On Generalist and Domain-Specific Music Classification Models and Their Impacts on Brazilian Music Genre Recognition

Diego Furtado Silva[1] **, Angelo Cesar Mendes da Silva**[2] **,**
Luís Felipe Ortolan[1] **, Ricardo Marcondes Marcacini**[2]

[1]Departamento de Computação – Universidade Federal de São Carlos, Brazil
Rod. Washington Luís, km 235, Jardim Guanabara – 13565-905 São Carlos, SP

[2]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo, Brazil
Av Trabalhador São-carlense, 400, Centro – 13566-590 São Carlos, SP

diegofs@ufscar.br, luisfelipe.ortolan@gmail.com, {angelo.mendes, ricardo.marcacini}@usp.br

Abstract. *Deep learning has become the standard procedure to deal with Music Information Retrieval problems. This category of machine learning algorithms has achieved state-of-the-art results in several tasks, such as classification and auto-tagging. However, obtaining a good-performing model requires a significant amount of data. At the same time, most of the music datasets available lack cultural diversity. Therefore, the performance of the currently most used pre-trained models on underrepresented music genres is unknown. If music models follow the same direction that language models in Natural Language Processing, they should have poorer performance on music styles that are not present in the data used to train them. To verify this assumption, we use a well-known music model designed for auto-tagging in the task of genre recognition. We trained this model from scratch using a large general-domain dataset and two subsets specifying different domains. We empirically show that models trained on specific-domain data perform better than generalist models to classify music in the same domain, even trained with a smaller dataset. This outcome is distinctly observed in the subset that mainly contains Brazilian music, including several usually underrepresented genres.*

1. Introduction

Music classification is one of the most investigated subtasks of Music Information Retrieval (MIR) [1]. Like other domains that rely on complex data, such as Computer Vision (CV) and Natural Language Processing (NLP), content-based Music classification has achieved significant improvements through deep neural networks [2].

A limitation of deep learning techniques is that they usually need a large volume of training data to obtain a good-performing model. Therefore, applying deep learning requires a significant effort of labeling the data or, at least, crowdsourcing the labeling responsibility. Both cases lead to a more severe difficulty for training models for underrepresented music genres in music classification.

A possible solution to circumvent this problem is the use of pre-trained models [3, 4]. In this scenario, we use a model trained in a large dataset and fine-tune its weights using the dataset we have at hand. This procedure, called transfer learning [5], is increasingly becoming standard in CV, NLP, and MIR applications. Besides, it is possible to use the pre-trained model in some scenarios with no fine-tuning. This procedure is called zero-shot learning [6]. Finally, recent advancements in self-supervised learning have led researchers to create models where other researchers or practitioners may use the model as a feature extractor and apply the learned representation in different downstream tasks.

However, several efforts on CV and NLP have shown that the transference of knowledge is limited, mainly when we use a pre-trained model in a completely different application domain [7, 8]. Usually, fine-tuning from a domain-specific model (similar to the one it will be applied) leads to better results than generalist models.

Although pre-trained models and their ability to transfer knowledge are well-studied on CV and NLP applications, these techniques are less explored in MIR applications. A few works have used CV-based models, like VGG, to classify music data. More recent papers achieved state-of-the-art music classification using musically motivated models, creating evidence that supports domain-specific models for music data [9]. However, there is no effort to assess the capabilities of domain-specific or general music classification models on low-represented music genres.

This work presents a study of the impact of domain-specific training of music models for Multi-label Music Genre Classification. Notably, we investigate how a widely-used deep architecture behaves on different Western music genres when trained using songs in other groups. For example, we assume that songs composed in English usually have significant differences from songs written in Portuguese, so we apply different language-based filters to train general and specific models.

More importantly, the portion of our dataset composed in Portuguese includes Brazilian genres usually low-represented in music datasets, such as samba, sertanejo, and axé. Moreover, we also understand that songs in the same genre, such as rock and reggae, may present different characteristics if written in English or Portuguese, and the domain-specific models should capture these differences.

2. Background and Related Work

In the last few decades, we have witnessed a significant increase in Machine Learning applications in Computer Music and Music Information Retrieval tasks [10, 11, 1]. The

most common approach in this context is using a supervised learning algorithm. This category of techniques may be applied in a plethora of tasks, such as genre classification [12, 13], auto-tagging [14, 4], instrument identification [15], mood [16] and emotion classification [17], chord recognition [18], and popularity estimation [19].

Early systems usually applied a feature extraction step to obtain a structured dataset before submitting it to "traditional" Machine Learning algorithms [20]. For instance, different features like signal or spectrum parameters, mel-frequency cepstrum coefficients (MFCC), and psychoacoustic characteristics [21] may be extracted using different windowing and aggregation strategies from the raw audio in genre classification [22, 23, 12]. However, aggregating the windows and the influence of distinct features and their combinations make music classification difficult.

For this reason, the community has moved to deep learning algorithms [3, 2]. As a result, most of the literature that defines the state-of-the-art of music classification is based on learning a deep neural network-based model [3, 24, 25, 26, 27, 4, 28, 13]. The main advantage of using these models is the ability to learn the best representation to solve the problem at hand.

Several recent works have used neural models trained in a dataset to deal with another dataset, especially with only a few labeled examples. For example, we may rely on transfer learning [5, 29, 30] or zero-shot [6] learning strategies to use these models. Moreover, we may leverage on different supervision levels, such as supervised [31] or self-supervised learning [32, 33] to learn these models.

Considering the resemblance of these strategies to how researchers and practitioners have used *language models* to approach NLP problems, we refer to these models for MIR tasks as *music models*. Among the various music models found in the literature, we highlight `musicnn` [4], which has received considerable attention.

`musicnn`[1] is a set of pre-trained musically motivated convolutional neural networks designed for music auto-tagging. A researcher interested in using this music model may train it from scratch, fine-tune its weights, or use it as a feature extractor. Its default output consists of a taggram consisting of each pre-defined tag's strength (prediction score) in each time window of the input recording. Alternatively, the user may extract the penultimate layers of the model and use them as low-level features.

One limitation of these music models, including `musicnn`, is that they are majorly trained using American-like music, such as pop, hip-hop, and blues songs. As a result, when a song from a different culture is presented to the model, it may fail in extracting suitable features to describe it. For instance, Brazilian music is rich in influences worldwide. Consequently, Brazilian musicians have created several regional genres, such as forró, axé, samba, funk carioca, bossa nova, and sertanejo. Unfortunately, the `musicnn` models have no tags regarding these genres and the instruments used to record most of the songs in them.

A few papers noticed the lack of Brazilian music in the MIR literature. For instance, Conceição et al. [34] studied the most used music classification datasets and stated a lack of cultural diversity in them. So, the authors created a database containing 613 Brazilian songs from 6 genres. Using low-level feature extraction and traditional Machine Learning algorithms, the authors achieved slightly better results using only the Brazilian songs instead of merging them with a public dataset.

It was not the first time authors approached the lack of cultural diversity on music datasets, especially considering Brazilian or Latin music. For instance, the Latin Music Database (LMD) [35] was proposed to fill this lack. The LMD contains features extracted from 3160 songs in 10 Latin genres, such as salsa and samba. Later, de Sousa, Pereira, and Veloso [36] proposed the Brazilian Music Dataset, containing 120 songs labeled in 7 musical genres. Other datasets of Brazilian genres were presented for different tasks, such as the Brazil Northeast data [37], the Brazilian Popular Music [38], the SambaSet [39], and the Forró em Vinil Dataset [40].

Recently, the 4MuLA dataset was proposed: A Multitask, Multimodal, and Multilingual Dataset of Music Lyrics and Audio Features [41] which has richer musical information than the datasets aforementioned. 4MuLA makes available 96,458 Brazilian and international songs that belong to 15,310 artists and are organized into 76 genres in its full version. The collected songs were from the Brazilian portal Vagalume. All pieces have a set of preprocessed metadata, lyrics, and acoustic features to represent them in different scenarios.

3. Domain-Specific Models

This work is based on the widely spread idea that domain-specific deep models usually perform better than generalist ones. This idea is remarkably studied on Natural Language Processing (NLP) tasks. The current state-of-the-art NLP algorithms are based on pre-trained language models, such as Bidirectional Encoder Representations from Transformers (BERT) [42].

However, BERT (and other language models) present several flaws identified in the literature. For instance, multilingual models have deficiencies in modeling underrepresented languages [8]. Usually, replacing the multilingual model with a specialized monolingual version improves the performance on different NLP tasks [43].

Besides, training the language model on texts of a specific domain also leads to better results. For this reason, the literature presents a vast diversity of domain-specific BERT. Some examples of knowledge domains used to train domain-specific language models are biomedicine [44], finance [45], and scientific writing [46].

[1] `https://github.com/jordipons/musicnn`

This work hypothesizes that music models behave like language models for domain-specific data. To evaluate this hypothesis, we use a dataset containing several songs of Brazilian music. We also use a similar dataset containing other songs concentrated on American music, but not limited to. For this, we separate songs from the 4MuLA dataset [41] composed in Portuguese and English. The complete dataset, which contains these and other languages, is used to train a generalist model.

These datasets are used to train `musicnn` [4], a musically motivated convolutional neural network designed for auto-tagging. Using the different datasets, we will evaluate how `musicnn` behaves on various combinations of general and specific-domain training and test.

Figure 1 describes the `musicnn`'s architecture. The input of the deep network is a log-scaled melspectrogram. First, it is submitted to musically motivated convolutional layers [47], creating the first representation that may be used, named by the authors as front-end features. Next, these features are passed through layers of fully connected (dense) unities, transforming the front-end into mid-end features. Finally, after pooling and dense layers, the network output a taggram.

The taggram represents the predicted score of each of the 50 tags comprised in the training dataset in short time windows. In this work, we are interested in the genres of the assessed songs. So, we substitute the last layer to adapt it to the number of genres comprised by each subset (general or domain-specific). Then, we used the mean value, across all the windows, for each genre as the final scores.

4. Experimental Setup

We used the 4MuLA dataset in our experiments due to the completeness of its information. We chose to use the small version, which concentrates songs from artists in the top 100 ranking positions according to the source of data Vagalume website[2]. In this version, each audio has 30 seconds and is represented by a melspectrogram. The 4MuLA small version has 9661 songs that belong to 419 artists and is organized in 50 different genres.

To obtain the domain-specific models, we first needed to define the rule to specify each domain. For this, we explore the main characteristics of the dataset. Table 1 show a summary of small 4MuLA. We note that each song in the dataset contains a top genre and (possibly) a list of other/secondary genres.

While we could use specific genres, we opted to use the language the lyrics were composed as a filter. We understand that it brings sufficient information to define the domains to verify the quality of the studied model on Brazilian music. Besides, considering English and Portuguese as the filters, we create two specific-domain datasets with a similar number of instances.

[2]https://www.vagalume.com.br/

Table 1: Overview of 4MuLA small

Feature	Amount
Number of songs/instances	9661
Number of songs (EN)	4722
Number of songs (PT)	4654
Number of songs (others)	285
Unique artists	491
Unique genres	50
Max instances in a single “top” genre	1169
Min instances in a single “top” genre	11
Avg instances in a single “top” genre	189.43
Std instances in a single “top” genre	260.86

On the other hand, we recognize that we may find Bossa Nova and other Brazilian genres written in English. However, filtering by language also has the impact of separating songs of the same genre with different influences. For instance, Brazilian rock music may reflect a significant influence from local music. So, American or British rock music, for example, may sound considerably different from Brazilian songs in the same genre.

Therefore, we created three sub-datasets. The first one uses the whole 4MuLA small, comprising songs from all the languages and, consequently, varied cultures. We refer to this dataset and the models created from it using the tag `All`. From this dataset, we created random training and test partitions, named as `All-train` and `All-test`, respectively.

In a second stage, we used `All-train` and `All-test` to create the domain-specific data. By filtering the data regarding songs written in Portuguese, we obtained the datasets `PT-train` and `PT-test`. Similarly, we applied the filter for English and obtained the datasets `EN-train` and `EN-test`. Figure 2 illustrates this procedure.

Finally, we recall we use `musicnn` as the base model of our evaluation. While the currently available `musicnn` may be used as a pre-trained model, we opted for initially training it from scratch. This option relies on avoiding possible intersections of the songs in our dataset and the datasets used to train `musicnn`, as well as the possibility of the pre-trained model already reflects some characteristics of our domain-specific data. Moreover, we consider the number of examples in 4MuLA small as sufficient for training a suitable model.

Therefore, we used `All-train` to create our pre-trained model, which we refer to as `musicnn-All`. For the sake of comparison, we also trained specific-domain models from scratch. Using the datasets `PT-train` and `EN-train`, we trained the models `musicnn-PT` and `musicnn-EN`, respectively.

Finally, we noticed that we repeated the experiments ten times to present results robust to variations caused by randomness.

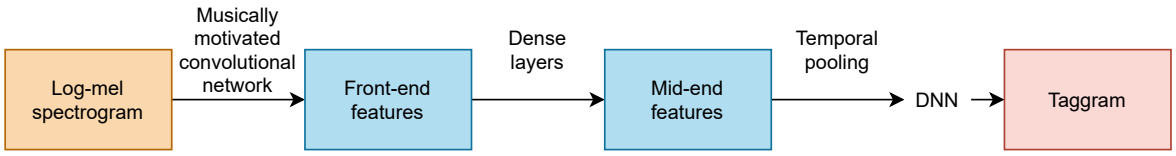

Figure 1: General architecture of `musicnn`, the model we used as based in our experiments

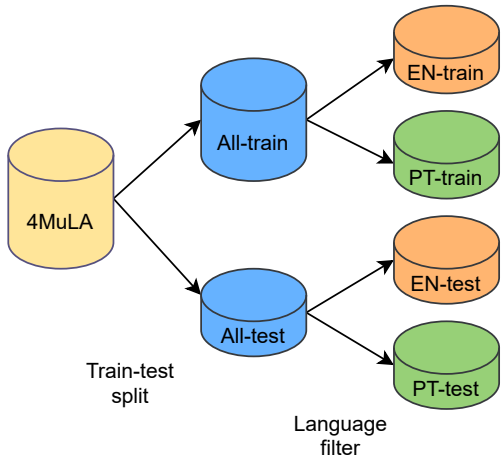

Figure 2: We split our dataset in train and test partitions and apply filter on these subsets to obtain domain-specific data

5. Results and Discussion

Using the described experimental setup, we assessed the area under the Receiver Operating Characteristic Curve (ROC-AUC) and the Precision-Recall (PR-AUC) curves obtained by each model on each test set. Table 2 presents the average results.

Table 2: Detailed results

Model	Test set	ROC-AUC	PR-AUC
`musicnn-All`	`All-test`	0.68	0.43
	`EN-test`	0.65	0.45
	`PT-test`	0.64	0.34
`musicnn-EN`	`All-test`	0.60	0.36
	`EN-test`	0.65	0.44
	`PT-test`	0.53	0.27
`musicnn-PT`	`All-test`	0.63	0.39
	`EN-test`	0.56	0.37
	`PT-test`	0.69	0.39

We plot these results differently aggregated in separate charts for visualization and, consequently, better interpretation of the results. Figure 3a presents the ROC-AUC performances aggregated by model so that we can compare their performances for each language filter. Figure 3b aggregates the results by language filter.

When observing the ROC-AUC, we note that the best performance of each model occurs on the test set obtained by the same filter used in the training set. This behavior is expected for `musicnn-EN` and `musicnn-PT` for several reasons. We can interpret this result as anticipated evidence that zero-shot learning does not work when the pre-trained models are domain-specific, especially if applied on another domain-specific dataset. For instance, `PT-test` comprises several songs from genres not included in `EN-train`. Even the genres included in both datasets may present differences on the separate data, as previously discussed. So, these songs appear on both generalist and specific (for the other language) test sets. Once these songs represent a fraction of `All-test`, the performance decrease is lower on this data.

On the other hand, the model trained using all data brings more interesting results. While we could expect better results on `PT-test` and `EN-test`, the best performance of `musicnn-All` was achieved on the `All-test` dataset. It means that, for some reason, `musicnn-All` performed well on songs that are neither in `PT-train` nor in `EN-train`.

Regarding each test dataset, using a model trained on a domain generally leads to better ROC-AUC rates for the same domain. This fact is clearly observed for `All-test` and `PT-test`. For instance, `musicnn-PT` is five percentage points better than `musicnn-All` for `PT-test`. However, the results obtained by the generalist and specific models are similar on `EN-test`.

Besides, `musicnn-PT` performs better than `musicnn-EN` on `All-test`. Since the subsets are not unbalanced, we can conclude that the songs comprised by `PT-train` lead to a better model (for general music) than the songs on `EN-train`. It possibly happens because the Portuguese songs, at least for this dataset, are more diverse.

When we observe the PR-AUC results, we obtain similar conclusions. However, there is a slight inversion in the results obtained on `EN-test`. The best PR rate on `EN-test` was obtained by `musicnn-All` (Figure 4b). At the same time, the highest PR-AUC obtained by `musicnn-All` was on `EN-test` (Figure 4a), not on `All-test` like for the ROC-AUC.

A general observation made on the results is that all the ROC-AUC are significantly higher than the PR-AUC. Won et al. [9] shown the same phenomenon when comparing diverse deep learning models for music auto-tagging. Moreover, they observed some inversions of result interpretations, like those we discussed for `EN-test`. However, the authors did not propose any solution for a better evaluation.

We added all the confusion matrices obtained in

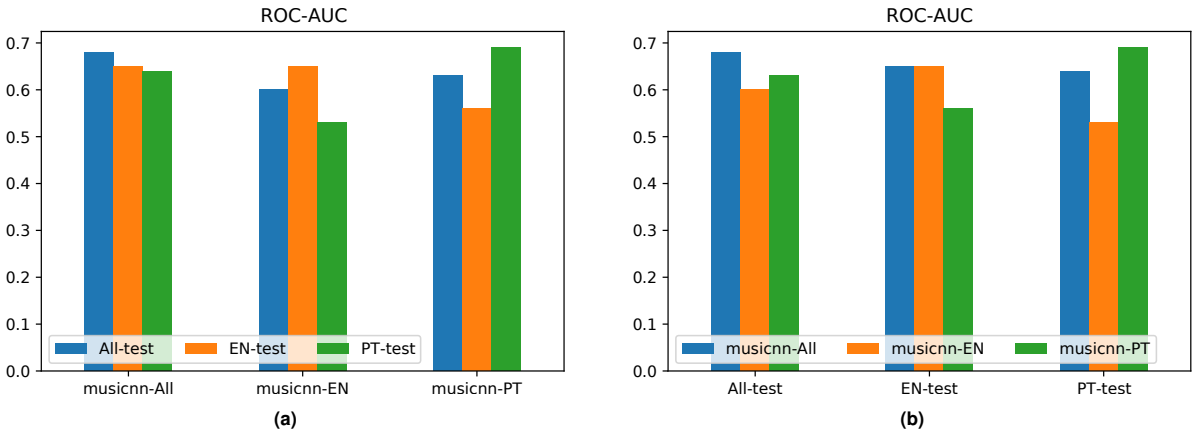

Figure 3: ROC-AUC obtained in our experimental evaluation, aggregated by model (a) and dataset (b)

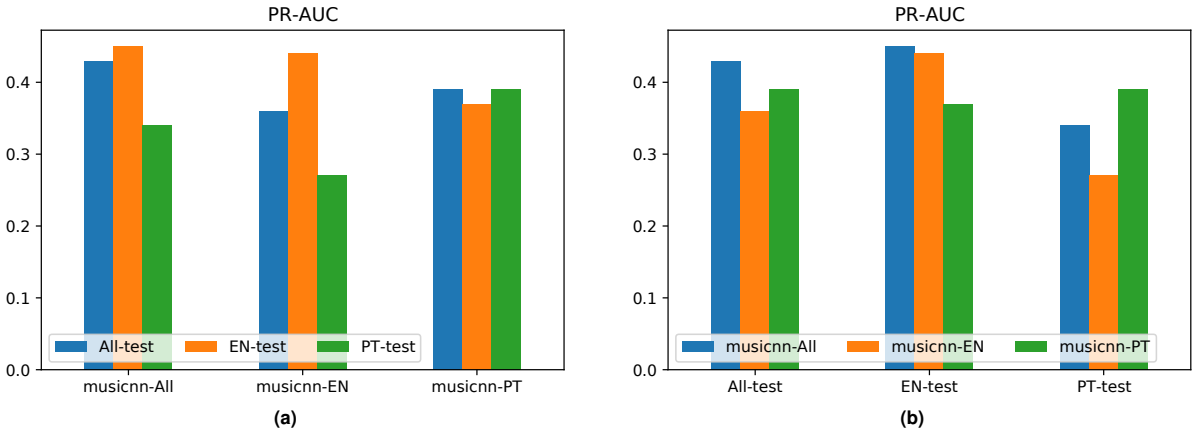

Figure 4: PR-AUC obtained in our experimental evaluation, aggregated by model (a) and dataset (b)

our experimental evaluation in the appendix to a better and detailed interpretation of the results. Carefully analyzing these matrices helps us interpret some outcomes that are not evident in the bar plots and tables. For instance, we can observe that our models obtain a significant number of false negatives and only a few false positives, causing low recall rates. It explains why the PR-AUC values are notably lower than the ROC-AUC.

6. Final Remarks

We presented an experiment on how a known music model behaves in general and specific domains. Notably, we filtered songs by the language of their lyrics as a heuristic to create a dataset containing several recordings of Brazilian music and another one more similar to well-known datasets, where the songs were composed in English.

Our results show that specific-domain models, in general, are more suitable to classify domain-specific data. Training a model from scratch using only the Portuguese portion of the data drove us to build better models for the same type of songs than training the same model using the same data and, additionally, data from other languages. At the same time, it is not clear for songs composed in English, which leads us to think that Brazilian music's complexity played a significant role in the model training.

Although the evaluated model performs similarly to other deep learning models for music classification [9], we intend to perform similar experiments using other architectures. Moreover, establishing and evaluating different strategies to define the specific domains is also left to future work.

Furthermore, we aim to understand the studied phenomena better through model explanations, examining the differences between general and specific domain models. Then, we may evaluate the feasibility of funding these models. Finally, we will include other modalities, like the songs' lyrics, and perform similar studies.

7. Acknowledgments

This work was financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code PROEX-12049601/D, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) – Finance Code 2020/07911-2.

References

[1] YV Srinivasa Murthy and Shashidhar G Koolagudi. Content-based music information retrieval (cb-mir) and its applications toward the music industry: A review. *ACM Computing Surveys*, 51(3):1–46, 2018.

[2] Geoffroy Peeters. The deep learning revolution in mir: The pros and cons, the needs and the challenges. In *International Symposium on Perception, Representations, Image,*

Sound, Music, pages 3–30. Springer International Publishing, 2021.

[3] Sander Dieleman, Philémon Brakel, and Benjamin Schrauwen. Audio-based music classification with a pretrained convolutional network. In *International Society for Music Information Retrieval Conference*, pages 669–674. University of Miami, 2011.

[4] Jordi Pons and Xavier Serra. musicnn: Pre-trained convolutional neural networks for music audio tagging. *arXiv preprint arXiv:1909.06654*, 2019.

[5] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Transfer learning for music classification and regression tasks. In *International Society for Music Information Retrieval Conference*, 2017.

[6] Jeong Choi, Jongpil Lee, Jiyoung Park, and Juhan Nam. Zero-shot learning for audio-based music classification and tagging. In *International Society for Music Information Retrieval Conference*, 2019.

[7] Hemanth Venkateswara, Shayok Chakraborty, and Sethuraman Panchanathan. Deep-learning systems for domain adaptation in computer vision: Learning transferable feature representations. *IEEE Signal Processing Magazine*, 34(6):117–129, 2017.

[8] Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert? In *Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, 2019.

[9] Minz Won, Andres Ferraro, Dmitry Bogdanov, and Xavier Serra. Evaluation of cnn-based automatic music tagging models. In *Sound and Music Computing Conference*, 2020.

[10] Naresh N Vempala and Frank A Russo. Modeling music emotion judgments using machine learning methods. *Frontiers in psychology*, 8:2239, 2018.

[11] Bob L Sturm, Oded Ben-Tal, Úna Monaghan, Nick Collins, Dorien Herremans, Elaine Chew, Gaëtan Hadjeres, Emmanuel Deruty, and François Pachet. Machine learning research that matters for music creation: A case study. *Journal of New Music Research*, 48(1):36–55, 2019.

[12] James Bergstra, Norman Casagrande, Dumitru Erhan, Douglas Eck, and Balázs Kégl. Aggregate features and a da b oost for music classification. *Machine learning*, 65(2-3):473–484, 2006.

[13] Jingxian Li, Lixin Han, Xiaoshuang Li, Jun Zhu, Baohua Yuan, and Zhinan Gou. An evaluation of deep neural network models for music classification using spectrograms. *Multimedia Tools and Applications*, pages 1–27, 2021.

[14] Douglas Eck, Thierry Bertin-Mahieux, and Paul Lamere. Autotagging music using supervised machine learning. In *International Conference of Students of Systematic Musicology*, pages 367–368, 2007.

[15] Vincent Lostanlen, Joakim Andén, and Mathieu Lagrange. Extended playing techniques: the next milestone in musical instrument recognition. In *International Conference on Digital Libraries for Musicology*, pages 1–10, 2018.

[16] Xiao Hu, Kahyun Choi, and J Stephen Downie. A framework for evaluating multimodal music mood classification. *Journal of the Association for Information Science and Technology*, 68(2):273–285, 2017.

[17] Renato Panda, Ricardo Malheiro, and Rui Pedro Paiva. Novel audio features for music emotion recognition. *IEEE Transactions on Affective Computing*, 11(4):614–626, 2018.

[18] Johan Pauwels, Ken O'Hanlon, Emilia Gómez, Mark Sandler, et al. 20 years of automatic chord recognition from audio. In *International Society for Music Information Retrieval Conference*, 2019.

[19] Carlos Soares Araujo, Marco Cristo, and Rafael Giusti. Predicting music popularity on streaming platforms. In *Brazilian Symposium on Computer Music*, pages 141–148. SBC, 2019.

[20] Zhouyu Fu, Guojun Lu, Kai Ming Ting, and Dengsheng Zhang. A survey of audio-based music classification and annotation. *IEEE transactions on multimedia*, 13(2):303–319, 2010.

[21] Martin McKinney and Jeroen Breebaart. Features for audio and music classification. In *International Society for Music Information Retrieval Conference*. Johns Hopkins University, 2003.

[22] George Tzanetakis, Andrey Ermolinskyi, and Perry Cook. Pitch histograms in audio and symbolic music information retrieval. *Journal of New Music Research*, 32(2):143–152, 2003.

[23] Kristopher West and Stephen Cox. Features and classifiers for the automatic classification of musical audio signals. In *International Society for Music Information Retrieval Conference*, 2004.

[24] Yandre MG Costa, Luiz S Oliveira, and Carlos N Silla Jr. An evaluation of convolutional neural networks for music classification using spectrograms. *Applied soft computing*, 52:28–38, 2017.

[25] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2392–2396. IEEE, 2017.

[26] Jordi Pons, Oriol Nieto, Matthew Prockup, Erik Schmidt, Andreas Ehmann, and Xavier Serra. End-to-end learning for music audio tagging at scale. In *International Society for Music Information Retrieval Conference*, 2017.

[27] Juhan Nam, Keunwoo Choi, Jongpil Lee, Szu-Yu Chou, and Yi-Hsuan Yang. Deep learning for audio-based music classification and tagging: Teaching computers to distinguish rock from bach. *IEEE signal processing magazine*, 36(1):41–51, 2018.

[28] Yang Yu, Sen Luo, Shenglan Liu, Hong Qiao, Yang Liu, and Lin Feng. Deep attention based music genre classification. *Neurocomputing*, 372:84–91, 2020.

[29] Deepanway Ghosal and Maheshkumar H Kolekar. Music genre recognition using deep neural networks and transfer learning. In *Interspeech*, pages 2087–2091, 2018.

[30] Beici Liang and Minwei Gu. Music genre classification using transfer learning. In *IEEE Conference on Multimedia Information Processing and Retrieval*, pages 392–393. IEEE, 2020.

[31] Yilun Zhao and Jia Guo. Musicoder: A universal music-acoustic encoder based on transformer. In *International Conference on Multimedia Modeling*, pages 417–429. Springer, 2021.

[32] Ho-Hsiang Wu, Chieh-Chi Kao, Qingming Tang, Ming Sun, Brian McFee, Juan Pablo Bello, and Chao Wang. Multi-task self-supervised pre-training for music classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 556–560. IEEE, 2021.

[33] Janne Spijkervet and John Ashley Burgoyne. Contrastive learning of musical representations. *arXiv preprint arXiv:2103.09410*, 2021.

[34] Júlia Luiza Conceição, Rosiane de Freitas, Bruno Gadelha, João Gustavo Kienen, Sérgio Anders, and Brendo Cavalcante. Applying supervised learning techniques to brazilian music genre classification. In *2020 XLVI Latin American Computing Conference*, pages 102–107. IEEE, 2020.

[35] Carlos N Silla Jr, Alessandro L Koerich, and Celso AA Kaestner. The Latin Music Database. In *International Conference on Music Information Retrieval*, pages 451–456, 2008.

[36] Jefferson Martins de Sousa, Eanes Torres Pereira, and Luciana Ribeiro Veloso. A robust music genre classification approach for global and regional music datasets evaluation. In *IEEE International Conference on Digital Signal Processing*, pages 109–113. IEEE, 2016.

[37] Jeronimo Barbosa, Cory McKay, and Ichiro Fujinaga. Evaluating automated classification techniques for folk music genres from the Brazilian Northeast. In *Brazilian Symposium on Computer Music*, 2015.

[38] Rodrigo Borges and Marcelo Queiroz. Evolution of timbre diversity in a dataset of brazilian popular music: 1950-2000. In *International Conference of Students of Systematic Musicology*, 2018.

[39] Lucas Maia, Magdalena Fuentes, Luiz Biscainho, Martín Rocamora, and Slim Essid. SAMBASET: A dataset of historical samba de enredo recordings for computational music analysis. In *International Society for Music Information Retrieval Conference*, 2019.

[40] Felipe Falcão, Nazareno Andrade, Flávio Figueiredo, Diego Silva, and Fabio Morais. Measuring disruption in song similarity networks. In *International Society for Music Information Retrieval Conference*, 2020.

[41] Angelo Cesar Mendes da Silva, Diego Furtado Silva, and Ricardo Marcondes Marcacini. 4mula: A multitask, multimodal, and multilingual dataset of music lyrics and audio features. In *Brazilian Symposium on Multimedia and the Web*, pages 145–148, 2020.

[42] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2019.

[43] Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. How good is your tokenizer? on the monolingual performance of multilingual language models. *arXiv preprint arXiv:2012.15613*, 2020.

[44] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234, 2020.

[45] Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. Finbert: A pre-trained financial language representation model for financial text mining. In *International Joint Conference on Artificial Intelligence*, pages 4513–4519, 2020.

[46] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. In *Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing*, pages 3615–3620, 2019.

[47] Jordi Pons, Thomas Lidy, and Xavier Serra. Experimenting with musically motivated convolutional neural networks. In *International Workshop on Content-Based Multimedia Indexing*, pages 1–6. IEEE, 2016.

Appendix

Here, we present all the confusion matrices obtained in our experiments. We note that the values on these matrices were standardized, so all cells' sum equals one.

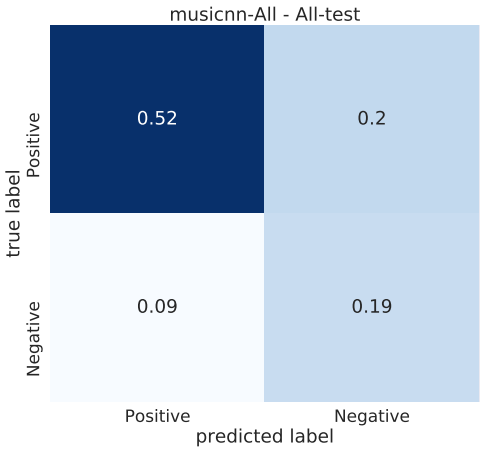

Figure 5: Confusion matrix of musicnn-All on All-test

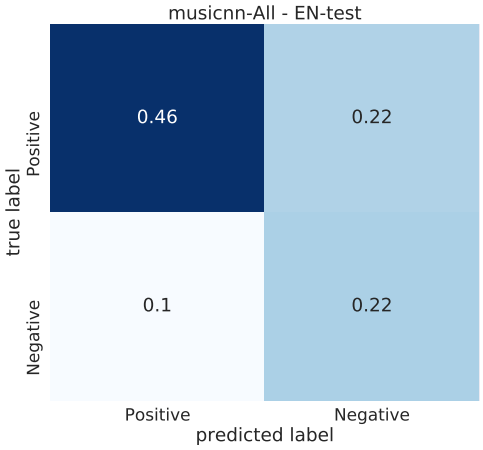

Figure 6: Confusion matrix of musicnn-All on EN-test

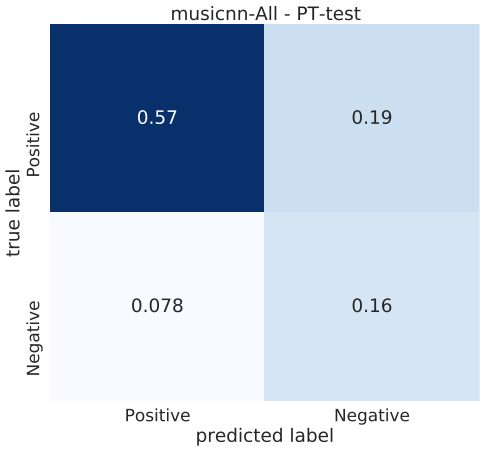

Figure 7: Confusion matrix of musicnn-All on PT-test

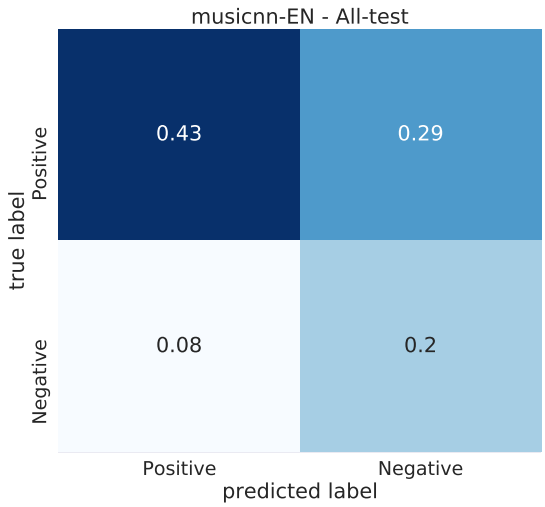

Figure 8: Confusion matrix of musicnn-EN on All-test

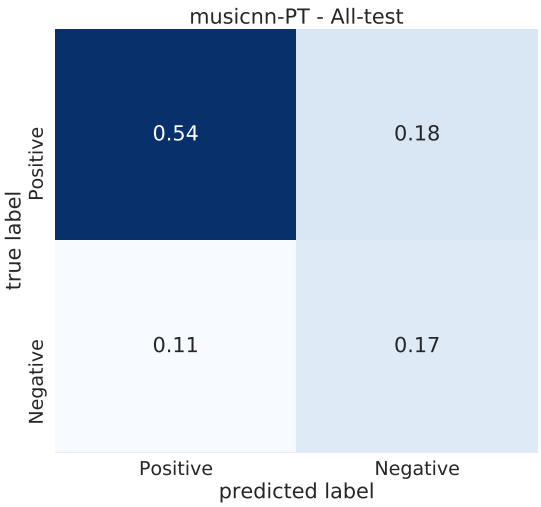

Figure 11: Confusion matrix of musicnn-PT on All-test

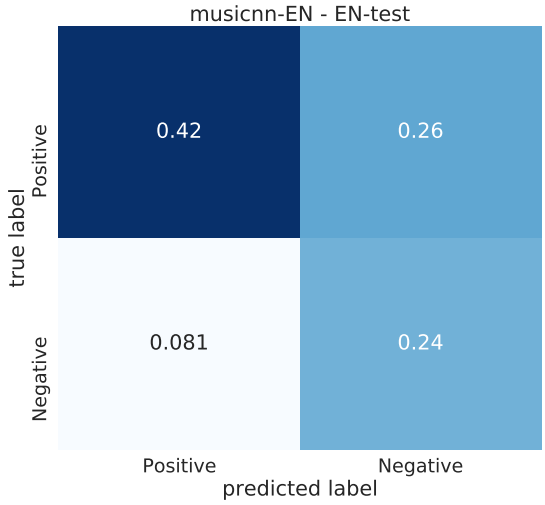

Figure 9: Confusion matrix of musicnn-EN on EN-test

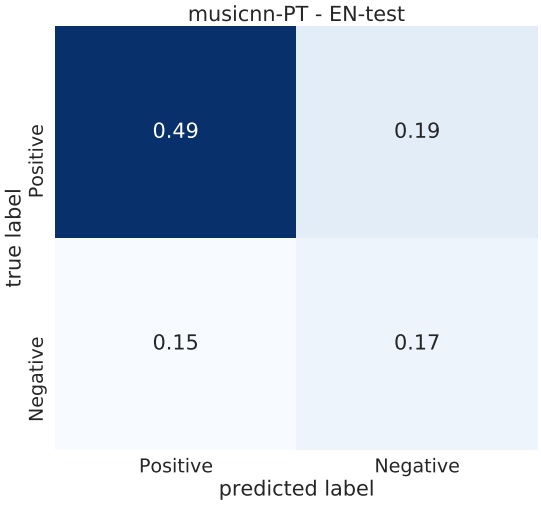

Figure 12: Confusion matrix of musicnn-PT on EN-test

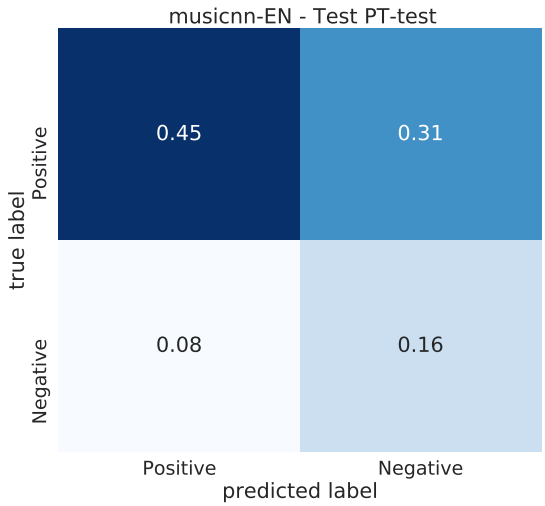

Figure 10: Confusion matrix of musicnn-EN on Test PT-test

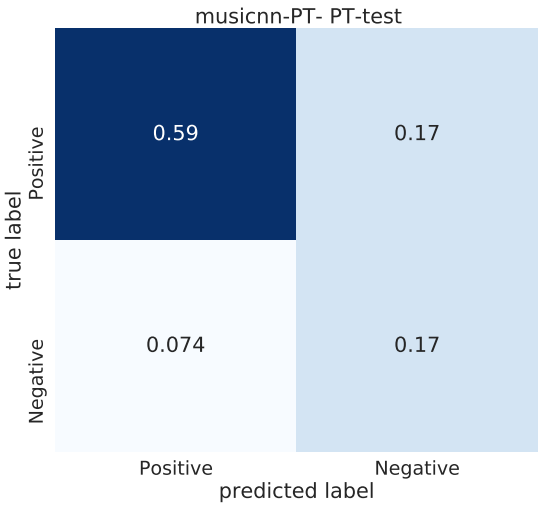

Figure 13: Confusion matrix of musicnn-PT - PT-test