A Deep Multimodal Approach for Cold-start Music Recommendation

Sergio Oramas
Music Technology Group
Universitat Pompeu Fabra
sergio.oramas@upf.edu

Oriol Nieto
Pandora Media Inc.
Oakland, CA
onieto@pandora.com

Mohamed Sordo
Pandora Media Inc.
Oakland, CA
msordo@pandora.com

Xavier Serra
Music Technology Group
Universitat Pompeu Fabra
xavier.serra@upf.edu

ABSTRACT

An increasing amount of digital music is being published daily. Music streaming services often ingest all available music, but this poses a challenge: how to recommend new artists for which prior knowledge is scarce? In this work we aim to address this so-called cold-start problem by combining text and audio information with user feedback data using deep network architectures. Our method is divided into three steps. First, artist embeddings are learned from biographies by combining semantics, text features, and aggregated usage data. Second, track embeddings are learned from the audio signal and available feedback data. Finally, artist and track embeddings are combined in a multimodal network. Results suggest that both splitting the recommendation problem between feature levels (i.e., artist metadata and audio track), and merging feature embeddings in a multimodal approach improve the accuracy of the recommendations.

KEYWORDS

recommender systems, deep learning, multimodal, music, semantics

1 INTRODUCTION

It is common for online music streaming services nowadays to offer ever-growing catalogs with dozens of millions of music tracks. Since manually managing these large libraries is not feasible due to size constraints, automatic exploration and exploitation of large-scale music collections has been an active area of research in the recent years [6]. While several existing algorithmic techniques are able to produce successful recommendations for popular content [16], the exploration of new or *undiscovered* artists (i.e., the long tail [7]) remains a major challenge that we aim to address.

Recommender systems can be broadly classified into collaborative filtering (CF), content-based, and hybrid methods. CF methods [16] use the item-user feedback matrix and predictions are based on the similarity of user or item profiles. Matrix factorization techniques are currently CF state-of-the-art [16]. CF methods suffer from the cold-start problem, as new items do not have feedback information [28]. Content-based methods [20] rely only on item features, and recommendations are based on similarity between such features. Finally, hybrid methods [5] try to combine both item content and item-user feedback.

Social tags have been extensively used as a source of artist content features to recommend music [15]. However, these tags are usually collectively annotated, which often introduce an artist popularity bias [32]. Artist biographies and press releases, on the other hand, do not necessarily require a collaborative effort, as they may be produced by artists themselves. However, they have seldom been exploited for music recommendation [26]. Part of this work focuses on learning artist features from these biographies. Furthermore, we also make use of audio signals, since these are generally always available and have shown to be helpful when recommending music in the long tail [33].

To combine these data, we first separate the problem of music recommendation into artist and song levels. Artist feature embeddings are learned from artist metadata in an artist recommendation scenario. Track feature embeddings are learned from audio signals in a song recommendation scenario. In both cases, a hybrid recommendation approach is used based on learning attribute-to-feature mappings [11]. This method addresses the lack of feedback for uncommon items in two steps: (1) factorizing the collaborative matrix, and (2) learning a mapping between item content features and item latent factors [1, 33]. Lastly, both feature embeddings are combined in a novel multimodal deep network to predict song recommendations in the long tail. We show how dividing the problem into artists and songs, and combining text and audio in a multimodal approach yields improved recommendations.

For the sake of reproducibility, source code and data splits used in the experiments have been released[1]. Our main contributions in this work are summarized as follows:

- Method to enrich artist biographies with semantic information leveraging an external Knowledge Base.
- Dividing the problem of music recommendation into artist and song levels to obtain better performance.
- A multimodal deep learning pipeline for music recommendation that combines audio with text and yields improved results.
- The release of an extended version of the Million Song Dataset with artist biographies and tags.

[1] https://github.com/sergiooramas/tartarus

2 RECOMMENDATION APPROACH

To produce cold-start music recommendations, we propose the following framework. Given the set of artist features A_s of a song s, and the set of track features T_s of s, the complete feature set of s is defined as the aggregation of its artist and track features $F_s = A_s \cup T_s$. Since songs from the same artist share the same set A_s, two problems may arise: (1) the learning process involved with the recommender system might not be properly optimized, and (2) if different songs from the same artist appear in multiple sets (e.g., train and test), a problem of overfitting may arise [10]. To address (1), we divide the problem into three phases (see Figure 1). First, we aggregate the collaborative information of all songs of the same artist, and learn an artist feature embedding A'_s from A_s in an artist recommendation scenario. Second, we learn a track feature embedding T'_s from T_s in a pure audio-based recommendation scenario. Third, we combine both feature embeddings A'_s and T'_s in a multimodal network and compute song recommendations. To approach (2), we use non-overlapping artists across the train, validation and test sets.

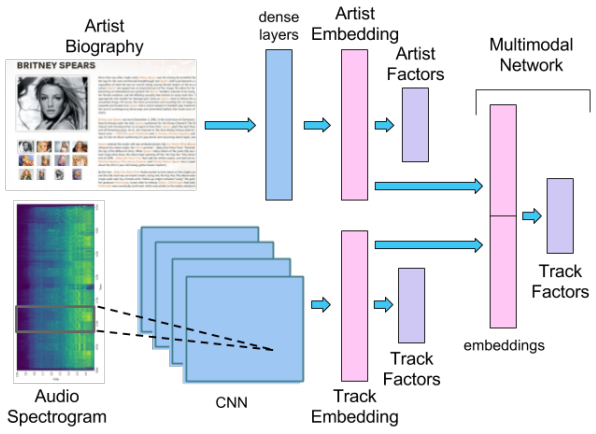

Figure 1: Model architecture.

More specifically, let M be the matrix of implicit feedback, where m_{us} is the number of play counts for user u on song s. M is split into M_{train}, M_{val} and M_{test}, for train, validation and test, respectively, where no artist is shared across sets. Factorizing M_{train} using weighted matrix factorization (WMF) [12] yields I_k and U_k, the sets of songs and users latent factors, respectively, of dimension k. We set $k = 200$, and apply the alternating least squares (ALS) optimization method.

To learn the artist embeddings, we obtain the matrix of artist implicit feedback R from M, being $R_{ua} = \sum_s m_{us}$ for all songs s from the same artist a. This matrix is split into train, validation and test sets following the same mutual exclusion restriction. Latent factors of artists and users are later obtained via WMF. Lastly, a deep neural network is trained on the prediction of artist latent factors from artist content features A. On the other hand, the song latent factors are predicted with a deep convolutional network, using I_k as training data and the track features T as input (similar to [33]).

Once the artist and track models are trained and optimized, we gather the activations from the penultimate layer of each network for all the sets. These activations constitute what we call the artist and track feature embeddings A'_s and T'_s, which are in turn used as input to a third network. This final multimodal network is trained on the prediction of song latent factors I_k from $S'_s = A'_s \cup T'_s$. Finally, the list of item recommendations for user u is obtained by ranking the results of computing the dot product between the user latent factor $f_u \in U_k$ and the set of item factors.

The different architectures used in each one of the three neural networks involved in the approach are described in Sections 3 ,4, and 5, respectively. Nevertheless, all networks have a final fully connected layer of 200 units[2] with linear activation and l2-normalization. In addition, mini batches of 32 items are randomly sampled from the training data to compute the gradient in all networks, and Adam [14] is the optimizer used to train the models, with the default suggested learning parameters. Given that the output of the architectures are l2-normalized, we use cosine proximity as the loss function, as in [8].

3 ARTIST TEXT EMBEDDINGS

In this section we describe two different, competing approaches to exploit artist texts in a deep learning process.

3.1 Semantic enrichment

We propose a method for enriching artist biographies by associating text fragments with relevant entities defined in online semantic datasets, and then gathering relevant information about these entities from a Knowledge Base (KB). For this purpose, we adopted Babelfy, a state-of-the-art tool for Entity Linking (EL) [21], a task to associate, for a given textual fragment candidate, the most suitable entry in a reference KB. Babelfy maps words from a given text to entities in the BabelNet Knowledge Base [22], which has a direct mapping to DBpeadia[3], a structured version of Wikipedia[4]. We use semantic information about the identified entities coming from DBpedia to enrich the biographies. DBpedia resources are generally classified using the DBpedia Ontology, which is a shallow, cross-domain ontology based on the most common infoboxes of Wikipedia.

EL systems are useful for music recommendation [26]. However, they are not optimized for the music domain, and are prone to errors [25]. The application of a filtering process over the set of identified entities based on their classification within the DBpedia Ontology, has demonstrated its utility to improve music retrieval tasks, such as artist similarity [27]. Therefore, we only keep entities of classes related to the music domain such as MusicalArtist, Band, MusicGenre, MusicalWork, RecordLabel, Instrument, Engineer and Place. Then, we query DBpedia to get all the available information about the filtered entities. From the information gathered, we keep some specific properties for every kind of entity, such as homeTown, instrument, genre or associatedBand for MusicalArtists, writer, producer or recordedIn for MusicalWorks, stylisticOrigin or

[2] to match the dimensions of the factors to be predicted.
[3] http://dbpedia.org
[4] http://wikipedia.org

instrument for MusicGenres, and so on[5]. In addition, we also kept all the Wikipedia categories associated to each entity. In Wikipedia, categories are used to organize resources, and they help users to group articles of the same subject.

To build the enriched biographies we proceed as follows: First, Babelfy is applied over the biography texts. Second, information is gathered from DBpedia for the entities of the selected classes. Finally, the collected data is added at the end of the biography text separated by spaces. Then a vector space model (VSM) is applied to the set of enriched biographies, and tf-idf weighting [34] is applied. We limited the vocabulary size to 10,000 terms for the VSM, as this vocabulary size provides a good trade of between performance and number of parameters required for training. Note that either words, entities, dates or categories may be part of this vocabulary. From this data representation, a feedforward network with two dense layers of 2048 neurons each is trained to predict the artist latent factors.

3.2 Word embeddings

Much of the work with deep learning in Natural Language Processing has involved the learning of word vector representations [3, 19], and their further composition [9]. Word embeddings aim to represent words as low-dimensional dense vectors. They have demonstrated to greatly benefit NLP tasks, such as word similarity, sentiment analysis or parsing [24].

The use of convolutional neural networks (CNN) over pre-trained word vectors has become state-of-the-art in sentence classification [13]. We re-adapt the architecture proposed in [13] for sentence classification to learn artist latent factors from artist biographies. This consists in an embedding layer, followed by a one dimensional convolutional layer with multiple filter widths, a max-over-time pooling layer, a dense hidden layer and the output layer. We employ the same architecture and parameters, changing only the output layer and the loss function. We initialize the input embedding layer of the network with word2vec word embeddings pre-trained on the Google News dataset, and also with word embeddings trained in our own corpus of biographies.

4 TRACK AUDIO EMBEDDINGS

It is common in the field of music informatics to make use of CNNs to learn higher-level features from spectrograms. These representations are typically contained in $\mathbb{R}^{\mathcal{F} \times N}$ matrices with $\mathcal{F}$ frequency bins and N time frames. In this work we compute 96 frequency bin, log-compressed constant-Q transforms (CQT) [29] for all the tracks in our dataset using `librosa` [18] with the following parameters: audio sampling rate at 22050 Hz, hop length of 1024 samples, Hann analysis window, and 12 bins per octave. Following a similar approach to [33], we address the variability of the length N across songs by sampling one 15-seconds long *patch* from each track, resulting in the fixed-size input to the CNN.

The deep model trained with these data is defined as follows: the CQT patches are fed to four convolutional layers with rectified linear units (ReLU) as activations. The four convolutions have the following number of filters, from first to last: 256, 512, 1024, and 1024. The convolutions are only applied to the time axis, leaving the frequencies fixed since the absolute and relative bin placement is important when aiming to capture particular sounds (as opposed to the irrelevance of *where* in time a certain sonic event occurs). Maxpooling of 4 units across the time axis is applied after each of the first three ReLUs, and 50% dropout is applied to all layers. The flattened output of the last layer has 4096 units, which becomes the vector embedding to later use in the multimodal approach described next.

5 MULITMODAL FUSION

There are several approaches in the literature for multimodal feature learning [23, 31], and late fusion of multimodal feature vectors [2, 30]. In this work, audio and text feature vectors are learned separately and then combined via late fusion in a multimodal network (see Figure 1). Given the different nature of the artist and track embeddings, a normalization step is necessary. Given a set of feature vectors, $l2$-norm is applied on each of them. They are then concatenated into a single feature vector, which is fed to a feed forward neural network (a simple Multi Layer Perceptron, MLP), where the input layer is directly connected to the output layer. Regularization is obtained by applying dropout with an empirically selected factor of 70% after the input layer.

6 EXPERIMENTS

6.1 Dataset

The Million Song Dataset (MSD) [17] is a collection of metadata and precomputed audio features for 1 million songs. Along with this dataset, the Echo Nest Taste Profile Subset [4] provides play counts of 1 million users on more than 380,000 songs from the MSD. Starting from this subset, we gather the artist biography and the social tags from Last.fm for all the artists that have at least one song in the dataset. When there are several artists with the same name, they are stored in the same page of Last.fm, which makes the biography and social tags ambiguous. We automatically removed all ambiguous artists by applying text processing on the biographies. The song features provided with the MSD are not generally suitable for deep learning, so we instead use audio previews between 7 and 30 seconds retrieved from `7digital.com`. After removing ambiguous artists and missing tracks, the final dataset consists of 328,821 tracks from 24,043 artists. Each track has at least 15 seconds of audio, each biography is at least 50 characters long, and each artist has at least 1 tag associated with it. All artist metadata, implicit feedback matrices, and splits are released as a new dataset called the MSD-A[6].

6.2 Artist Recommendation

To investigate to what extent the different feature sets, data models and architectures influence the quality of the deep artist features, we evaluate the different approaches in an artist recommendation scenario. Given the matrix of implicit feedback R, and the set of artist and user factors obtained through matrix factorization (see Section 2), we predict the artist factors for the test set, and use them to compute a ranked list of recommended artists for every

[5]The complete list of classes and properties is available at http://mtg.upf.edu/download/datasets/msd-a

[6]http://mtg.upf.edu/download/datasets/msd-a

Table 1: Artist Recommendation Results

Aproach	Input	Data model	Arch	MAP
A-TEXT	Bio	VSM	FF	0.0161
A-SEM	**Sem Bio**	**VSM**	**FF**	**0.0201**
A-W2V-GOO	Bio	w2v-pretrain	CNN	0.0119
A-W2V	Bio	w2v-trained	CNN	0.0145
A-TAGS	Tags	VSM	FF	0.0314
TAGS-ITEMKNN	Tags	-	itemKnn	0.0161
TEXT-RF	Bio	VSM	RF	0.0089
RANDOM	-	-	-	0.0014
UPPER-BOUND	-	-	-	0.5528

Mean average precision (MAP) at 500 for the predictions of artist recommendations in 1M users. VSM refers to Vector Space Model, FF to Feedforward, RF to Random Forest, CNN to Convolutional Neural Network, and itemKnn to itemAttributeKnn approach. Bio refers to biography texts and Sem Bio to semantically enriched texts.

Table 2: Song Recommendation Results

Approach	Artist Input	Track Input	Arch	MAP
AUDIO	-	audio spec	CNN	0.0015
SEM-VSM	Sem Bio	-	FF	0.0032
SEM-EMB	A-SEM	-	FF	0.0034
MM-LF	**A-SEM**	**AUDIO emb**	**MLP**	**0.0036**
MM	Sem Bio	audio spec	CNN	0.0014
TAGS-VSM	Tags	-	FF	0.0043
TAGS-EMB	A-TAGS	-	FF	0.0049
RANDOM	rnd emb	-	FF	0.0002
UPPER-BOUND	-	-	-	0.1649

Mean average precision (MAP) at 500 for the predictions of song recommendations in 1M users. AUDIO emb refers to the track embedding of AUDIO approach, SEM to artist embedding of SEM approach, TAGS to artist embedding of TAGS approach, and spec to spectrogram.

user. We use mean average precision (MAP) with a cut-off at 500 recommendations per user.

We compare four different approaches using the biography texts as input. (1) a pure text-based approach using a VSM and a feedforward network A-TEXT. (2) similar to (1) but with a semantically enriched version of the texts A-SEM (cf. Section 3.1). (3) An approach based on word embeddings initialized with vectors trained on Google News and a CNN A-W2V-GOO (cf. Section 3.2). (4) Similar to (3) but inizializing the embeddings with word vectors previously trained on the corpus of biographies A-W2V. To properly frame the results, we compute two baselines and one competitor approach. The TAGS baseline approach uses artist social tags as input features, and TEXT-RF uses biography texts as input, but Random Forest Regression for the learning instead of a deep neural network. The former baseline is added to compare the potential of biography texts with respect to curated metadata, whilst the latter was added to study to which extent the deep network improves the results over other learning methods typically used in natural language processing. There are few recommendation approaches able to deal with an extreme cold-start scenario like ours. Therefore, we select ItemAttributeKnn [11] as the competitor approach (TAGS-ITEMKNN), using artist social tags as attribute data and computed using the MyMediaLite library[7]. We also show the scores achieved when the latent factor vectors are randomized (RANDOM), and when they are learned from feedback data using matrix factorization (UPPER-BOUND).

Results reported in Table 1 show that the semantic enrichment of the biographies A-SEM outperforms the pure text approach A-TEXT. As expected, the use of tags improves the results over the use of text. However, the addition of semantic features reduces the gap in performance between the use of tags and unstructured text. Moreover, the difference between A-TEXT and TEXT-RF shows that the use of deep learning with respect to random forest improves the results. We also note that a VSM model with a feedforward network outperforms the use of word embeddings with convolutions. Although, according to the literature, this latter approach has demonstrated its utility for simple tasks like binary classification with short texts, our task puts forward two challenges for this architecture: the greater length of the input texts, and the higher dimensionality of the output. Although we have shown that initializing the embedding layer with word vectors trained on the corpus itself (A-W2V) outperforms the use of Google News pre-trained vectors (A-W2V-GOO), further work is necessary to properly optimize a convolutional architecture for this task. Finally, we observe that our approach A-TAGS outperforms the competitor approach TAGS-ITEMKNN using the same item attributes.

Once the network is trained, we predict the activations of the penultimate layer for the entire dataset of artists. Thus, we obtain a vector embedding of 2048 dimensions, which represents the artist deep features A'. From the evaluated approaches, we compute the artist embedding from the A-SEM and A-TAGS approaches.

6.3 Song Recommendation

In this experiment, audio embeddings are obtained after training the convolutional network (see Section 4) with 260k patches of 15 seconds, corresponding to the 80% of the tracks described in Section 6.1. Patches are divided into training (80%), validation (10%) and test (10%) sets. Results reported in Table 2 are computed over the remaining 20% of tracks. As opposed to [33], no artist appears in more than one subset to avoid overfitting. Finally, multimodal approaches are computed on the same sets.

In our experiments, we want to measure the impact of the artist embeddings in the song recommendation problem, and also the potential of the multimodal approach. We experimented with two approaches, SEM-EMB and TAGS-EMB, that exploit the feature embeddings learned from the artists features (see Section 6.2), either based on biography texts (A-SEM) or artists tags (A-TAGS). To measure the potential of the artist embeddings, we also computed two approaches based on the original artist features (SEM-VSM for semantic text features and TAGS-VSM for tag features). Results on Table 2 show that SEM-EMB and TAGS-EMB outperform SEM-VSM

[7] http://www.mymedialite.net/

and TAGS-VSM, suggesting that artist embeddings outperform artist features.

An approach based on the audio spectrograms was computed AUDIO. From this latter approach, audio embeddings where obtained (AUDIO emb) and combined with A-SEM in a multimodal late fusion approach MM-LF (cf. Section 5). We compared this network with a multimodal approach trained directly on the original features (semantically enriched text and audio spectrograms). Results on the combination of artist and track features show that the late fusion of artist and track embeddings MM-LF outperforms the simultaneous training of artist and track features MM. Finally, we observe that the multimodal approach that combines text and audio features with late fusion MM-LF improves the results of pure text SEM-EMB or pure audio AUDIO approaches. All the differences between the approaches are statistically significant ($p < 0.01$) according to the paired t-test.

We also compared the results with an upper-bound approach obtained from the feedback data and an approach trained with random vector embeddings. Although results are in general far from the upper-bound, the comparative analysis of the proposed approaches gives some insights of the behavior of different feature representations and modalities in the cold-start recommendation problem.

7 CONCLUSIONS

In this work, a multimodal approach for song recommendation has been presented. The approach is divided into three steps. (1) Artist feature embeddings are learned from text and semantic features in an artist recommendation scenario using a deep network architecture. (2) Track feature embeddings are learned from the audio spectrograms using convolutional neural networks. (3) Embeddings are combined in a multimodal network.

Results show that splitting the problem of song recommendation at artist and song levels improves the quality of recommendations. Learning artist feature embeddings separately benefits from the aggregation of the information about the different songs of the same artist, yielding more robust artist features. Related to this, an approach for the semantic enrichment of artist metadata has been proposed, leading to a significant improvement in the results. In addition, we have shown the potential of exploiting artist biographies in music recommendation. Moreover, the deep learning architectures of this work have demonstrated their capacity to improve upon other learning models under the music recommendation framework. Finally, we have shown how a multimodal approach, based on the late fusion of track and artist feature embeddings outperforms pure text or audio approaches.

ACKNOWLEDGMENTS

This work was partially funded by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502).

REFERENCES

[1] Trapit Bansal, David Belanger, and Andrew McCallum. 2016. Ask the GRU: Multi-task Learning for Deep Text Recommendations. *RecSys* (2016), 107–114. DOI: http://dx.doi.org/10.1145/2959100.2959180

[2] M. Rouvier Bechet, S. Delecraz, B. Favre, M. Bendris, and F. 2015. Multimodal embedding fusion for robust speaker role recognition in video broadcast. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)* (2015).

[3] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A Neural Probabilistic Language Model. *The Journal of Machine Learning Research* 3 (2003), 1137–1155. DOI: http://dx.doi.org/10.1162/153244303322533223

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

[5] Robin Burke. 2002. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* 12, 4 (2002), 331–370.

[6] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. 2008. Content-Based Music Information Retrieval: Current Directions and Future Challenges. *Proc. IEEE* 96, 4 (April 2008), 668–696. DOI: http://dx.doi.org/10.1109/JPROC.2008.916370

[7] Òscar Celma. 2010. *The Long Tail in Recommender Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 87–107. DOI: http://dx.doi.org/10.1007/978-3-642-13287-2_4

[8] François Chollet. 2016. Information-theoretical label embeddings for large-scale image classification. *CoRR* (2016), 1–10. http://arxiv.org/abs/1607.05691

[9] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research* 12 (2011), 2493–2537. DOI: http://dx.doi.org/10.1.1.231.4614

[10] Arthur Flexer. 2007. A Closer Look on Artist Filters for Musical Genre Classification. In *Proceedings of the 8th International Society for Music Information Retrieval Conference*.

[11] Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, Steffen Rendle, and Lars Schmidt-Thieme. 2010. Learning Attribute-to-Feature Mappings for Cold-Start Recommendations. In *Proceedings of the 2010 IEEE International Conference on Data Mining (ICDM '10)*. IEEE Computer Society, Washington, DC, USA, 176–185.

[12] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM '08)*. 263–272.

[13] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)* (2014), 1746–1751. DOI: http://dx.doi.org/10.1109/LSP.2014.2325781

[14] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). http://arxiv.org/abs/1412.6980

[15] Peter Knees and Markus Schedl. 2013. A survey of music similarity and recommendation from music context data. *ACM Transactions on Multimedia Computing, Communications, and Applications* 10, 1 (2013), 1–21. DOI: http://dx.doi.org/10.1145/2542205.2542206

[16] Y. Koren, R. Bell, and C. Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009), 42–49. DOI: http://dx.doi.org/10.1109/MC.2009.263

[17] Brian McFee, Thierry Bertin-Mahieux, Daniel P.W. Ellis, and Gert R.G. Lanckriet. 2012. The million song dataset challenge. *Proceedings of the 21st international conference companion on World Wide Web - WWW '12 Companion* (2012), 909. DOI: http://dx.doi.org/10.1145/2187980.2188222

[18] Brian Mcfee, Colin Raffel, Dawen Liang, Daniel P W Ellis, Matt Mcvicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and Music Signal Analysis in Python. *PROC. OF THE 14th PYTHON IN SCIENCE CONF* Scipy (2015), 1–7.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. *Nips* (2013), 1–9. DOI: http://dx.doi.org/10.1162/jmlr.2003.3.4-5.951

[20] Raymond J Mooney and Loriene Roy. 1999. Content-Based Book Recommending. *Proceedings of the SIGIR-99 Workshop on Recommender Systems: Algorithms and Evaluation* August (1999).

[21] Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: A Unified Approach. *Transactions of the Association for Computational Linguistics* 2 (2014), 231–244. https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/291

[22] Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193 (2012), 217–250. DOI: http://dx.doi.org/10.1016/j.artint.2012.07.001

[23] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. 2011. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. 689–696.

[24] Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2016. Neural-based Noise Filtering from Word Embeddings. *Coling-2016* (2016), 2699–2707. http://arxiv.org/abs/1610.01874

[25] Sergio Oramas, Luis Espinosa-Anke, Mohamed Sordo, Horacio Saggion, and Xavier Serra. 2016. Information extraction for knowledge base construction in the music domain. *Data and Knowledge Engineering* 106 (2016), 70–83. DOI:

http://dx.doi.org/10.1016/j.datak.2016.06.001

[26] Sergio Oramas, Vito Claudio Ostuni, Tommaso Di Noia, Xavier Serra, and Eugenio Di Sciascio. 2015. Sound and Music Recommendation with Knowledge Graphs. *ACM Trans. Intell. Syst. Technol.* 9, 4 (2015), 1–21. DOI: http://dx.doi.org/10.1145/2926718

[27] Sergio Oramas, Mohamed Sordo, Luis Espinosa-Anke, and Xavier Serra. 2015. A Semantic-based Approach for Artist Similarity. *Proceedings of 16th the International Society for Music Information Retrieval Conference* October (2015), 100–106.

[28] Martin Saveski and a Mantrach. 2014. Item cold-start recommendations: learning local collective embeddings. *RecSys '14 Proceedings of the 8th ACM Conference on Recommender systems* (2014), 89–96. DOI: http://dx.doi.org/10.1145/2645710.2645751

[29] Christian Schörkhuber and Anssi Klapuri. 2010. Constant-Q transform toolbox for music processing. *7th Sound and Music Computing Conference* JANUARY (2010), 3–64.

[30] Olga Slizovskaia, Emilia Gómez, and Gloria Haro. 2017. Musical instrument recognition in user-generated videos using a multimodal convolutional neural network architecture. In *ICMR '17: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*. Bucharest, Romania.

[31] Nitish Srivastava and Ruslan Salakhutdinov. 2012. Learning representations for multimodal data with deep belief nets. In *International conference on machine learning workshop*.

[32] Douglas Turnbull, Luke Barrington, and Gert Lanckriet. 2008. Five approaches to collecting tags for music. *Proceedings of 9th the International Society for Music Information Retrieval Conference* (2008), 225–230. http://books.google.com/books?hl=en

[33] Aaron van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. *Electronics and Information Systems department (ELIS)* (2013), 9. DOI: http://dx.doi.org/10.1109/MMUL.2011.34.van

[34] Justin Zobel and Alistair Moffat. 1998. Exploring the similarity space. *ACM SIGIR Forum* 32, 1 (1998), 18–34. DOI: http://dx.doi.org/10.1145/281250.281256