Identificação de Áudio Vocaloid com *Support Vector Machines*: um Estudo de Caso da Hatsune Miku

Felipe V. de Almeida[1], **Victor T. Hayashi**[1]

[1]Escola Politécnica – Universidade de São Paulo (USP)

{felipe.valencia.almeida,victor.hayashi}@usp.br

Resumo. *O processamento de sinais de áudio em conjunto com modelos de aprendizado de máquina tem aplicações em diversas áreas: música, análise forense e análise de fala humana e ruído ambiente. Analisar música de diferentes gêneros pode encorajar investigações interessantes pela comunidade científica, como a investigação de tendências culturais. Este trabalho apresenta uma iniciativa para o processamento de canções vocaloides que ganharam grande popularidade nas redes sociais. Os classificadores Support Vector Machine (SVM) foram treinados em dois experimentos para distinguir canções da vocaloide Hatsune Miku de canções instrumentais e canções de outros vocaloides, apresentando resultados promissores de precisão acima de 80%, que validam a iniciativa.*

1. Introdução

A área de pesquisa de processamento de áudio usando modelos de aprendizado de máquina possui grande interesse da comunidade científica. Há exemplos na literatura em diversos cenários como o monitoramento não invasivo de colônias de abelhas [1], categorização de músicas [2], análise forense [3], além de análises de fala e ruídos ambientais [4].

Por exemplo, o desafio ASVspoof 2017 [5] consistiu em um desafio da área de reconhecimento automático de locutor para obter um classificador que pudesse diferenciar sons provenientes de humanos de sons provenientes de outros equipamentos. Este classificador pode ser usado para mitigar ataques de *replay* físico realizados a partir de um dispositivo próximo à interface de voz.

O processo de extração de informações de músicas necessita de bases de dados representativas e que estejam de acordo com o escopo de trabalho definido [6]. Esforços de pesquisa relevantes foram encontrados para criação de bases de dados de músicas árabe-andaluzas a partir de escolas de música no Marrocos [7] e de músicas indianas [8]. O estudo das músicas em uma perspectiva histórica fomenta também um melhor entendimento da cultura e gostos de diferentes comunidades.

Vocaloid pode ser definido como um sintetizador vocal desenvolvido originalmente pela empresa *Yamaha Corporation*, sendo um dos primeiros recursos deste tipo para uso da comunidade [9]. Inicialmente o uso deste recurso era limitado para a comunidade oriental, tendo pouca difusão no ocidente, porém isso mudou com o advento das VTubers (*Virtual Youtubers*) que tiveram boa aceitação no ocidente principalmente pela participação em canais como Youtube, Twitch, Tik Tok, entre outros [10]. Tanto o vocaloid quanto a VTuber além de possuírem o recuso de sintetização vocal costumam acompanhar uma representação tridimensional de uma personagem. No caso das VTubers, pode-se citar a Kizuna AI como a primeira e mais famosa [11], e já no caso dos vocaloides destaca-se a Hatsune Miku, que é o estudo de caso deste trabalho. A importância da representação tridimensional de uma personagem está vinculada a uma identidade visual e um apelo ao público [12], o que reflete no sucesso dos casos como apontado por sites como NicoNico [13] e BiliBili [14].

Hatsune Miku, ilustrada pela Figura 1, é a vocaloid mais conhecida, tendo gerado diversos produtos comerciais, desde shows com representação holográfica até jogos e miniaturas. Criada em 2007 pela *Crypton Future Media*, seu surgimento impulsionou o uso dos vocaloides [15]. Sua escolha para o contexto deste trabalho foi consequência da sua disseminação pela comunidade *creative commons* desde 2012 [16], resultando num grande número de recursos de áudio para uso pela comunidade científica.

Figura 1: Hatsune Miku (Crypton Future Media)

Considerando a tendência de crescimento da popularidade de vocaloides, segue a pergunta de pesquisa considerada: é possível utilizar processamento de sinais de áudio e aprendizado de máquina para identificar áudios destes vocaloides?

A extração de informações das músicas deste novo gênero pode contribuir para a pesquisa da evolução histórica da cultura (e.g., quais são os aspectos que levam à maior popularidade de determinada vocaloid?), auxiliar processos de análise de infração de direitos autorais (e.g., uso de vocaloide para ganhos comerciais sem a devida referência), e até fomentar discussões filosóficas e sociais sobre o tema (e.g., qual o impacto das vocaloides sobre as *idols* e outros cantores cujas músicas podem ser similares? As vocaloides contribuem para a padronização de músicas e podem limitar a liberdade artística de seus criadores?).

Este trabalho apresenta um experimento de processamento de áudio vocaloid realizando um estudo de caso da Hatsune Miku. Amostras de áudio humano e vocaloid foram obtidas do domínio público e utilizadas para treinar um classificador *Support Vector Machine* (SVM).

2. Trabalhos Relacionados

Uma busca nas grandes bases de pesquisa foi realizada com o intuito de identificar trabalhos relacionados com o aqui proposto. A *query* de busca utilizada foi "*vocaloid AND audio AND processing*", sem uma delimitação do período de publicação. A Tabela 1 apresenta a quantidade de artigos obtidos por base de pesquisa.

Tabela 1: Quantidade de artigos obtidos por base de pesquisa

Base de Pesquisa	nº de artigos
ACM	12
Elsevier	2
IEEE	1
Springer	17
Total	**33**

Devido ao baixo número de artigos obtidos, não foi necessário realizar um refinamento na *query* de busca. Após a leitura destes artigos em questão, inicialmente observando o resumo, e caso necessário lendo o corpo do texto, identificou-se que a maior parte deles propõe sistemas/soluções com funcionamento análogo ao de um vocaloid, ou que utilizem seus princípios e fundamentos, sem a realização de um estudo de caso envolvendo seu processamento de áudio, como é proposto neste artigo, ou então apenas uma menção à sua existência. Desta forma, este trabalho apresenta caráter de ineditismo, conforme suportado pelos resultados obtidos nesta pesquisa na literatura.

3. Método

O método aplicado, ilustrado pela Figura 2, consiste nas seguintes etapas:

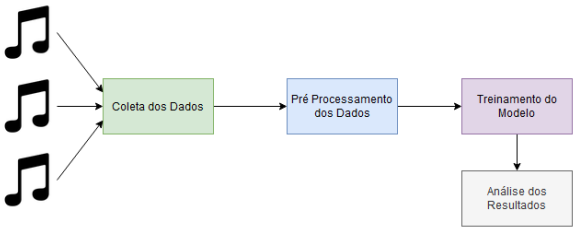

Figura 2: Método aplicado

1. **Coleta de Dados**: Obtenção de músicas sob licença *Creative Commons* instrumentais e de vocaloides. Neste passo destaca-se a dificuldade de obter músicas de cantores humanos de forma gratuita ou de forma ágil. O total de arquivos de humanos também é mais limitado quando comparado aos arquivos disponíveis das vocaloides. Os arquivos instrumentais foram obtidos em `https://www.bensound.com/royalty-free-music/`, enquanto os arquivos das vocaloides foram obtidos em `http://mikudb.moe/`.
2. **Pré Processamento dos Dados**: Conversão da música em formato *mp3* para o formato *wav* compatível com a biblioteca *pyAudioAnalysis* utilizada. Foi utilizado o *FFmpeg* [17] por meio de chamada no notebook Python no ambiente de execução local Jupyter Notebook. Houve também o processamento dos áudios usando a biblioteca Python *pyAudioAnalysis* [18] para obter os parâmetros chroma. Por simplicidade, foram considerados apenas os 39 primeiros parâmetros de cada um dos 12 parâmetros chroma nos experimentos realizados.
3. **Treinamento dos Modelos**: Separação em bases de treinamento e teste, com 70% da base para treinamento e 30% para teste. Em seguida, seleção dos parâmetros chroma usando o método *Lasso* da biblioteca *sklearn* [19], e também a seleção de hiperparâmetros para posterior treinamento do classificador SVM usando o método *GridSearchCV* da mesma biblioteca *sklearn*. Por fim, treinamento do classificador SVM usando os hiperparâmetros obtidos.
4. **Análise dos Resultados**: considerando cenários previstos nos experimentos, analisar os resultados obtidos com o classificador com métricas acurácia, precisão, *f1 score*, além da curva *Receiver Operating Characteristic* (ROC). O uso da curva ROC está relacionado com o cálculo da área embaixo da curva (AUC), que apresenta um valor entre 0 e 1. Quanto mais próximo o valor obtido está do valor máximo, melhor o desempenho do classificador.

É importante destacar que a escolha do *Support Vector Machine* (SVM) foi motivado pelo seu bom desempenho na identificação de áudio sintético [20], e sua maior simplicidade quando comparado a técnicas com arquiteturas de redes neurais. O *Lasso* é um modelo linear que estima coeficientes esparsos, e seu uso foi motivado pela possibilidade de reduzir o número de parâmetros sob os quais uma solução é dependente [21], o que pode ajudar a evitar o *overfitting* do classificador.

Dois experimentos foram realizados. O primeiro consistiu em treinar um classificador para diferenciar músicas instrumentais de gêneros diversos das músicas da vocaloid Hatsune Miku. O segundo consistiu em treinar um classificador para diferenciar músicas de outras vocaloides (e.g., Kagamine Rin) das músicas da vocaloid Hatsune Miku.

No total foram utilizadas 60 músicas, divididas em três classes. A classe 1 possui 20 músicas de autoria da vocaloid Hatsune Miku, enquanto a classe 0 possui 20 músicas instrumentais de diversos gêneros. Por fim, a classe 0' possui 20 músicas de outras vocaloides. Desta forma, as músicas correspondentes as classes 1 e 0 foram utilizadas no primeiro experimento, enquanto as músicas correspondentes as classes 1 e 0' foram usadas no segundo experimento.

4. Resultados

Ambos os experimentos foram realizados utilizando o ambiente de execução local do *Jupyter Notebook*, em um computador com 8GB de RAM, processador Intel Core i3 em sistema operacional Windows 10.

A Figura 3 apresenta um chromagrama de uma das músicas de Hatsune Miku obtido após processamento do áudio por meio da biblioteca *pyAudioAnalysis*. O chromagrama é uma representação visual dos parâmetros chroma, apresentando graficamente no intervalo de tempo do áudio a intensidade de cada um dos 12 semitons de uma oitava {C, C#, D,..., B}.

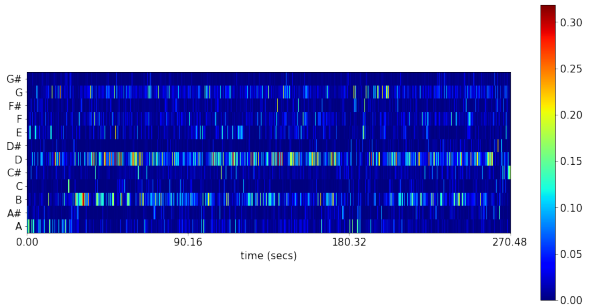

Figura 3: Chromagrama de uma das músicas da vocaloid Hatsune Miku

O primeiro experimento é ilustrado na Figura 4. Arquivos de músicas instrumentais de gêneros diversos e músicas da vocaloid Hatsune Miku são processadas pela biblioteca *pyAudioAnalysis*. A partir do total de 468 parâmetros chroma, o método *Lasso* seleciona 18 parâmetros. Os hiperparâmetros obtidos com o *GridSearchCV* usando como métrica o *f1 score* e validação cruzada *5-fold* são $C = 5$, *gamma* = *scale* e *kernel* = *rbf*.

Foram coletadas métricas e traçada a curva ROC correspondente ao desempenho do classificador. Obteve-se uma acurácia de 81,67%, precisão de 80%, *recall* de 67%, e o *f1 score* de 62%. A Figura 5 apresenta a curva ROC, que mostra que o valor da área sob a curva foi de 0.83.

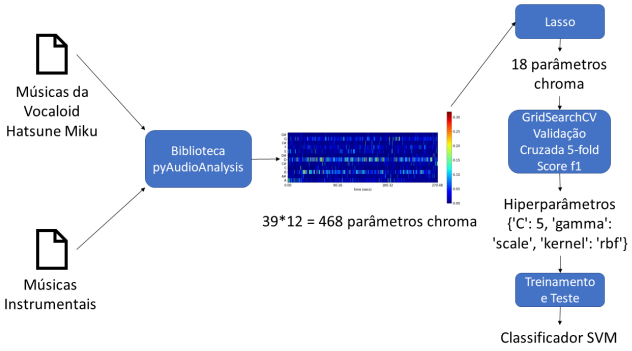

Figura 4: Experimento 1 de detecção de música da Hatsune Miku em comparação com músicas de outros gêneros

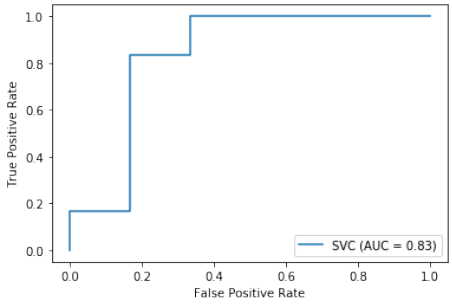

Figura 5: Curva ROC do Experimento 1

O segundo experimento é ilustrado na Figura 6. Arquivos de músicas de outros vocaloides e músicas da vocaloid Hatsune Miku são processadas pela biblioteca *pyAudioAnalysis*. A partir do total de 468 parâmetros chroma, o método *Lasso* seleciona 14 parâmetros. Os hiperparâmetros obtidos com o *GridSearchCV* são $C = 50$ e *kernel* = *linear*.

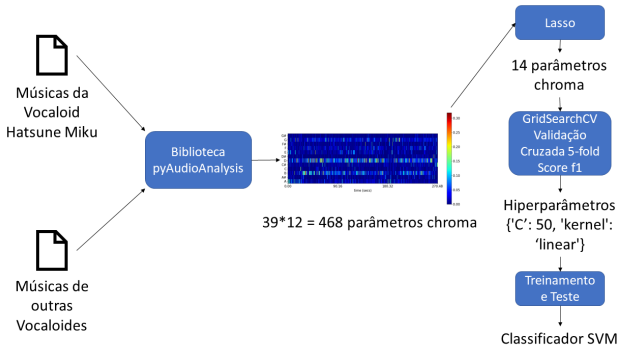

Figura 6: Experimento 2 de detecção de música da Hatsune Miku em comparação com músicas de outras vocaloides

Novamente foram coletadas métricas e traçada a curva ROC correspondente ao desempenho do classificador. Obteve-se uma acurácia de 80%, precisão de 88%, *recall* de 83%, e o *f1 score* de 83%. A Figura 7 apresenta a curva ROC, que mostra que o valor da área sob a curva foi de 0.86.

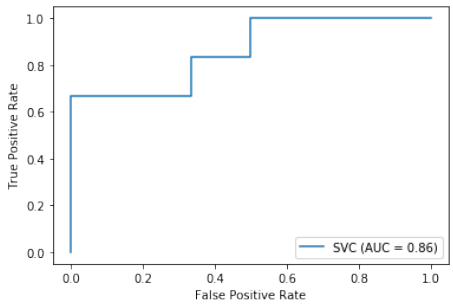

Figura 7: Curva ROC do Experimento 2

5. Conclusões

Este trabalho provou a viabilidade de obter um classificador que diferencia áudios da vocaloid Hatsune Miku de músicas instrumentais e de músicas de outras vocaloides, constituindo uma iniciativa na área de extração de informação de músicas deste novo gênero.

Como trabalhos futuros, destaca-se os testes e comparação com outros modelos de aprendizado de máquina, escolha de outros parâmetros dos áudios como o *Mel-Frequency Cepstral Coefficients* (MFCC), ou seleção de parâmetros a partir de técnicas de *deep learning* em conjunto com classificadores SVM [22]. Também é possível realizar testes com um conjunto maior de amostras da música, de forma a investigar o impacto no desempenho dos classificadores. Outra possibilidade é integrar vocaloides a um método para construção automática de bases de dados abertos de músicas para fomentar mais pesquisas na área.

Agradecimentos

Os autores agradecem a *Crypton Future Media* desenvolvedora da Hatsune Miku e toda a comunidade responsável por criar conteúdo e disponibilizá-lo sob licença *creative commons* viabilizando este trabalho.

Referências

[1] Vladimir Kulyukin, Sarbajit Mukherjee, and Prakhar Amlathe. Toward audio beehive monitoring: Deep learning vs. standard machine learning in classifying beehive audio samples. *Applied Sciences*, 8(9):1573, 2018.

[2] Hareesh Bahuleyan. Music genre classification using machine learning techniques. *arXiv preprint arXiv:1804.01149*, 2018.

[3] Gerson Albuquerque Silva. Proposta de construção de um banco de dados de amostras de fala para uso forense em um arcabouço bayesiano. *Revista Brasileira de Criminalística*, 5(1):35–45, 2016.

[4] Garima Sharma, Kartikeyan Umapathy, and Sridhar Krishnan. Trends in audio signal feature extraction methods. *Applied Acoustics*, 158:107020, 2020.

[5] Roberto Font, Juan M Espín, and María José Cano. Experimental analysis of features for replay attack detection-results on the asvspoof 2017 challenge. In *Interspeech*, pages 7–11, 2017.

[6] Xavier Serra. Creating research corpora for the computational study of music: the case of the compmusic project. In *Audio engineering society conference: 53rd international conference: Semantic audio*. Audio Engineering Society, 2014.

[7] Mohamed Sordo, Amin Chaachoo, and Xavier Serra. Creating corpora for computational research in arab-andalusian music. In *Proceedings of the 1st International Workshop on Digital Libraries for Musicology*, pages 1–3, 2014.

[8] Ajay Srinivasamurthy, Gopala Krishna Koduri, Sankalp Gulati, Vignesh Ishwar, and Xavier Serra. Corpora for music information research in indian art music. In *Georgaki A, Kouroupetroglou G, eds. Proceedings of the 2014 International Computer Music Conference, ICMC/SMC; 2014 Sept 14-20; Athens, Greece.[Michigan]: Michigan Publishing; 2014.* Michigan Publishing, 2014.

[9] Hideki Kenmochi and Hayato Ohshita. Vocaloid-commercial singing synthesizer based on sample concatenation. In *Eighth Annual Conference of the International Speech Communication Association*, 2007.

[10] Zhicong Lu, Chenxinran Shen, Jiannan Li, Hong Shen, and Daniel Wigdor. More kawaii than a real-person live streamer: Understanding how the otaku community engages with and perceives virtual youtubers. CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.

[11] Xin Zhou. Virtual youtuber kizuna ai. *ies, L u n d U*, page 205.

[12] RAISING THEIR. The hatsune miku phenomenon: More than a virtual j-pop diva. *The Journal of Popular Culture*, 49(5), 2016.

[13] Niconico. Niconico. `https://www.nicovideo.jp/`, 2021. Acesso: 05/06/2021.

[14] Bilibili. Bilibili. `https://www.bilibili.com/`, 2021. Acesso: 05/06/2021.

[15] Hideki Kenmochi. Vocaloid and hatsune miku phenomenon in japan. In *Interdisciplinary Workshop on Singing Voice*, 2010.

[16] Chiaki. Hatsune miku joins the cc community. `https://creativecommons.org/2012/12/14/hatsune-miku-joins-the-cc-community/`, 2021. Acesso: 05/06/2021.

[17] FFmpeg-developers. Ffmpeg. `https://www.ffmpeg.org/`, 2021. Acesso: 05/06/2021.

[18] Theodoros Giannakopoulos. pyaudioanalysis: An open-source python library for audio signal analysis. *PloS one*, 10(12):e0144610, 2015.

[19] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[20] Pavel Korshunov and Sébastien Marcel. A cross-database study of voice presentation attack detection. In *Handbook of Biometric Anti-Spoofing*, pages 363–389. Springer, 2019.

[21] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

[22] Ivan Himawan, Srikanth Madikeri, Petr Motlicek, Milos Cernak, Sridha Sridharan, and Clinton Fookes. Voice presentation attack detection using convolutional neural networks. In *Handbook of Biometric Anti-Spoofing*, pages 391–415. Springer, 2019.