Using Deep Learning to recommend Punjabi Music

by

Harpreet Singh
B.Tech, Punjab Technical University, 2012

A Project Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Computer Science

Using Deep Learning to recommend Punjabi Music

by

Harpreet Singh
B.Tech, Punjab Technical University, 2012

Supervisory Committee

Dr. George Tzanetakis, Supervisor
(Department of Computer Science)

Dr. Kwang Moo Yi, Departmental Member
(Department of Computer Science)

Supervisory Committee

Dr. George Tzanetakis, Supervisor
(Department of Computer Science)

Dr. Kwang Moo Yi, Departmental Member
(Department of Computer Science)

ABSTRACT

A recommender system allows users to discover and listen to songs similar to the song they have been listening to. Collaborative filtering has been the system of choice for most music streaming services, but this type of recommendations ignore actual musical content of the songs. For genres like Punjabi music, these systems suffer from lack of historic data and hence recommendation quality is unsatisfactory. In this project, we perform experiments to determine how Convolutional Neural Networks can be used to learn musical features from Indian and specifically Punjabi Music. We investigate using these features to perform content based recommendations. We use mel-spectrograms of the songs to train a CNN on classification tasks and then use the learned weights as a feature extractor for songs. We investigate the performance of CNNs with weights trained on Western Music and fine-tuned on Punjabi Music and perform a simple study to understand the quality of recommendations. These experiments demonstrate how features learned on western music can be transferred to learn features for non-western music.

Contents

List of Figures

ACKNOWLEDGEMENTS

I would like to thank:

my supervisor Dr. George Tzanetakis, for his continuous support, guidance, mentoring and for giving me this wonderful opportunity.

my parents Gurpreet Kaur and Inderjit Singh, for everything.

'Think deeply about things. Don't just go along because that's the way things are or that's what your friends say.'

- Aaron Swartz

DEDICATION

I dedicate this to Professor George Tzanetakis, my family and friends.

Chapter 1

Introduction

There is a big shift happening in the way media is consumed; from owning media to paying for a subscription and just streaming the media. This has been the case with music media, with streaming services like Spotify, Apple Music, etc becoming increasing popular and mainstream [1]. Streaming has made music more accessible and users are listening to more music [2]. YouTube streams are now used as a factor in calculating chart positions. The second most subscribed channel on YouTube is in fact an Indian music label [3]. Automatic music recommendation has therefore become a very relevant and well researched problem. The importance of focusing on recommendation systems that work well with Indian languages is also justified.

Recommender systems as a field have been studied in great depth [4], but the problem of music recommendation is unique in the sense that it is greatly affected by the vast variety of genres and styles. It is also greatly impacted by demographic, geographic and psychographic factors [5]. Among other factors are user taste; for e.g. users prefer certain songs more than others from even the same artist. This problem is exacerbated in a region like India with 22 official languages and thousands of years of music history. The Punjab region is divided between India and Pakistan and the Punjabi language has a total of 122 million Native Speakers [6] [7]. Punjabi music is very diverse, ranging from folk to Sufi to Classical. Contemporary Punjabi Music has several further sub-genres. Recommending Punjabi music therefore is not trivial.

1.1 Collaborative Filtering

Most music streaming services rely on user based collaborative filtering [8][9] where in the general idea is that people who agreed in the past are very likely to agree again. Similar users are identified and the opinion of similar users is used to predict opinion of a user regarding an item. Several algorithms exist under the umbrella of collaborative filtering but the general idea is based on user similarity.

Collaborative filtering has a number of problems. Firstly, it fails to work of items that are not popular because there isn't enough data to calculate opinion on items that few users have rated. Secondly, it exhibits the so called cold-start problem i.e. recommender systems can not predict ratings for a new item until similar users have rated it.

Due to these limitations with collaborative filtering, music services that don't have a critical mass of non-western users suffer from subpar recommendations for non-western music. Punjabi music is a vast set and cold start problem causes items that have never been consumed to not be recommended.

1.2 Content Based Recommendation

Content based recommendations work on music features that are extracted either automatically or by humans. Pandora is an example of a music service that uses humans to tag songs [10] and then uses this information to recommend music to users of their online radio service. Automatic content based systems [11] work on information extracted from music files automatically. This can be achieved by calculating the similarity between audio signals. These systems can be further improved by training these systems against user preferences.

1.3 Structure of the Report

This section provides information on what each Chapter of this Report will discuss:

Chapter 1 gives an introduction to the current state of recommendation systems in music and popular ways to handle music recommendation tasks

Chapter 2 describes the problem being worked on and related work.

Chapter 3 discusses the datasets and models used for the experiments.

Chapter 4 describes the tasks performed as part of experiments and their results.

Chapter 5 deals with evaluation of results and limitations. A conclusion for project is also stated.

Chapter 6 describes the scope of future work.

Chapter 2

Problem and Related Work

2.1 Punjabi Music

While there has been significant research on Hindustani (North Indian) [12] and South Indian Music [13], there isn't much research on music information retrieval for Punjabi Music. Punjabi Music although overlapping with North Indian music to an extent, is a whole music culture by itself, being that Punjabi is spoken by 122 million people. The focus of this project is more on contemporary/vernacular music [14]. Bollywood[15] also has appropriated Punjabi rhythms to a big extent, so any learnings and findings from this project would also be relevant to Bollywood music.

Punjab region although divided among two nations after the Independence of India and Pakistan is brought together by Punjabi music [16] [17]. Punjabi music represents feelings of a culture separated by politics. This is especially prominent in the Punjabi music scene among the diaspora. In UK for example the UK Bhangra/Asian genre includes artists from both Indian and Pakistani Punjab.

2.2 Related Work

While colloborative filtering has been shown to work really well in music recommendation tasks[9] , using similarity of audio signals can be used to recommend audio to a listener based on the song the listener chooses or has already liked or been listening to. Convolutional neural networks have been used to learn latent factors for audio and these latent factors have been used to find similarity of audio signals[18]. Transfer learning has been shown to work for both audio [19] and computer vision [20][21].

Chapter 3

Data

3.1 Datasets

DS1complete Consists of 450 Punjabi and Hindi songs belonging to diverse range of sub-genres.

DS1 consists of 250 songs with multiple tags available for each song. This dataset is a subset of dataset - DS1complete but only contains songs with acceptable quality of meta-tags. Tags for this dataset are derived from last.fm artist tags for the songs.

DS2 has 750 songs and each song in this dataset belongs to only one class.

We also use:
GTZAN Genre[22]: This dataset consists of 1000 audio tracks each 30 seconds long. It contains 10 genres, each represented by 100 tracks. The tracks are all 22050Hz Mono 16-bit audio files in .wav format.

Problems with datasets

There hasn't been much academic research in the field of Music Information Retrieval for Punjabi Music and for this reason there isn't a dataset available for vernacular Punjabi Music. Most research on music of the Indian Subcontinent deals with classical music genres like Hindustani music (North Indian) or Carnatic (South Indian). Although sub-genres of Punjabi music like Sufi and Patiala Gharana do fall under academically studied Hindustani music, it isn't the subject of focus. Contemporary

Punjabi Music is a severely understudied field and therefore doesn't have a public dataset.
So the decision was made to use my personal music library for the project. Due to music files not containing proper artist and song tags a lot of difficulty was faced to find appropriate meta tags for music files. After performing pre-processing we were left with a total of around 1000 songs with acceptable quality of meta tags.
Related research in audio analysis mostly use collaborative filtering data [23] for audio files as a parameter to train against. This was not possible with our dataset due to the unavailability of such metadata for Punjabi music.

3.2 Predicting music sub-genres from music audio

Predicting music genres from music audio is a classification problem. Mel-spectrogram [24] of audio is extracted using Kapre [25] and used to train Convolutional Neural Networks to predict sub genres.

3.3 Convolutional Neural Networks for Music Information Retrieval

A decision was made to use Convolutional Neural Networks for audio analysis. Just like fully connected neural networks, CNNs are made of neurons that can learn weights and biases [26].

CNNs in simplest terms are a collection of layers that convert an image into an output that contains class scores. Each convolutional layer consists of filters that can be learned. During the forward pass, each filter is slid across the width and height of the input volume. This results in creation of an activation map. These filter activations represent learning of certain visual features. So when the filters see this visual pattern again, they activate. [27][28] CNNs are the standard when it comes to computer vision tasks and outperform other techniques [29].

CNNs have been applied to audio analysis tasks with the assumption that the network would learn audio features from the time-frequency representation of audio. In computer vision tasks, filters in lower layers learn simpler features like edge detection

while later layers higher level features like human faces, etc. This learning translates well to field of audio analysis because lower level filters can be used to learn features like pitch and beats while filters in higher layers learn characteristics more relevant to certain genre [19].

3.4 Transfer Learning

The single biggest challenge in the project was data. There isn't any Punjabi music dataset available with an amount of data that can be used effectively with deep learning techniques, which rely on a notoriously high amount of data. With the 1000 song dataset that was prepared for this project, a deep convolutional neural network would simply overfit.

With transfer learning, we can use weights learned on a similar problem and adapt a network with these pretrained weights to a different but similar problem. Transfer learning is used for vision tasks[30][31] and has been shown to work with audio analysis as well.

In this project we use the model used here[19] which is trained on the Million Song Dataset [32] subset and adapt the model to our problem. We achieve this by removing the last fully connected layer and replace it with a new fully connected layer that has N outputs where is N is the number of classes in the experiment. A train-validation split of 80-20 is used.

Pre-trained weights were useful to our experiments because they already encode features of the Million Song Dataset in their weights. By fine-tuning the model on our Punjabi Dataset we learn higher level features that are more suited to Punjabi music sub-genres.

Cosine similarity is used to generate recommendations [18].

Each song in the dataset is represented as a vector. We use a time-frequency representation of the audio files to feed into the network. By using a CNN to predict music sub-genres, we use the weights learned by the network as latent factors for music audio. These vectors are then used for the recommendation process. By using a predicted vector for a given music audio, we then use cosine distance to find other songs in the dataset whose predicted vectors are close to our given song. The playlists are then generated and ordered by the cosine similarity to the seed song. When generating playlists, we always leave out the first song as it is always the seed song itself.

Chapter 4

Experiments and Results

4.1 Overview

This section provides an introduction to the experiments performed:

Experiment 0 Investigate if transfer learning model is viable.

Experiment 1 Investigate if features predicted by transfer learning work for Indian music.

Experiment 2 Investigate feature performance of Indian dataset mixed with GTZAN genre dataset.

Experiment 3 Predict GTZAN genres using features predicted by Transfer learning.

Experiment 4 Finetune model with DS1 and investigate performance.

Experiment 5 Finetune model with DS2 and investigate performance.

Experiment 6 Investigate in detail performance of models learned in experiments 4 and 5 .

Experiment 7 Investigate in detail what each layer of the model learns.

4.2 Challenges

We use Keras[33] which is a high level neural network API running on top of Tensorflow[34] to run and build models for this experiment.

CNN model used by transfer learning[19] is used. **See figure** 4.1

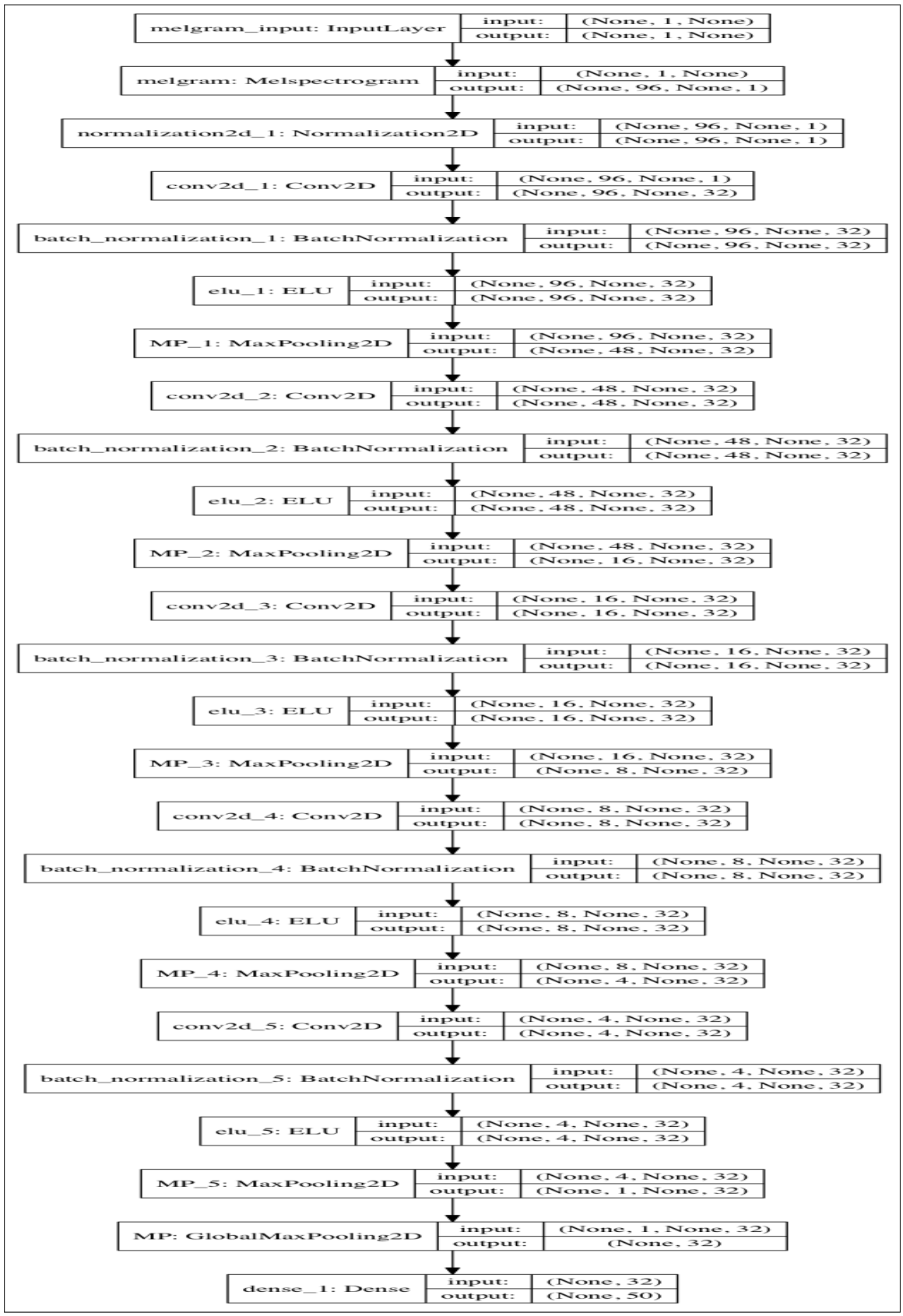

Figure 4.1: Base Model

We also used the weights learned by this model. For each experiment hyperparameter tuning was performed to achieve the best results. A really small rate of 0.001 was used for stochastic gradient descent during the fine-tuning process. This was done so that when new data is encountered during the fine-tuning process, there are are no drastic updates to the already learned weights thereby destroying them.

Key features about the architecture[19]:

1. Kapre is used to create a Keras layer that generates mel-spectrogram. Input has a single channel and 96-mel bins.
2. Sampling-Rate of 12000 Hz and 29 sec of audio is used.
3. Each Convolutional layer uses 32 filters and a kernel size of (3, 3) and stride of (1, 1)
4. Batch normalization is used.
5. Exponential Linear Unit is used as activation function in all Convolutional layers.
6. Max pooling of (2, 4), (3, 4), (2, 5), (2, 4), (4, 4) is applied after every Convolutional layer.

4.3 Experiment 0:

To see if transfer learning model indeed works for our problem this experiment was setup. In this experiment we used the GTZAN Genre dataset[22] and the features extracted using transfer learning were visualized using t-SNE[35] after performing PCA on the features. PCA[36] was performed because transfer learning outputs 160 features for each audio file and we need lower dimensionality to effectively use tSNE. We use n_components = 50.

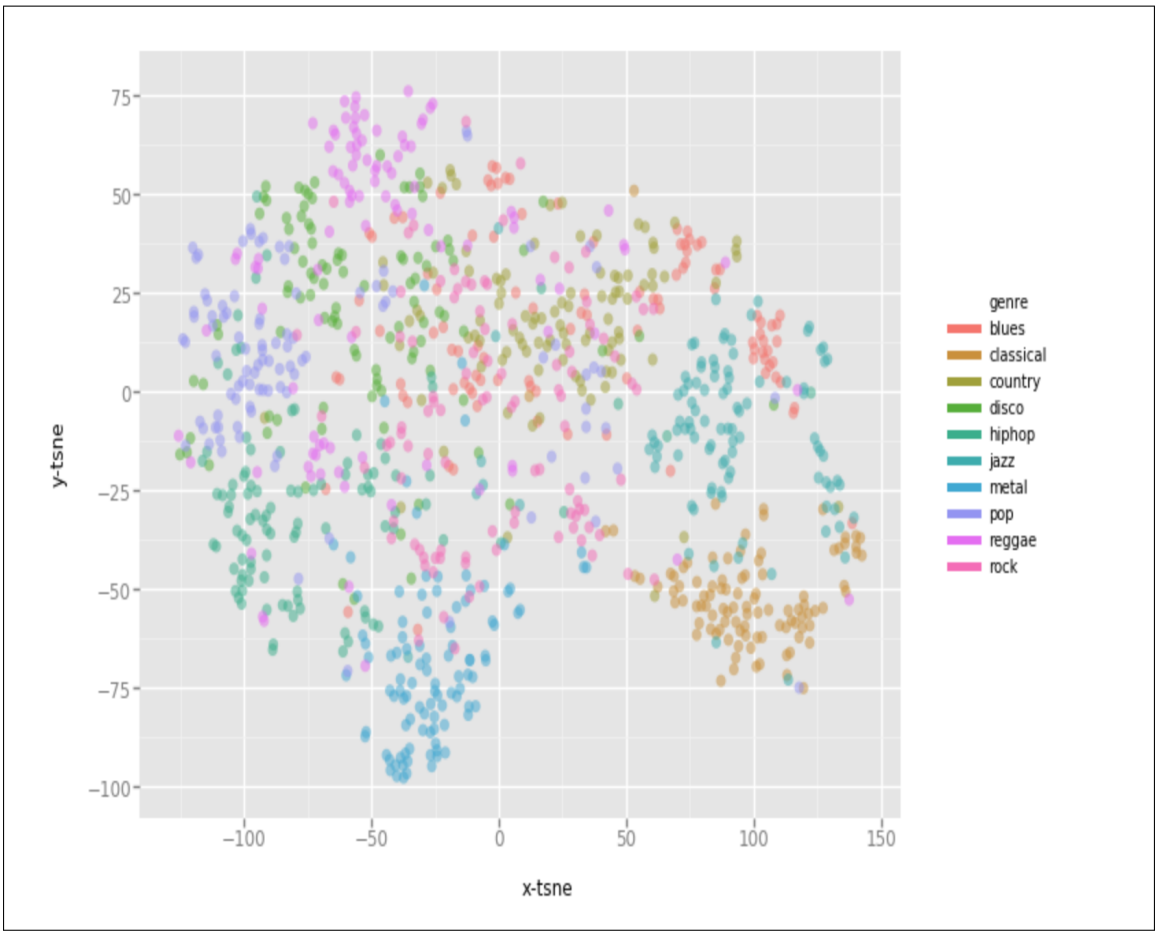

Figure 4.2: tSNE Visualization for features extracted Experiment 0

4.3.1 Result of experiment 0:

Features generated are useful and clustering works well. Therefore we can use similarity of audio features to perform music recommendations.

4.4 Experiment 1:

In this experiment we use a very small set of audio belonging to Punjabi, Hindi and Bengali genre for feature extraction using transfer learning. tSNE works to a degree for clustering with Punjabi and a subset of Hindi audio files being clustered together, and a subset of Hindi audio files being clustered with Bengali audio files. On playing the audio, the reasons became clearer. Songs that have similar audio features are

getting clustered together and in the absence of higher level features that could help in differentiating a slow Punjabi song from a slow Hindi song, the two audio files end up being clustered together. This experiment was just to get intuition into how the transfer learning features work with non-western music.

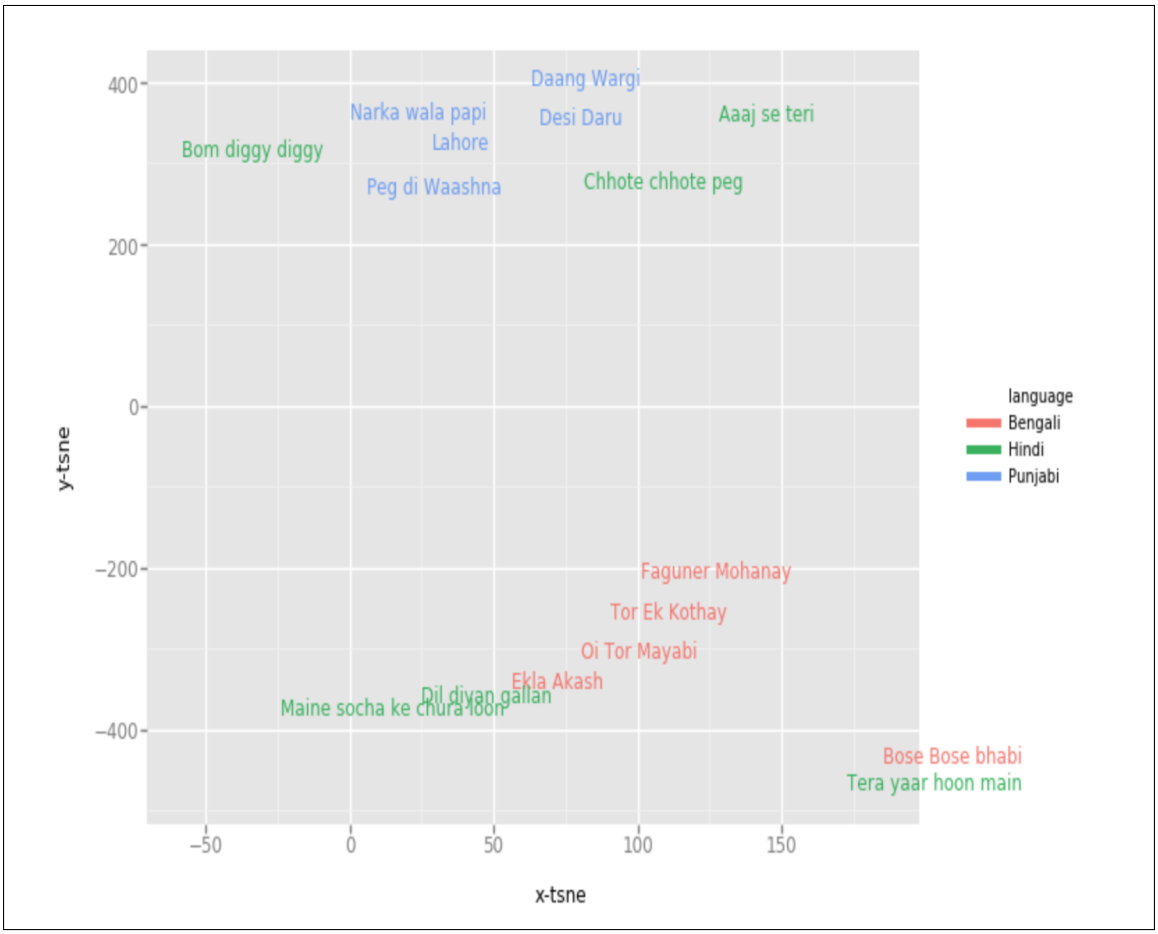

Figure 4.3: tSNE Visualization for features extracted Experiment 1

4.5 Experiment 2:

For experiment 2 we extract features for GTZAN Genre dataset mixed with 500 Punjabi songs. While other GTZAN genres cluster well, Punjabi songs cluster poorly and are mixed with other genres.

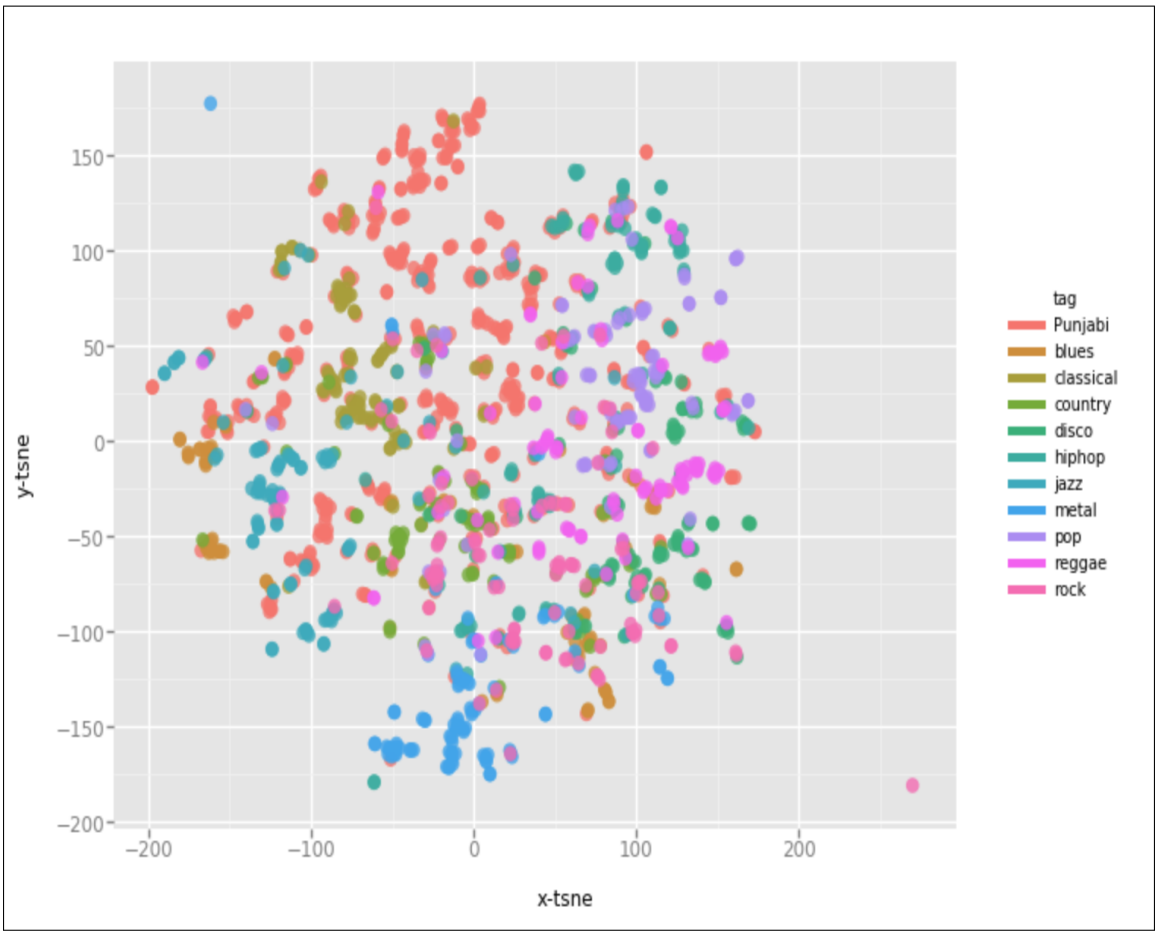

Figure 4.4: tSNE Visualization for features extracted Experiment 2 before PCA

Even after performing dimensionality reduction using PCA, clustering for Punjabi songs is poor.

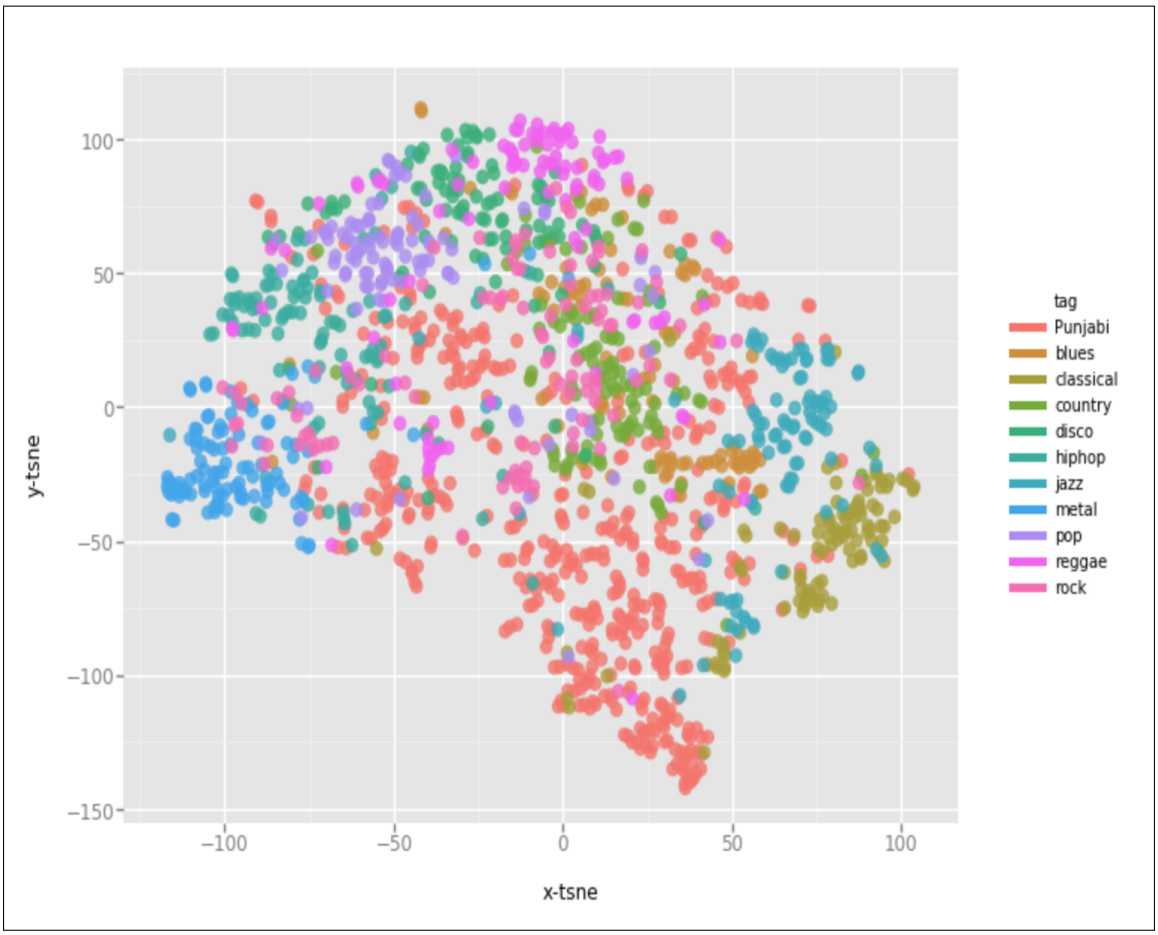

Figure 4.5: tSNE Visualization for features extracted Experiment 2 after PCA

Recommendations also don't seem to respect genre.

Example1: A slow song with flute playing in background and bells even though Hindi gets Punjabi songs with fairly different vocal style as recommendations.

Example2: Song with drum beats and a string instrument is construed as similar to an EDM song with a different string instrument and bass style beats.

4.6 Experiment 3:

In this experiment we use transfer learning setup to get spectrograms for Gtzan Genre dataset and train the model to predict genres. A 63% accuracy is achieved on the dataset.

4.7 Experiment 4:

In this experiment a significant amount was devoted to gathering meta tags for songs. Based on quality of tags, we ended up with 250 songs with acceptable tag quality. Quality parameters for tags were that the tag should represent a genre for e.g. Bhangra, UK underground, etc. and it should have at least 25 member tracks.

The tags are: bhangra, world, punjabi folk, india, hindi, world music, punjabi, desi, bollywood and world music.

This experiment was a multi label problem with each audio file having multiple labels.

For generating predictions we treat anything with a probability of greater than or equal to 50% as being true.

With this assumption we achieve a 70% on the multi-label classification.

Using the weights learned from this training exercise, we create a feature extractor. To do this we export weights of each activation layer in the model and concatenate them to create a feature extractor.

We test this feature extractor on our super small Punjabi, Hindi, Bengali dataset and there is clear visible improvement in clustering. Recommendations are also improved.

See visualization in Figure 4.6

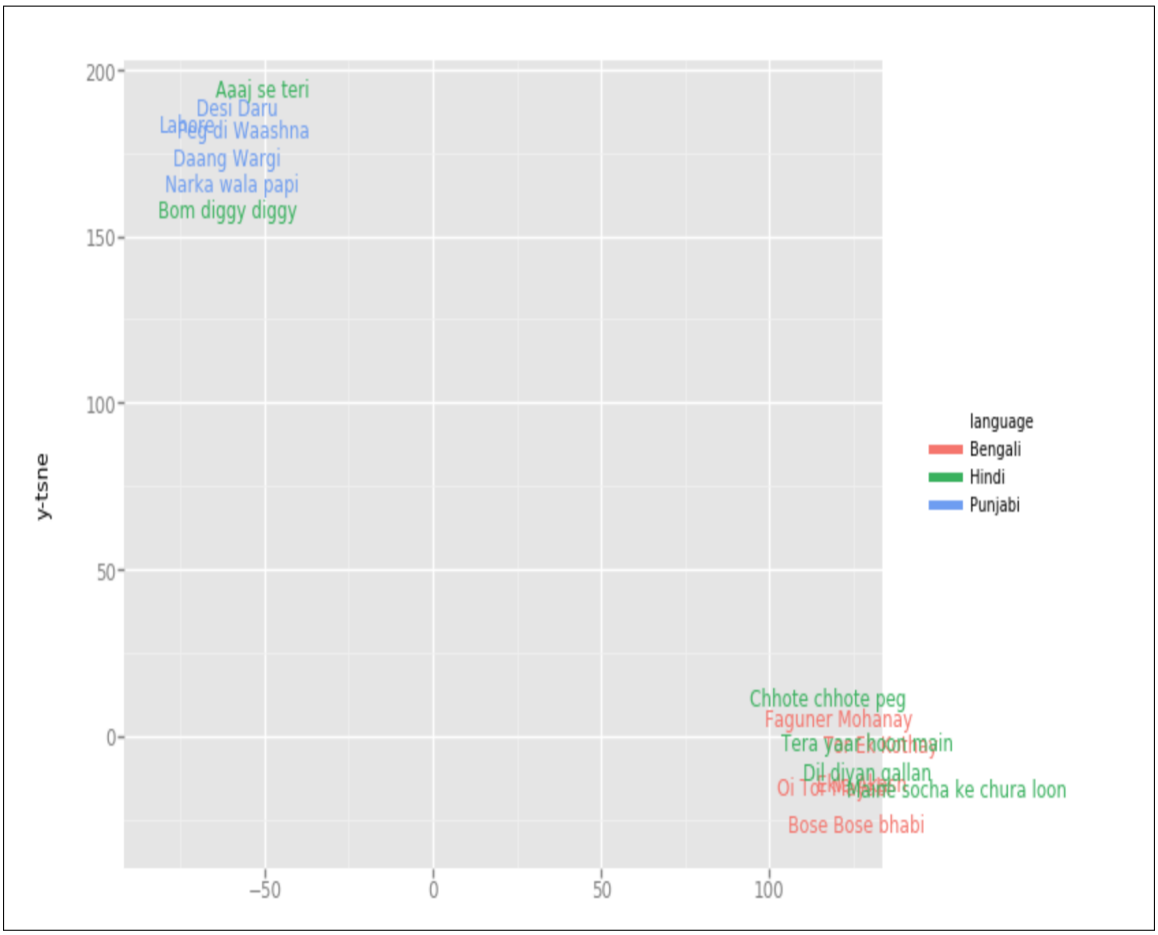

Figure 4.6: tSNE Visualization for Experiment 4

4.8 Experiment 5:

For this experiment we use a bigger dataset containing 750 songs. Unlike the previous dataset, in this dataset each audio file belongs to a single class or genre.

The list of genres is: 90s_pop, bhangra, bollywood, kirtan, punjabi_rap, sufi, UK_Asian

We achieve a **50%** accuracy on the test set for this dataset.

4.9 Experiment 6:

For this experiment model_DS1 is model from Experiment 4, and model_DS2 is model from Experiment 5. model_DS1 is trained on DS1 dataset and model_DS2 is trained on DS2 dataset.

We will refer to these names in this experiment.

Because of limited amount of data available, we didn't have a subset of data that was untouched in the train and test cycle. To properly validate our findings we needed data that was completely outside what the model has seen. But it was also essential to use most amount of data for training and we couldn't just leave data out. As a workaround we use DS2 as test data for model_DS1. And DS1 as test data for model_DS2.

We use the weights from model_DS1 and model_DS2 to build a feature extractor. We generate recommendations from test datasets using these feature extractors. We tried several seed songs and recommendations from model_DS2 are observed to be really close to the seed song.

4.10 Experiment 7:

The goal of this experiment was to study features learned by filters at each layer. For this experiment the dataset we use is GTZAN Genre dataset appended to entire DS1complete. We study recommendations from just layer 1, layer 1+2, layer 1+2+3, and layer 1+2+3+4 by using an audio file from DS1complete as seed. Earlier layers learn simpler audio features and therefore recommendations are based on more broad genre and later layers return features based on the specific genre. As an example a Punjabi sufi or kirtan would be close to classical genre based on earlier layers and close to kirtan/sufi in later layers.

Chapter 5

Evaluation and Conclusion

We proposed a recommender system and a transfer learning system fine-tuned for Punjabi music. Transfer learning system trained on Million Song Dataset subset was fine-tuned on Punjabi music and then the weights learned in each layer were concatenated to be used as a feature extractor. These features were used in conjunction with cosine similarity function to recommend songs whose predicted latent factors are close to the seed song's latent factors. In the experiments, the fine-tuned system worked well on songs it had never seen before with the recommendations being of good quality overall.

The main goal of this project was to research content based recommendations for Punjabi music and the findings can be extended to make existing recommendation systems better for ethnic music. As with recommendation systems, user testing is the ultimate test for any system. While the recommendations generated by the trained model are quite good, user taste can be hard to predict.

A quantitative summary for various statistics in the project:

1. 70% accuracy on multi-label classification problem
2. 50% accuracy on the multi-class genre classification problem.

A qualitative summary for recommendations by the model:

1. Model trained on DS2 provides better recommendations than model trained on DS1complete and the model trained on MSD subset.
2. One clear improvement is some recommendations returned by MSD model are way off. Model trained on DS2 has very few of these outliers.

3. Since we use the first 30 seconds of audio to generate mel-spectrograms, songs that sound quite different later don't get good recommendations.

4. User study needed to understand quality better.

Chapter 6

Future Work

To aid more research in the field of Punjabi music information retrieval a good dataset is needed. For this project the number of data members per class label was quite small and a test set is needed to create a reliable benchmark. Creating a copyright free dataset of Punjabi music with good quality of meta-tags would be ideal. Second problem that I want to work on is train custom CNN models and find which models work best for Punjabi music.

Siamese networks [37] work well for tasks that involve finding similarity between different entities. Siamese networks can therefore be used to learn similarity between audio (represented as vectors). For future work we want to work on an architecture to learn a similarity metric on audio vectors.

For this study, the metrics used to evaluate models were accuracy in predicting genre and to evaluate the recommendations we mostly relied on user studies. Such a metric doesn't scale well and therefore for future work we want to use Mean Average Precision to evaluate recommendations. Since a good characteristic of recommendation for this task would be getting a Punjabi song as recommendation for a Punjabi seed song, using Mean Average Precision to evaluate the recommendations makes sense. For e.g. after mixing Punjabi songs with Western songs, if a Western song is recommended for a Punjabi seed song, that is a bad recommendation. Mean Average Precision works well for tasks returning a ranked ordering of items with each item being either relevant or irrelevant and this metric would therefore be a good choice to evaluate our recommendations in the future.

Bibliography

[1] Billboard 200 makeover: Album chart to incorporate streams track sales. URL https://www.billboard.com/articles/columns/chart-beat/6320099/billboard-200-makeover-streams-digital-tracks.

[2] Consumers in canada are turning up the volume on music. URL http://www.nielsen.com/ca/en/insights/news/2017/consumers-in-canada-are-turning-up-the-volume-on-music.html.

[3] T-series. URL http://youtube.wikia.com/wiki/T-Series.

[4] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. Research-paper recommender systems: a literature survey. *International Journal on Digital Libraries*, 17(4):305–338, Nov 2016. ISSN 1432-1300. doi: 10.1007/s00799-015-0156-0. URL https://doi.org/10.1007/s00799-015-0156-0.

[5] Yading Song, Simon Dixon, and Marcus Pearce. A survey of music recommendation systems and future perspectives. In *9th International Symposium on Computer Music Modeling and Retrieval*, volume 4, 2012.

[6] URL https://web.archive.org/web/20090926230905/http://www.statpak.gov.pk/depts/pco/index.html.

[7] Orgi. Statement 7. URL http://www.censusindia.gov.in/Census_Data_2001/Census_Data_Online/Language/Statement7.aspx.

[8] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.

[9] Chris Johnson. Algorithmic music recommendations at spotify, Jan 2014. URL `https://www.slideshare.net/MrChrisJohnson/algorithmic-music-recommendations-at-spotify`.

[10] Douglas Turnbull, Luke Barrington, and Gert RG Lanckriet. Five approaches to collecting tags for music. In *ISMIR*, volume 8, pages 225–230, 2008.

[11] Michael A Casey, Remco Veltkamp, Masataka Goto, Marc Leman, Christophe Rhodes, and Malcolm Slaney. Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE*, 96(4):668–696, 2008.

[12] Suvarnalata Rao and Preeti Rao. An overview of hindustani music in the context of computational musicology. *Journal of new music research*, 43(1):24–33, 2014.

[13] Xavier Serra. A multicultural approach in music information research. In *Klapuri A, Leider C, editors. ISMIR 2011: Proceedings of the 12th International Society for Music Information Retrieval Conference; 2011 October 24-28; Miami, Florida (USA). Miami: University of Miami; 2011.* International Society for Music Information Retrieval (ISMIR), 2011.

[14] Gibb Stuart Schreffler. Vernacular music and dance of punjab. *Journal of Punjab Studies*, 11(2):197–213, 2004.

[15] Madhuja Mukherjee. The architecture of songs and music: soundmarks of bollywood, a popular form and its emergent texts. *Screen Sound Journal*, 3:9–34, 2012.

[16] Ananya Jahanara Kabir. Musical recall: postmemory and the punjabi diaspora. *Alif: Journal of Comparative Poetics*, (24):172–191, 2004.

[17] Anjali Gera Roy. Bhangranation new meanings of punjabi identity in the twenty first century. *Journal of Punjab Studies*, 19(1), 2012.

[18] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In *Advances in neural information processing systems*, pages 2643–2651, 2013.

[19] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Transfer learning for music classification and regression tasks. *arXiv preprint arXiv:1703.09179*, 2017.

[20] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

[21] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[22] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on speech and audio processing*, 10(5):293–302, 2002.

[23] Brian McFee, Luke Barrington, and Gert Lanckriet. Learning content similarity for music recommendation. *IEEE transactions on audio, speech, and language processing*, 20(8):2207–2218, 2012.

[24] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, pages 18–25, 2015.

[25] Keunwoo Choi, Deokjin Joo, and Juho Kim. Kapre: On-gpu audio preprocessing layers for a quick implementation of deep neural network models with keras. In *Machine Learning for Music Discovery Workshop at 34th International Conference on Machine Learning*. ICML, 2017.

[26] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[27] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10): 1995, 1995.

[28] URL `http://cs231n.github.io/convolutional-networks/`.

[29] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11–26, 2017.

[30] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.

[31] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1717–1724. IEEE, 2014.

[32] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

[33] François Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

[34] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[35] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[36] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[37] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. In *Advances in Neural Information Processing Systems*, pages 737–744, 1994.