporabimo uteži in prage tega modela. Predučenje smo za vse končne poskuse izvedli samo enkrat, seveda pa smo tudi pri tem modelu iskali prave hiperparametre s predhodnimi poskusi predučenja.

4.3.1 Predučenje

Ker globoke nevronske mreže omogočajo vedno bolj abstraktno razumevanje podatkov v globljih nivojih, smo predučenje posameznih nivojev poskusili z vedno bolj abstraktnimi podatki. Uporabili smo 3 nivoje in prvega predučili na posameznih tonih, drugega na akordih in tretjega na glasbenih delih. Prvi nivo je obsegal 200 nevronov, drugi 160 in tretji 120.

Podatke smo pripravili skoraj popolnoma enako kot pri MLP, le da smo pri transformaciji FFT uporabili 375 koeficientov in da smo vhodne vektorje standardizirali namesto skalirali. Razlog za to leži v zgradbi in delovanju RBM-ja, ki v osnovni izvedbi deluje le na binarnih podatkih, za zvezne (kakršni so zvočni podatki) pa potrebujemo tako imenovano Gaussovo različico RBM-ja, ki za optimalno delovanje potrebuje standardizirane podatke.

Pri predučenju smo določili svojo stopnjo učenja za vsak nivo, na prvem je bila $5 * 10^{-4}$, na drugem in tretjem pa $2 * 10^{-3}$. V predhodnih poskusih se je izkazalo, da pri predučenju momentni člen pripelje do hitrejše konvergence, zato smo ga nastavili na 0.9. Izvedli smo le eno predučenje na vseh učnih primerih, ki so približno enakomerno zajemali vse tone, saj smo hoteli zajeti splošno predstavitev glasbene informacije. Zaradi tega je bila učna množica veliko večja kot pri učenju mrež za zaznavanje posameznih tonov in posledično je predučenje trajalo dlje časa. Za pospešitev smo zato povečali velikost miniserije na 100 učnih vzorcev. Za vzorčenje verjetnostne porazdelitve smo uporabili algoritem *obstojne kontrastivne divergence* [14] (angleško *persistent contrastive divergence*) z desetimi ponovitvami za vsak vzorec. Trajanje predučenja smo nastavili na 60 prehodov čez vse učne primere.

4.3.2 Nadzorovano učenje na značilkah

Nadzorovano učenje na značilkah iz prednaučenega modela je bilo delno uspešno. Značilke na izhodu drugega in tretjega nivoja niso nosile zadosti informacij, da bi se mreža naučila kaj dosti, saj smo v predhodnih poskusih dobili veliko slabše rezultate na testni množici kot pri osnovnem modelu večnivojskega perceptrona. Značilke na izhodu prvega nivoja pa so v predhodnih poskusih rezultate MLP-ja izboljšale za približno odstotek, zaradi česar smo se odločili, da bomo za izvedbo celotnega poskusa uporabili le prvi nivo globoke verjetnostne mreže. Za nadzorovano učenje smo tako zgradili MLP, ki je imel na vhodu 200 značilk posameznih tonov, predračunanih na prvem nivoju DBN-ja, in en skriti sloj s 50 nevroni.

Pri izvedbi končnega poskusa smo uporabili stopnjo učenja $2 * 10^{-3}$, momentni člen pa spet ni igral nobene opazne vloge, zato ga zopet nismo uporabili. Učenje smo tudi tokrat izvajali v miniserijah velikosti 20 učnih vzorcev. Učenje je bilo zaradi manjšega modela hitrejše, zaradi česar smo dovolili več prehodov čez učne podatke. Za osnovo smo nastavili 30 prehodov, nato pa učenje prekinili po 6 prehodih brez napredka. Učenje je povprečno trajalo 40-50 prehodov.

4.3.3 Fino učenje

Pri poskusu finega učenja prednaučenega modela uspeha nismo dosegli. Pri predhodnih poskusih je bila napaka na testni množici sicer primerljiva in celo malo boljša kot pri poskusu z MLP, vendar pa je model odpovedal pri testiranju na realnih glasbenih podatkih, kar očitno kaže na težavo prevelikega prileganja. Razlog za to najverjetneje leži v prevelikem številu parametrov tega modela, saj je imel 3 skrite nivoje, ki so bili vsi večji kot pri poskusu z modelom MLP. Poskusili smo tudi z manjšimi skritimi nivoji, a potem predučenje ni bilo uspešno, saj so bili nivoji premajhni za zajem prave predstavitve zvočnih podatkov. Tega poskusa zaradi predhodnih neuspehov nismo izvedli v celoti na vseh mrežah za vsak ton.

Poglavje 5

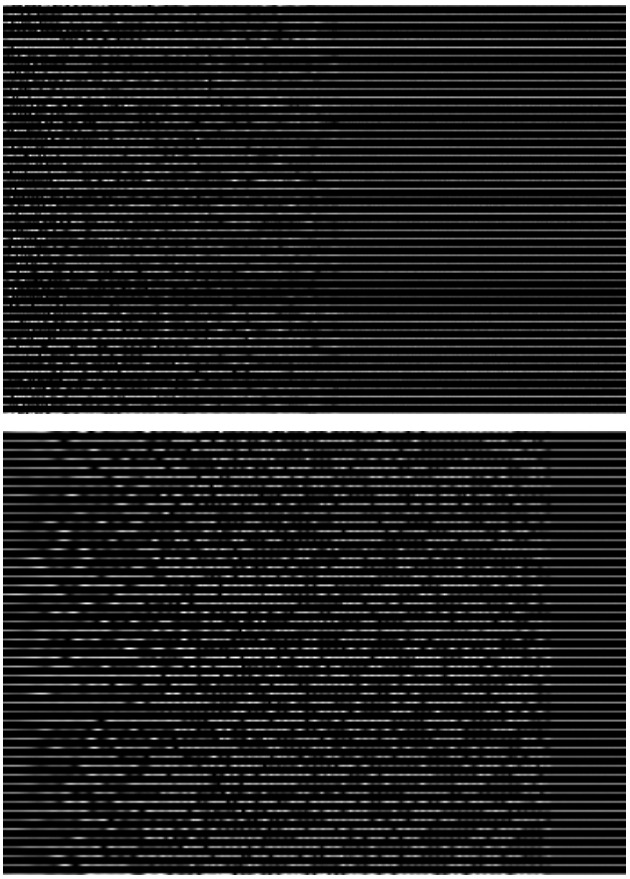
Rezultati in zaključki

Med učenjem vsakega modela smo sproti zapisovali njegovo uspešnost na testni množici, na koncu pa smo še preizkusili model na glasbenih delih, ki niso bila prisotna niti v učni niti v testni množici, da smo dobili oceno uspešnosti pri reševanju realnega problema. Ta rezultat je bolj relevanten od rezultatov na testni množici, saj je bila le-ta naključno izbrana izmed vseh podatkov in je tako vsebovala tudi posamezne tone, katerih transkripcija je veliko preprostejša od transkripcije glasbenih del. Kljub temu pa je mogoče nekaj zaključkov razbrati tudi iz uspešnosti na testni množici.

5.1 Rezultati predučenja pri modelu DBN

V končnih rezultatih smo zajeli le rezultate uspešnih poskusov pri modelu globoke verjetnostne mreže, torej rezultate učenja na značilkah iz prvega nivoja modela DBN. Na katere značilnosti vhodnih podatkov je bil ta nivo posebej pozoren si lahko ogledamo na sliki 5.1, kjer so izrisane uteži prvih petdesetih nevronov prvega nivoja v obliki vodoravnih vektorjev enake dolžine kot vhod z vrednostmi med ena in nič. Vsak nevron deluje kot filter, ki iz vhoda filtrira le značilnosti, za katere je v procesu predučenja ugotovil, da so bolj pomembne. Te preference izraža z močnejšimi povezavami na tem delu vhodnega vektorja. Na sliki 5.1 vidimo, da je bilo predučenje uspešno,

saj je model razbral, da so pri transformaciji FFT pomembnejše informacije v spodnjem (levem) delu frekvenčnega spektra, pri CQT pa čez cel spekter, razen v najvišjih frekvencah (najbolj desno), kjer so prisotni le še redki najvišji harmoniki.



Slika 5.1: Izris prednaučenih uteži prvih 50 nevronov prvega nivoja globoke verjetnostne mreže. Svetlejša slikovna točka pomeni, da je ta povezava močnejša, temnejša pa, da nevronu ta del vhoda ni toliko pomemben. Zgoraj: uteži prednaučenega modela pri poskusu s FFT. Spodaj: uteži prednaučenega modela pri poskusu s CQT.

5.2 Uspešnost učenja glede na testno množico

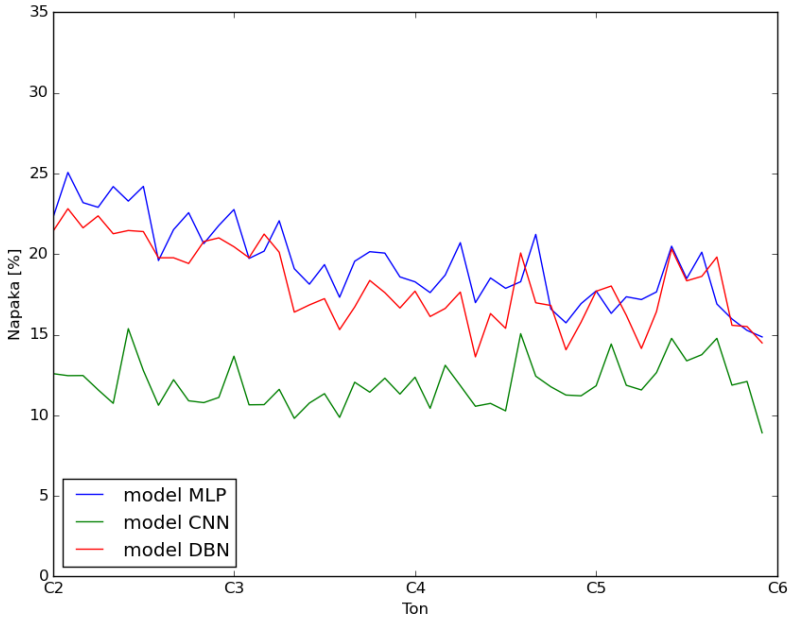
Med učenjem smo desetkrat na vsak prehod čez vse učne primere izračunali srednjo absolutno napako modela in vsakič, ko se je napaka izboljšala, še statistične mere preciznost, priklic in točnost. Po končanem učenju posameznega modela, ki je zajemalo učenje 48 mrež za tone preko srednjih štirih oktav klavirja, smo za vsak model izračunali povprečje statističnih mer vseh mrež ter končno mero F glede na povprečno preciznost in priklic. Končni rezultati so strnjeni v tabeli 5.1 in na podlagi le-teh se je za najboljši model izkazala konvolucijska nevronska mreža. Opazimo lahko tudi, da je bilo učenje na značilkah pri poskusu z modelom DBN malenkost uspešnejše od učenja na surovem vhodu pri poskusu z MLP, kar kaže na dejstvo, da je mogoče s pravim predučenjem iz glasbenih podatkov izluščiti značilke, na katerih je nato mogoče preprosteje in natančneje učiti. Primerjavo med modeli si lahko ogledamo tudi na sliki 5.2.

	napaka		preciznost		priklic		točnost		mera F	
model	FFT	CQT	FFT	CQT	FFT	CQT	FFT	CQT	FFT	CQT
MLP	19,46%	21,52%	0,79	0,77	0,70	0,65	0,81	0,78	0,74	0,71
CNN	11,92%	13,21%	0,89	0,87	0,80	0,78	0,88	0,87	0,84	0,83
DBN	18,17%	19,85%	0,81	0,79	0,71	0,68	0,81	0,80	0,76	0,73

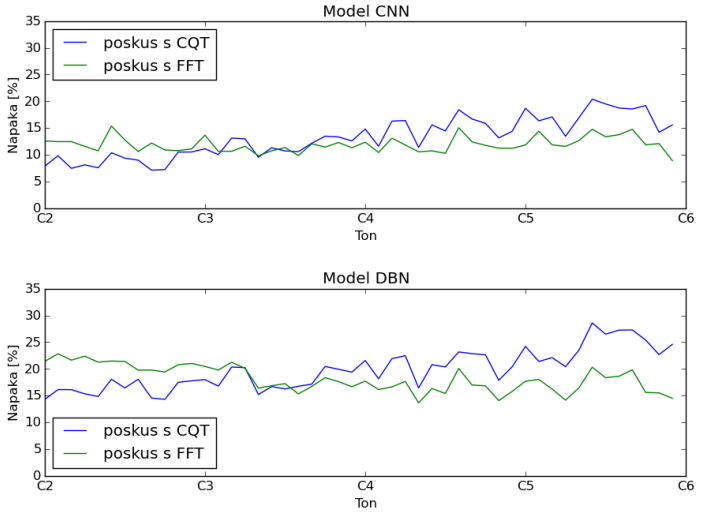
Tabela 5.1: Izpis povprečja srednje absolutne napake in statističnih mer za vsak model.

Rezultati na testni množici so nam dali zelo dobro primerjavo med transformacijama FFT in CQT. Za boljšo se je izkazala FFT, kot lahko razberemo iz tabele 5.1, kjer so rezultati poskusov s FFT v vseh merah boljši od rezultatov poskusov s CQT. Razlog za to je najverjetneje v tem, da FFT bolje ohrani informacijo o harmonikih.

Na sliki 5.3 si lahko ogledamo neposredno primerjavo transformacij čez vse tone. Pri poskusu s transformacijo CQT je model bolje zaznaval tone v nižjih oktavah, pri FFT pa v višjih. To je bilo za pričakovati, saj so pri



Slika 5.2: Primerjava srednje absolutne napake vseh treh modelov na vseh naučenih tonih pri poskusu s transformacijo FFT.



Slika 5.3: Primerjava rezultatov transformacij pri modelih CNN in DBN.

CQT spodnji toni bolje ločeni kot pri FFT, kjer je lahko več nižjih tonov predstavljenih z istim koeficientom. Za prelomno se je izkazala oktava od tona C3 do C4 (mala oktava), kjer so bili rezultati približno primerljivi.

5.3 Rezultati na glasbenih delih

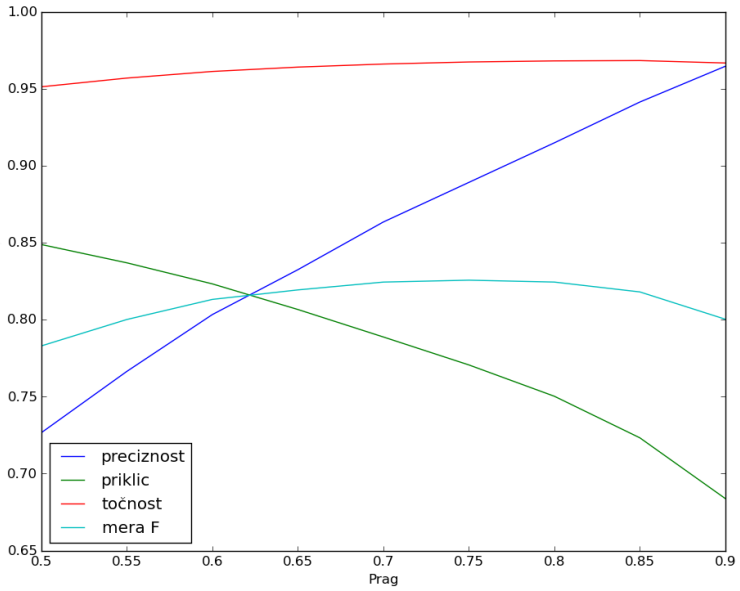
Iz učne in testne množice smo izvzeli 4 posnetke glasbenih del različnih kompleksnosti in na teh posnetkih ocenili končno uspešnost transkripcije. Ta dela smo spustili skozi vse naučene nevronske mreže vsakega modela in pri vsakem izračunali preciznost, priklic, točnost in mero F kot povprečje teh mer za vsak ton. V tabeli so našteta uporabljena dela, najvišja polifonija ter šifre, s katerimi se bomo v nadaljnjih tabelah nanašali na ta dela.

delo	šifra	stopnja polifonije
E. Grieg, Waechterlied	grieg-waechter	9
J. S. Bach, BWV 850	bach-850	10
A. Borodin, Petite Suite 6	borodin-ps6	10
R. Schumann, Op. 15, 2. st.	schumann15-2	11

Tabela 5.2: Tabela uporabljenih del. Stopnja polifonije je bila izračunana glede na najvišje število sočasnih tonov v oknih dolžine 46,44 ms (2048 vzorcev).

Ker smo z aktivacijsko funkcijo v izhodnem sloju izhod omejili na interval [0, 1], si lahko ta izhod razlagamo kot verjetnost, ali je ton prisoten ali ne. Tako imamo potem več možnosti nastavitve praga verjetnosti, nad katerim je ton prisoten. Poskusili smo s pragi od 0,5 do 0,9 in ugotovili, da je pri večini modelov mera F najvišja pri pragih med 0,65 in 0,75, zato smo za končno oceno uspešnost določili prag 0,7. Na sliki 5.4 si lahko ogledamo tipično spreminjanje preciznosti, priklica, točnosti ter mere F čez celotni interval testiranih pragov. Vidimo, da z višjim pragom narašča preciznost, priklic pa pada. Razlog za to je v tem, kar ti dve meri predstavljata: preciznost je

odstotek pravilno klasificiranih pozitivnih primerov glede na vse primere, ki so klasificirani kot pozitivni, priklic pa odstotek pravilno klasificiranih pozitivnih primerov glede na vse dejanske pozitivne primere. Z zviševanjem praga odstranjujemo pozitivno klasificirane primere o katerih model ni najbolj prepričan, in s tem odstranjujemo tako napačno kot pravilno klasificirane pozitivne primere. Posledica je, da je vedno manj klasificiranih pozitivnih primerov, ne glede na to, ali so pravilni ali ne in zato priklic pada. Po drugi strani pa nam ostajajo le še pozitivni primeri, o katerih je model bolj prepričan, in tako je vedno manj napačnih pozitivnih primerov, zato preciznost narašča. Mera F nam pomaga izbrati pravo razmerje med preciznostjo in priklicom. Na sliki 5.4 lahko vidimo tudi, da je klasifikacijska točnost dosti višja od ostalih mer, kar je posledica tega, da ostale mere ne upoštevajo pravilno klasificiranih negativnih primerov.



Slika 5.4: Spreminjanje statističnih mer uspešnosti modela glede na izbrani prag. Na sliki smo uporabili model konvolucijske nevronske mreže in transformacijo FFT na glasbenem delu grieg-waechter.

	preciznost		priklic		točnost		mera F	
delo	FFT	CQT	FFT	CQT	FFT	CQT	FFT	CQT
grieg-waechter	0,71	0,61	0,62	0,51	0,94	0,94	0,66	0,56
bach-850	0,43	0,36	0,53	0,38	0,94	0,93	0,48	0,37
borodin-ps6	0,67	0,64	0,52	0,45	0,92	0,92	0,59	0,53
schumann15-2	0,57	0,48	0,57	0,50	0,92	0,92	0,57	0,49

Tabela 5.3: Rezultati modela MLP na realnih glasbenih podatkih

	preciznost		priklic		točnost		mera F	
delo	FFT	CQT	FFT	CQT	FFT	CQT	FFT	CQT
grieg-waechter	0,86	0,79	0,79	0,66	0,97	0,95	0,82	0,72
bach-850	0,62	0,64	0,74	0,73	0,95	0,95	0,68	0,68
borodin-ps6	0,84	0,74	0,68	0,63	0,94	0,95	0,75	0,68
schumann15-2	0,80	0,75	0,64	0,71	0,95	0,95	0,71	0,73

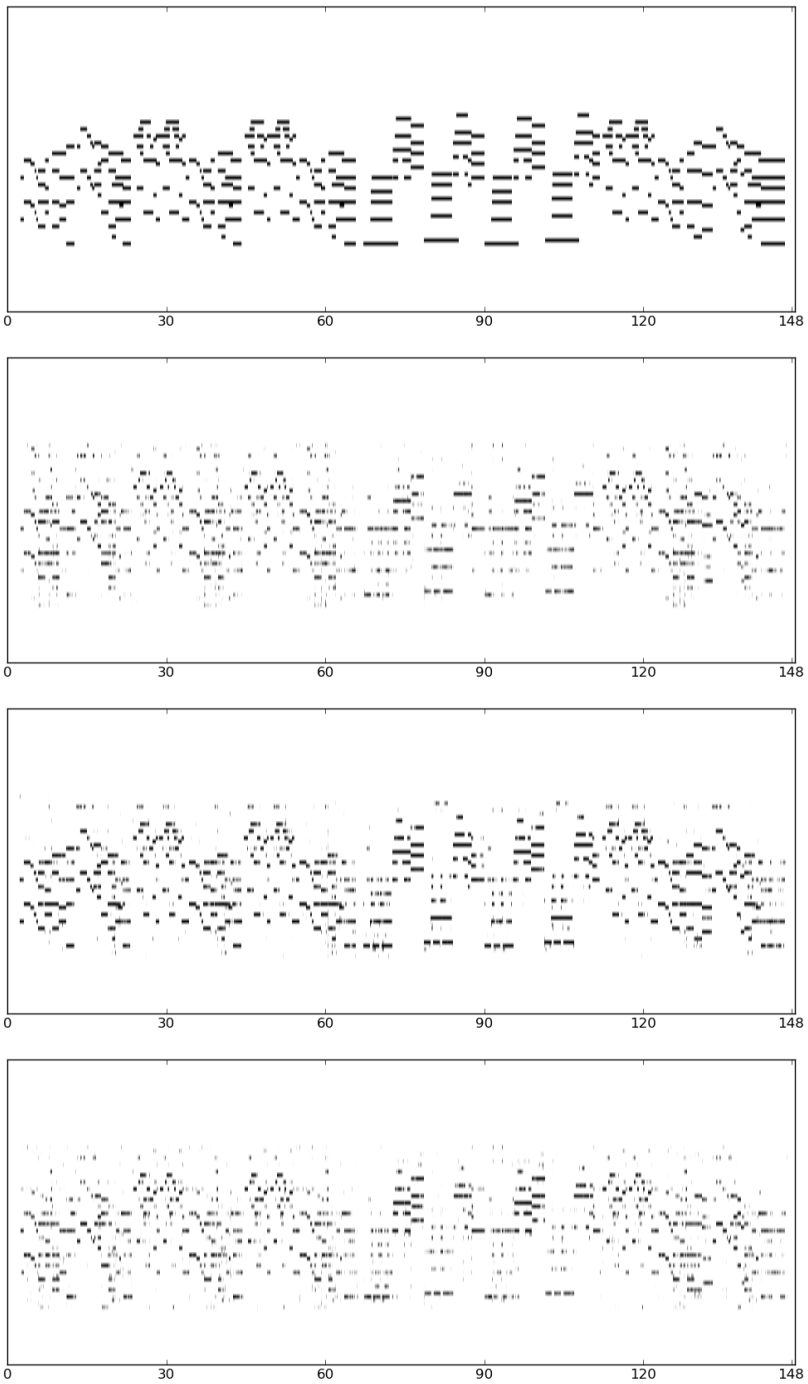
Tabela 5.4: Rezultati modela CNN na realnih glasbenih podatkih

	preciznost		priklic		točnost		mera F	
delo	FFT	CQT	FFT	CQT	FFT	CQT	FFT	CQT
grieg-waechter	0,75	0,66	0,67	0,55	0,95	0,94	0,70	0,60
bach-850	0,45	0,40	0,59	0,53	0,94	0,93	0,51	0,45
borodin-ps6	0,73	0,64	0,53	0,52	0,93	0,93	0,61	0,57
schumann15-2	0,59	0,52	0,59	0,59	0,93	0,93	0,59	0,55

Tabela 5.5: Rezultati modela DBN na realnih glasbenih podatkih

V tabelah 5.3, 5.4 in 5.5 imamo primerjavo vseh modelov pri pragu 0,7. Takoj opazimo, da so rezultati dokaj podobni rezultatom na testni množici, le nekoliko slabši, kar je posledica dejstva, da je transkripcija celotnih glasbenih del veliko težja kot transkripcija posameznih tonov, ki so bili med drugimi tudi prisotni v testni množici. Na podlagi teh rezultatov lahko zopet zaključimo, da je model konvolucijske nevronske mreže najuspešnejši, model globoke verjetnostne mreže pa ima tudi pri reševanju realnih problemov boljše rezultate od osnovnega večnivojskega perceptrona. Opazimo pa lahko tudi velika nihanja uspešnosti med posameznimi deli, kar pa je posledica dejstva, da so dela različno kompleksna. Bachov preludij in fuga št. 5 v D-duru (oznaka BWV 850) je namreč izredno hitra skladba s hitrimi menjavami tonov, Griegov Wächterlied pa je dosti počasnejša in ima tudi nižjo stopnjo polifonije, zaradi česar je uspešnost transkripcije dosti višja.

Na sliki 5.5 imamo vizualizirano skladbo Wächterlied, ter njene transkripcije s posameznimi modeli. Vidimo lahko, da je v transkripcijah prisotnega dosti šuma, ki pa bi ga lahko do neke mere odstranili z naknadno obdelavo. Na prvi pogled sta transkripciji z modelom MLP in DBN praktično enaki, a ob natančnejšem pogledu lahko vidimo, da je pri transkripciji z DBN prisotno malo manj šuma. Iz tega lahko sklepamo, da je predučenje modela dobro zajelo pravo predstavitev zvočne informacije.



Slika 5.5: Od zgoraj navzdol si sledijo: vizualizacija skladbe Wächterlied, transkripcija te skladbe z modelom MLP, transkripcija z modelom CNN in transkripcija z modelom DBN. Pri vseh treh transkripcijah je bila uporabljena transformacija FFT.

5.4 Zaključki

V tem diplomskem delu smo pokazali, da je transkripcija klavirske glasbe z globokimi nevronskimi mrežami mogoča. Poleg tega smo pokazali tudi, da je mogoče z nenadzorovanim predučenjem vplivati na uspešnost učenja. Po pričakovanjih smo z globljimi modeli konvolucijske nevronske mreže in globoke verjetnostne mreže nadgradili rezultate osnovnega modela večnivojskega perceptrona. Pri primerjavi predpriprave podatkov s transformacijama hitre Fourierove transformacije in transformacije s konstantnim Q smo ugotovili, da se na splošno bolje obnese prva, a ima druga boljše rezultate pri nižjih tonih.

Naše delo je pokazalo, da je globoko učenje močno orodje pri zvočni analizi oz. natančneje pri transkripciji glasbe. Naši poskusi z globokimi verjetnostnimi mrežami niso bili popolnoma uspešni, a nam je delni uspeh pokazal, da je mogoče z nenadzorovanim učenjem iz zvočnih podatkov izvleči njihove značilnosti ter jih uporabiti za lažje in hitrejše nadzorovano učenje. Z nadaljnjimi poskusi predučenja bi bilo verjetno mogoče izvleči tudi bolj abstraktne zakonitosti zvoka in jih uporabiti za natančnejše prepoznavanje tonov. Ugotovili smo tudi, da je konvolucijska nevronska mreža odličen model za klasifikacijo tonov, kljub dejstvu, da je zasnovana na biološkem vidnem sistemu. Iz tega lahko morda sklepamo tudi na dejstvo, da sta si biološki vidni in slušni sistem podobna.

Področje globokih nevronskih mrež je relativno novo in raziskovanje globokega učenja je v polnem zagonu. Naši rezultati so bili spodbudni za nadaljnje raziskovanje na področju transkripcije glasbe, predvsem pri raziskovanju nenadzorovanega predučenja. Glede na to, da nam je uspelo s predučenjem izboljšati rezultate nadzorovanega učenja večnivojskega perceptrona, lahko sklepamo, da je s predučenjem na konvolucijskih omejenih Boltzmannovih strojih možno izboljšati tudi že tako dobre rezultate konvolucijske nevronske mreže.

Zaenkrat popoln tranksripcijski sistem še ne obstaja in ne vemo, če je sploh mogoč. Niti človeški možgani, ki so trenutno najmočnejši procesni

sistem, ki ga poznamo, niso sposobni iz zvoka razbrati vseh njegovih zakonitosti. Vseeno pa človeškega zanimanja to ne ustavi in raziskovanje tega področja se nadaljuje. Odkar se pri nalogah, ki jih človek brez težav opravlja vsak dan, kot recimo pridobivanje informacij iz zvoka, vedno bolj uporablja (globoke) nevronske mreže, pa je to raziskovanje dobilo še novo razsežnost: raziskovanje nas samih, saj nevronske mreže vsaj približno modelirajo procese v našem živčnem sistemu in preko njih lahko dobimo nova spoznanja tudi o nas.

Literatura

[1] S. Böck, M. Schedl, "Polyphonic piano note transcription with recurrent neural networks", v zborniku *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, 2012, str. 121-124.

[2] M. Davy, S. J. Godsill, "Bayesian Harmonic Models for Musical Signal Analysis", v zborniku *Bayesian Statistics 7: Proceedings of the Seventh Valencia International Meeting*, 2003, str. 105-124.

[3] D. Gerhard, "Audio Signal Classification: History and Current Techniques", Department of Computer Science, University of Regina, Regina, Kanada, Technical Report TR-CS 2003-07, nov. 2003.

[4] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, V. Shet, "Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks", *CoRR*, abs/1312.6082, jan. 2014.

[5] M. Goto, "A Predominant-F0 Estimation Method for Polyphonic Musical Audio Signals", v zborniku *18th International Congress on Acoustics*, apr. 2004, str. 1085-1088.

[6] S. Haykin, *Neural networks: a comprehensive foundation*, Prentice Hall, New Jersey, 1999.

[7] A. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method", v zborniku *2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, ZDA, okt. 2005, str. 291-294.

[8] I. Kononenko, *Strojno učenje*, Založba FE in FRI, Ljubljana, 2005.

[9] H. Lee, Y. Largman, P. Pham, A. Y. Ng, "Unsupervised Feature Learning for Audio Classification using Convolutional Deep Belief Networks", v zborniku *Advances in Neural Information Processing Systems 22*, Vancouver, dec. 2009, str. 1096-1104.

[10] T. L. H. Li, A. B. Chan, A. Chun, "Automatic Musical Pattern Feature Extraction Using Convolutional Neural Network", v zborniku *Proceedings of the International MultiConference of Engineers and Computer Scientists 2010*, Hong Kong, mar. 2010.

[11] M. Marolt, "A Connectionist Approach to Automatic Music Transcription of Polyphonic Piano Music", v *IEEE Transactions on Multimedia 6*, 2004, str. 439-449.

[12] J. Schlüter, S. Böck, "Improved Musical Onset Detection with Convolutional Neural Networks", v zborniku *Proceedings of the 2014 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*, maj 2014, str. 6979-6983.

[13] P. Smaragdis, J. C. Brown, "Non-Negative Matrix Factorization for Polyphonic Music Transcription", v zborniku *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, ZDA, okt. 2003, str. 177-180.

[14] T. Tieleman, "Training Restricted Boltzmann Machines using Approximations to the Likelihood Gradient", v zborniku *Proceedings of the 25th international conference on Machine learning*, New York, ZDA, jul. 2008, str. 1064-1071.

[15] K. Ullrich, J. Schlüter, T. Grill, "Boundary Detection in Music Structure Analysis using Convolutional Neural Networks", v zborniku *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014)*, Taipei, okt. 2014.

[16] (2013) The Oldest Song In The World [Online]. Dosegljivo na: http://www.amaranthpublishing.com/hurrian.htm. [Dostopano 9.8.2015].