University of Glasgow | School of Computing Science

Multi-Modal Music Recommendation with Denoising Diffusion

William Traynor (2381173T)

April 13, 2023

ABSTRACT

Music recommendation is a crucial task within the multi-billion dollar music industry and music research community, but existing approaches fail to account for the nuanced makeup of tracks by including lyrical and granular audio information. In this study, we propose a general recommendation architecture using variational autoencoders (VAEs) and denoising diffusion to incorporate multi-modal features for better music track recommendation.

Our experimental results on the Music4All-Onion dataset demonstrate that the MacridVAE architecture outperforms other models, and multi-modal incorporation can increase performance by 15% on NDCG@10.

Our study also shows that denoising diffusion improves model initialisation and efficiency per epoch. Additionally, our findings suggest that denoising diffusion could be a promising solution for addressing the cold-start problem in music recommendation.

1. INTRODUCTION

Music platforms, such as Spotify[1], serve as a valuable recommendation medium in which the tracks that are recommended are not just mere songs, but also rich in information and artistic nuances. One way to understand these aspects of the tracks is by gathering information on their lyrics and acoustics. Having access to this information will allow for a better understanding of the lyrical themes and the acoustic makeup of songs in the search for better recommendations. Furthermore, due to the artistic nature of songs, music becomes a more subjective medium to recommend as it can depend on the mood of a user and their intent behind listening to music [12]. Music recommendation is an important recommendation task because it is a technology responsible for large proportions of income within a multi-billion dollar industry.

Many recent approaches to music recommendation fail to account for the nuanced make-up of the track such as lyrical and granular audio information [21, 39, 31]. While there do exist some methods that make use of multi-modal features [25, 38, 23], they do not solve the task of track recommendation which is the problem we tackle in this research. We look to improve the performance of existing baseline models (MultiVAE [20], MacridVAE [22]) on the task of track recommendation by implementing an architecture that accounts for multi-modal features. Both audio and lyrical features are important in music recommendation, as there exists semantic information within a track that is exclusively expressed within each mode [2].

Furthermore, popular recommendation architectures like BPR [29] and LightGCN [14] do not allow for simple and obvious incorporation of multi-modal information into the model. Thus, we propose using variational autoencoder (VAE) general recommendation approaches, specifically MultiVAE [20] and MacridVAE [22], and denoising diffusion [16]. The motivation behind using VAEs is that these architectures provide a simple solution for multi-modal information incorporation and, based on our experimental studies, result in greater performance due to their ability to account for multi-modal information and model user intent - the motivation or goal that drives a user to listen to a given track. Denoising diffusion also allows for easy integration of multi-modal information and has demonstrated superior performance to VAE approaches in image generation tasks [16, 37]. Therefore, we aim to investigate whether they can improve the performance of music recommendation models by effectively incorporating multi-modal information.

Our methodology and results demonstrate that MacridVAE is the best-performing architecture on our music recommendation dataset over NDCG, MRR, Recall and HIT (all @10). Our results also show that multi-modal incorporation can increase performance by 15% on NDCG@10. Furthermore, from our results, we see that using denoising diffusion can lead to better initialisation and increased training efficiency per epoch.

Our contributions include:

- We propose a general recommendation architecture that incorporates multi-modal information and denoising diffusion, which can significantly improve performance and initialisation, respectively.
- We propose a multi-modal conditioning component that can be integrated for music recommendation.
- From our experimentation, we find that the use of conditioning conflicts with the task of modelling user intent.

2. BACKGROUND

2.1 Music as a Recommendation Medium

Music recommendation is a problem within the industry of streaming and in computational music research. Annual revenue for global music streaming has increased exponentially in the last 10 years with $25.1bn in revenue in 2021 alone [1]. In addition, there are five streaming services with at least 10% of the global market share [1]. Due to the indus-

[1]https://www.spotify.com/

try's quick growth and healthy competition, platforms must be able to attract new users while retaining their current ones.

There are various subtle aspects of music that can influence whether someone likes a song or not. Some users may like a song because the lyrics poeticise subjects they feel close to whereas other people may like a song purely for how it sounds. This could be due to a host of musical aspects like melody, rhythm or bass and due to the complexity of music and user preferences, we cannot be sure what musical aspects, if any, are sufficient. This also occurs because audio and lyrical features can be contradictory and exclusively represent certain semantic information within a track [2].

In recommendation mediums such as retail and accommodation, user-generated tags and reviews can give detailed information on items [5], but, this can result in contradictory information that could lead to confusion in the model [11]. So, we want to look at more granular and unbiased information for our track catalogues like the lyrics in text and granular audio features. This will allow for more in-depth knowledge of the items a user enjoys. In turn, we will allow the model to learn more complex representations by providing lower-level acoustic and lyrical information. Our methodology and the proposed model are not subject to this problem as we do not use user-generated information - preventing potential bias and contradiction in information.

Why is it different?

Music recommendation poses a more difficult recommendation task than other mediums for a variety of reasons; track duration is generally short, music is a highly saturated medium, content-based features are more influential, repeated recommendations can be well-received, music is emotive, music is consumed sequentially and music consumption is regularly experienced passively [33]. These factors indicate why music is a difficult medium to recommend and worthy of specific research.

Furthermore, the same song can evoke different emotions in different people depending on how they absorb the song. Additionally, depending on the user's level of expertise they may classify and determine song similarity differently [26]. Additionally, a song can evoke different emotions in the same person depending on the context in which they are listening to it. This is because past experiences and the current emotional state of users influence their perception of music and song similarity [12]. This would imply that to recommend well, what the user has listened to is not enough, we need to know why they have listened to it. Thus, music is a recommendation medium that is reliant on user intent. In our methodology we account for this and use a model architecture that learns user intent, MacridVAE [22].

2.2 Multi-Modality Information in Music Recommendation

Multi-modal recommendation refers to the idea of a recommender system that is concerned with the content of items [50]. The motivation behind using multi-modal information is to provide the model with more valuable information to mimic the information the user would account for when listening to the track. *For example, when a user decides if they like a song they will generally take into account the lyrics spoken by the artist as well as the non-verbal audio aspects (melody, rhythm, base, tempo and so on).*

The use of lyrics and audio within computational music research is primarily applied to emotion and mood recognition. Various authors have spoken about the efficacy of using multi-modality in music-related recommendation tasks [19, 28, 24, 38]. However, the problem with these solutions is that they do not use granular audio information, except for [38] which uses audio features extracted from downloaded audio previews of the songs. However, while it could be a fair assumption, it may be incorrect to assume audio previews of a song are a good representation of their acoustic composition. Nonetheless, this paper showed that the inclusion of their audio features improved performance. The other papers, however, make use of high-level features that aim to be representative of the audio. These features will provide numerical values for aspects like "acousticness", "danceability", and "loudness". To fully utilise a neural network's ability to model complex functions we want to be using as close to raw audio as possible [17]. Our methodology and proposed model mitigates this problem as we use only granular lyrical and audio information.

Various authors have concluded that the audio and lyrical information is sufficient and can improve work in music recommendation [42, 41]. Additionally, we cannot use audio or lyrical information exclusively as they are independent modalities and there exists semantic information within a track that is uniquely expressed within each mode [2]. Our research employs audio and lyrical features to enhance performance while excluding track-specific details such as song titles, album names, and album covers. This is because the value of such information is unknown, and acoustic and lyrical information is sufficient to improve performance.

Outwith the task of music recommendation, multi-modal models are proven to have success in other recommendation topics like fashion and short-form video [49, 50]. This speaks to the generalisability in the value of using multi-modal information in the recommendation and supports our application to the task of music recommendation. The work from Wu et al. [49] introduced a new model for interactive recommendation called the multi-modal recurrent attention network (MMRAN). Their model utilises a different approach to handle textual feedback sequences and visual item sequences in order to combine them into a multi-modal sequence through the use of a gated recurrent network and multi-head attention. While they showed their novel approach to be more effective at capturing multi-modal information, they applied their work to the task of sequential recommendation. Additionally, their textual information is one-hot encoded as they claim a fashion vocabulary is not diverse, thus, does not require a pre-trained language model like BERT. Yi et al. [50] propose a method called Multi-Modal Graph Contrastive Learning (MMGCL) which utilises positive pairs of nodes to train an encoder to learn the correlation between different modalities. To augment the data, two techniques are used: modality edge dropout and modality masking. The former removes edges from different modality graphs while the latter selectively masks one modality of user/item features. Negative samples are generated by perturbing one modality of the positive sample. Overall, the joint contribution of these techniques ensures that the encoder effectively learns from all modalities. Both of these approaches use different architectures compared to this work and use neural parameterised approaches to multi-

modal fusion rather than our proposition of a parameterless approach discussed in section 4.3.

2.2.1 *Multi-Modal Fusion Methods*

One of the main problems in using multi-modal information is how we meaningfully represent the information. This practice is known as data fusion. Two of the most popular methods for this task are aggregation-based and alignment-based fusion [44]. Alignment-based fusion aims to line up the embeddings of all modalities through the use of regularisation loss and thus keeps the vectorised modality information separate throughout training. Aligment-based fusion is more complex and in this work, we only look at aggregation-based fusion.

Aggregation-based fusion merges different mode information into a single vector representation by use of some operation like pooling [10], concatenation [52] or averaging [13]. This single vector is then used as input for the model of choice.

Another simple method of multi-modal fusion is tensor fusion. Tensor fusion can represent intra-modality and inter-modality dynamics [51]. The author's use of tensor products means that this method has learned parameters. However, they do admit their multi-modal representations are subject to high dimensionality. This is a problem we have mitigated in our methodology.

2.3 Recommendation Architectures

Collaborative filtering (CF) is one of the most popular methods in recommender systems and works on the assumption that people who have a history of shared interests are more likely to agree in the future [32]. BPR-MF [29] that is known to perform well [53] and commonly used as a baseline architecture [15, 43, 14]. Another widely used methodology in recommender systems is a graph-based approach [48]. A popular model of this type is LightGCN [14]. The authors found this model to obtain strong performance on a variety of datasets. Furthermore, this is another commonly used baseline model in other works [50, 47]. Finally, there exist autoencoder-based approaches such as MultiVAE and MacridVAE [22]. The issue with collaborative filtering and graph-based architectures is that they do not allow for simple and obvious incorporation of multi-modal information, whereas this task is simple in the aforementioned autoencoder-based approaches. We will delve further into the workings of these architectures in section 3.2.

3. PRELIMINARIES

3.1 Problem Formulation

Our initial data consists of an interaction matrix $X \in \{0,1\}^{U \times I}$, where we have U users and I tracks. The interaction between the u^{th} user and i^{th} track is denoted $x_{i,u}$, where a value 1 indicates the user has listened to the track and a value 0 indicates the user has not listened to the track. Thus, we do not account for repeated listens of a track by a user, the user has either interacted with the track or not.

The objective for the problem is to provide item recommendation likelihoods for each user, u over all tracks, I. Our output recommendations are then the top-k unheard tracks our user is likely to enjoy.

3.2 VAEs

We will begin with the idea of a Variational Autoencoder (VAE), explaining the workings introduced by Kingma & Welling [18]. In their work, they formalised the goal of autoencoders (AEs) which is to reconstruct input data, while compressing it, so as to discover a more efficient representation of the information. With VAEs, the input is aligned with a probability distribution, p_θ (parameterised by θ, rather than a fixed vector. To generate samples resembling a real data point x we begin by sampling a latent encoding vector z from a prior distribution $p_{\theta^*}(z)$. From this, we generate our value x from a conditional distribution $p_{\theta^*}(x|z)$). The parameter θ^* is optimised to maximise the probability of generating a real sample. This formulation is expensive as we need to check all possible values of z, thus we reduce the value space by using an approximation function, $q_\phi(z|x)$ (parameterised by ϕ) that gives a likely output from an input x.

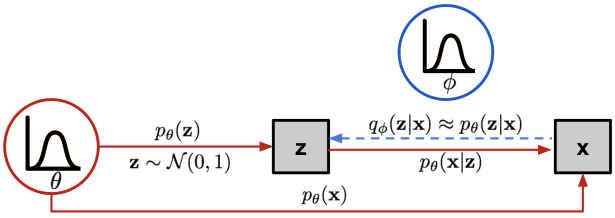


Figure 1: The graphical model involved in Variational Autoencoder. Solid lines denote the generative distribution $p_\theta(\cdot)$ and dashed lines denote the distribution $q_\phi(z|x)$ to approximate the intractable posterior $p_\theta(z|x)$. The diagram and caption come from Weng L. [45].

3.2.1 *Conditioning Approximation Functions*

Conditional variational autoencoders were introduced by Sohn et al. [36]. These models have the same objective as standard VAE models, however, they use additional information (conditioning) to guide the approximation. Conditioning is additional information, in our case multi-modal information, that relates to the function we try to approximate. To condition, we define a conditioning vector, c, which we account for in our function approximation.

The encoder tries to learn $q_\phi(z|x,c)$, which is equivalent to encoding the initial data x into the hidden representation and conditioned by c. The decoder part tries to learn $p_\theta(x|z,c)$ which decodes the hidden representation to input space conditioned by c.

Conditioning has become a popular and successful method of improving model performance and making use of additional information. The most prominent inspiration lead to success in image diffusion models with CLIP embeddings [27] to direct the models from random noise with some direction and knowledge of what the end image should be.

3.2.2 *VAEs in Recommender Systems*

There are two primary works in VAE-based recommender systems which have formed the basis of the implementation of this project.

Firstly, Liang et al. [20] extended the use of VAEs to collaborative filtering for implicit feedback. Their method con-

sists of sampling a l dimensional latent representation, z_u for each user, u, which is then turned into a nonlinear function $f_\theta(\cdot) \in \mathbb{R}^I$ to obtain a probability distribution over the set of I items, $\pi(z_u)$. This probability distribution is where we assume the true interaction data x_u has been drawn from. With that, we have the formal definitions of z_u, $\pi(z_u)$ and x_u shown below.

$$z_u \sim \mathcal{N}(0, I_l), \tag{1}$$

$$q(z_u) \propto \exp\{f_\theta(z_u)\}, \tag{2}$$

$$x_u \sim Multi(I, q(z_u)). \tag{3}$$

To learn the generative model we want to learn the parameters $f_\theta(\cdot)$. We do this by approximating the posterior distribution $p(z_u|x_u)$. In the MultiVAE paper, they do this through variational inference. This is like the approximation function q_ϕ from section 3.2. Given the set of items I, x_u sampled from a multinomial distribution with probability $q(z_u)$ The authors set their approximation distribution $q(z_u)$ to be a fully factorised Gaussian distribution, $q(z_u) = \mathcal{N}(\mu_u, diag\{\sigma_u^2\})$.

Improving upon MultiVAE, Ma et al. [22] introduced the MacridVAE model that looked to disentangle the latent defining aspects hidden within the recommendation data. This aim of disentanglement results in more explainable results, mitigating incorrect inferences in the scenario of limited training data. This is particularly applicable to the problem of music recommendation due to the high sparsity of interactions and a large catalogue of items. The main contribution of the author's implementation is the use of macro and micro disentanglement.

Macro disentanglement looks at learning the diverse interests of users (the same user can like music from a variety of genres). This is done through learning a K dimensional factorised representation of a user, u, where we assume K distinct high-level user interest concepts. This representation is defined as $z_u = [z_u^{(1)}, z_u^{(2)}, ..., z_u^{(K)}] \in \mathbb{R}^{Kd}$, where d is the dimensionality of each vector $z_u^{(k)}$.

Micro disentanglement learns user intentions (why the user is listening to the song). This also looks at user preference in a finer fashion. So for a user u, we would expect each of the d dimensions in $z_u(k)$ to encapsulate the artists, themes and sounds a user appreciates in a given category K. This methodology is supported our motivation for modelling user intent. As previously mentioned, music recommendation is a task that is heavily decided by the user's current emotional state and thus it would be advantageous to learn the intentions behind our user's listening habits.

3.3 Diffusion Models

Diffusion models simulate how data points diffuse through the latent space to learn the latent structure of a dataset. Multiple works have proposed similar implementations for a diffusion model [35, 16]. Diffusion models learn through the forward and reverse diffusion processes.

3.3.1 *Forward Diffusion Process*

If we take a data point x_0, sampled from our real data distribution $q(x)$, we can define the forward process as a sequence in which we repeatedly add Gaussian noise to the sample over our pre-defined number of steps, T. This gives us a sequence of increasingly noisy images, $x_0, ..., x_T$ with x_0 being the original data and x_T essentially being pure noise.

Noise is added by a schedule, $\{\beta_t \in (0,1)\}_{t=1}^T$, where β_t is the variance of the normal Gaussian noise distribution at the timestep, t.

So our method of adding noise to images is as defined below:

$$q(x|x_{t-1}) = \mathcal{N}(x_t;\ \sqrt{1-\beta_t}\, x_{t-1},\ \beta_t\ ). \tag{4}$$

From this, we can see that as our step, t, increases our data, x_0, will become increasingly noisy with the final step, T, becoming normal Gaussian noise as $T \to \infty$.

Fortunately for computation, x_t can be sampled without the need to perform the sequence of forward diffusion steps that would lead to it.

To do this we first introduce a variable $\alpha_t = 1 - \beta_t$ with $\bar{a}_t = \prod_{i=1}^t \alpha_i$.

This gives, from equation 4,

$$q(x|x_{t-1}) = \mathcal{N}(x_t;\ \sqrt{\alpha}\, x_{t-1},\ (1-\alpha)\ ), \tag{5}$$

that can be reformatted, with use of $\bar{a}$, to

$$q(x_t|x_0) = \mathcal{N}(x_t;\ \sqrt{\bar{\alpha}}\, x_0,\ (1-\bar{\alpha})\ ). \tag{6}$$

3.3.2 *Reverse Diffusion Process*

The motivation here is simple, if we can reverse the forward diffusion process sampling from $q(x_{t-1}|x)$ then we will recreate our initial data point from a state of pure noise, $x_T \sim \mathcal{N}(0, I)$.

Similarly, to VAEs, we cannot estimate $q(x_{t-1}|x_t)$ in a simple manner as the full dataset would be required. Thus, we learn an approximation of these conditional probabilities, p_θ. This is how we can carry out the reverse diffusion process.

3.3.3 *Latent Diffusion*

In order, the speed up the diffusion process, Rombach & Blattmann [30] implemented an architecture in which the diffusion process is run after mapping the original data into a latent space. This reduces the dimensionality which results in decreasing training costs. Their reasoning behind this change was in reference to the task of image generation and posed that the composition of an image remains over compression.

4. METHODOLOGY

4.1 Rationale of Diffusion Integration

Firstly, the motivation behind the use of diffusion as an extension to a VAE approach was due to the fact that diffusion models have seen state-of-the-art performance in the task of image generation. This is a task that can also be solved using VAEs. With that, given two of our baseline models are VAE-based approaches we investigate if diffusion-based approaches can see a similar performance improvement for recommendation as in image generation.

Diffusion is commonly used on image datasets, where the data is continuous, however, in our interaction data the values are discrete and either one or zero. Thus, initially, similar to the methodology in VQ-VAE [40], we encode our interaction matrix, $X \in \{0,1\}^{U \times I}$, into a smaller space U users and I tracks. After we have encoded our data we are

left with a variable, $z \in \mathbb{R}^{U \times L}$, where, L is the number of latent dimensions we choose to map our interaction matrix to. Thus, we have a representation z such that for each user, u, we have z_u that describes each user's item interactions.

Now that we have a continuous representation of the data we can make use of the diffusion step. In our implementation, we add a further diffusion step to obtain the approximation $q_\theta(z|x)$.

In performing the denoising diffusion we use the training and sampling algorithms from Ho et al. [16].

4.2 Model Architectures

In our work, we propose two model architectures that are enhanced with the use of diffusion sampling. The first is inspired by the MultiVAE model architecture [20]. This implementation works by simply obtaining the approximation function mean from the diffusion step and using that as z' to input to our decoder and then obtain our recreated interaction matrix X'. This architecture is shown in Figure 2.

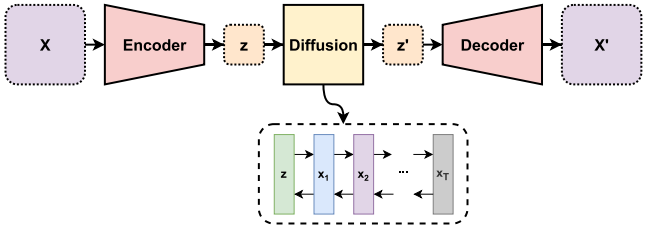


Figure 2: An architecture diagram of the proposed method of integrating diffusion architecture to the task of recommendation.

Our second model architecture is similar to the MultiVAE but follows the implementation of the MacridVAE model [22]. This means we perform recommendation predictions over the K latent factors used to represent user intent. Then we recreate the interaction matrices for each latent factor k and sum this to obtain our total reconstruction loss. This architecture can be shown in Figure 3.

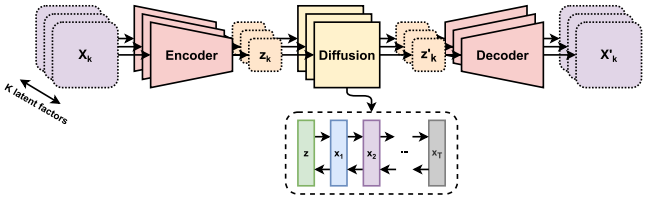


Figure 3: An architecture diagram adapted from the MacridVAE paper [22] demonstrating the inclusion of diffusion architecture to the task of recommendation while modelling user intent.

4.3 Multi-Modality Integration

4.3.1 Multi-Modality Representation

Given our two modes of information - textual and audio - we look to create a single representation that we can pass into our model.

The textual data we use is from the original text file lyrics provided by the Music4All-Onion dataset [23]. The textual data is passed through a pre-trained BERT [6] model to obtain a 768-dimensional vector representation of the song lyrics. The use of a BERT model is due to the fact song lyrics create a large vocabulary and can utilise the complexity of BERT embeddings as opposed to smaller vocabularies. Then we use principal component analysis (PCA) [46], still a popular method for multi-modal dimensionality reduction [50, 7], to reduce the vector to 128 dimensions. Thus, we are left with a matrix, $v^l = [v_0^l, v_1^l, ..., v_I^l]$ where v_i^l represents the lyrical information for a given track, i.

Next for the audio information, we initially start with a 512 dimension i-vector that represents the 13 Mel-frequency cepstral coefficients for each track [8]. The Mel-frequency cepstrum (MFC) is a sound processing technique that utilises a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency to represent the short-term power spectrum of a sound. The Mel-frequency cepstral coefficients are a group of coefficients that together comprise an MFC. Similarly to the lyrical vectors, we use PCA to reduce the initial i-vector to 128 dimensions. Thus, we are left with a matrix, $v^a = [v_0^a, v_1^a, ..., v_I^a]$ where v_i^a represents the lyrical information for a given track, i.

Now that we have our two lyrical and audio 128-dimensional vector representations we look to perform tensor fusion [51]. This is done by performing the tensor-product between our audio and lyric matrices, v^a and v^l, respectively. This results in a tensor, $v^a \otimes v^l$ of shape $[I, 128, 128]$. From here, we flatten our two final dimensions and again perform PCA to result in our final modality information matrix of shape $[I, 128]$. This process is shown in Figure 4.

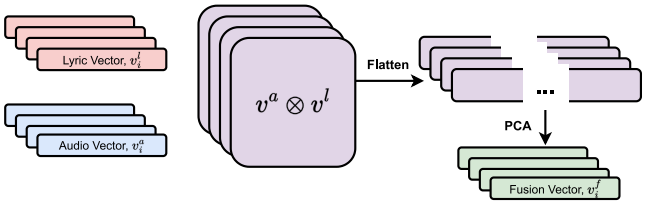


Figure 4: A diagram of the multi-modal fusion approach used in our methodology.

The reasoning behind this choice of fusion is that it is simple to perform and provides meaningful information. One issue with this method though is the high dimensionality of the output [51]. So, we use the tensor-product with PCA to reduce the dimensionality of the output and obtain the most important dimensions on which the audio and lyrical vectors differ amongst our collection of tracks, I. This means our resultant matrix to represent the multi-modal make-up of the track catalogue contains the dimensions from the tensor-product which see the most variance and thus the dimensions in which audio and lyrical information will be the most informative.

4.3.2 User Representation

With our fusion vectors, v_i^f, for each track, i, we calculate user representation vectors for each user based on the tracks they have interacted with. To do this, we perform a max pooling over all fusion vectors, v_i^f, where the user, u has interacted with track i. More formally, for each user, u, we take $MaxPool(\{v_t^f | x_{i,u} = 1\})$ as the user conditioning vector.

4.3.3 Multi-Modality Models

Once we have obtained our multi-modal information in the form of a single vector, c, we will use this vector as a conditioning input in both of our VAE approaches.

For this, we follow the standard conditional variational autoencoder objective from [36]. This differs from the standard VAE objective because rather than approximating $q_\theta(z|x)$ we approximate $q_\theta(z|x, c)$. This means that when inputting to our model we concatenate the encoded input with the conditioning c, shown in Figure 5.

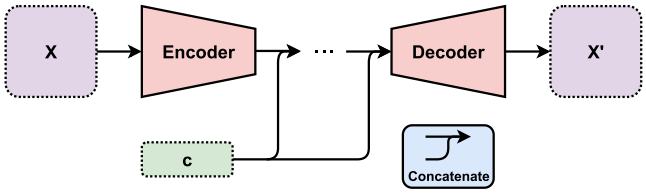


Figure 5: A high-level architecture diagram of the proposed method of integrating multi-modal information.

5. EMPERICAL STUDY

5.1 Datasets

There are many music recommendation datasets available, including two provided by Spotify: the Million Playlist Dataset (MPD) [4] and the Music Streaming Sessions Dataset (MSSD) [3]. While these datasets are reliable, the MPD lacks interaction data and focuses on playlist names and groupings and the MSSD is focused on session recommendation and lacks track titles and raw audio features.

LastFM is a prominent entity in the realm of music recommendation and has made available several expansive datasets. The latest of these datasets, LFM-2b [34], comprises over 2 billion listening interactions with associated timestamps, from over 150 thousand users, and pertaining to 2 million tracks. This dataset provides lyrical features in the form of lexical features, compression ratios, entropy values, and vector embeddings, but, does not provide audio features that closely approximate raw audio.

There also exists the WASABI dataset [9] which is a large corpus of songs which in the most recent work enriched the catalogue with song lyrics. While this corpus is extensive and could be paired with interaction data from LFM-2b it still lacks any audio information.

Finally, the Music4All-Onion dataset [23] provides us with the last piece of information required. This dataset merges with LFM-2b's interaction dataset and provides raw lyric text files allowing the use of our own lyrical embeddings. While there are still no raw audio files in this dataset, they do provide short-term and block-term audio features. Short-term features have been obtained at the frame level whereas block-level features are determined from multi-second intervals.

Information on the aforementioned datasets is shown in Table 1. We can see clearly that Music4All-Onion is the only dataset that meets our mode information needs. Furthermore, we see that it has a far greater user-to-track ratio and presents a lower sparsity than the LFM-2b dataset. This would imply that the Music4All-Onion dataset may be easier to obtain good recommendations and this allows more focus on the impact of multi-modal information and diffusion on a dataset where we can already achieve good recommendation performance.

5.2 Baseline Models

In choosing a model for this project I made the decision to compare 5 established recommendation models. I compared two general recommenders (BPR-MF [29] and LightGCN [14]) and two variational autoencoder-based recommenders (MultiVAE [20], MacridVAE [22]).

BPR-MF: A model using MF that incorporates a general optimisation criterion - the maximum posterior estimator - obtained from a Bayesian study of the issue is known as BPR-Opt for personalised ranking.

LightGCN: A GCN-based model that encodes collaborative signals in the form of high-order connectivities by performing embedding propagation. This model was found to improve efficiency in comparison to other GCN-based solutions through a reduction in model complexity while improving performance.

MultiVAE: This approach surpasses the modelling limitations of linear factor models that are currently used in collaborative filtering. The authors introduce a generative model with multinomial likelihood and use Bayesian inference to estimate the parameters. They also introduce a new regularisation parameter for the learning objective, which is essential for achieving strong performance.

MacridVAE: The authors of the study developed an approach that achieves macro disentanglement by identifying high-level concepts associated with user intentions and capturing user preferences for each concept independently. They also introduced a micro-disentanglement regulariser that enforces each dimension of the representations to independently reflect a unique low-level factor. This regulariser is based on an information-theoretic interpretation of VAEs.

5.3 Experimental Settings

In our experimental settings, we split the interaction data into 80/10/10 train/validation/test split for users. In our validation and testing, we evaluate our model predictions, for each user, against the entire item set. Thus, we train our model on 80% of users and then validate and test our results on 10% of users respectively, to estimate how well our models generalise for unseen users.

5.4 Parameter Tuning

To tune the model parameters we train each model on standard hyperparameters. Furthermore, we tune on model-specific parameters also if they exist. The learning rate is tuned over the values $[.01, ..., .0001]$ and batch size is tuned over the values $[64, ..., 1024]$ - both values are sampled using a logarithmic scale. To tune our parameters we use Bayesian optimisation with a maximum of 75 runs. Furthermore, if we find optimum values at the border of a given value range we further tune these parameters with extended value limits.

5.5 Evaluation Metrics

In our evaluation we use four popular metrics in recom-

Name	Lyrics	Audio (LL)	Users	Tracks	Sparsity
MSSD	×	×	-	4.7M	-
LFM-2b	✓	×	150k	2M	99.97
Music4All-Onion	✓	✓	119k	57k	99.29
WASABI	✓	×	-	1.73M	-

Table 1: Information on existing music datasets.

mender systems; Normalised Discounted Cumulative Gain (**NDCG**), Mean Reciprocal Rank (**MRR**), **Recall** and Hit Ratio (**HIT**). NDCG measures the quality of a ranked list of items based on relevance and position. MRR evaluates how quickly the first relevant item is found in the ranked list of items. Recall measures the completeness of the recommendation by calculating how many relevant items were recommended compared to the total number of relevant items that were available to be recommended. The HIT ratio measures the percentage of recommended items that were eventually selected or interacted with by the user. In our evaluation, we take all metrics @10 - meaning we are only concerned with the top 10 recommendations from the model. Also, we obtain our final metrics by averaging the results from three different random seeds (2020, 2021, 2022).

5.6 Research Questions

From the setup defined in this section, we perform experiments to provide answers to the following research questions;

RQ1: What model architecture performs best for track recommendation?

RQ2: How does the incorporation of multi-modal information impact performance?

RQ3: How does the use of denoising diffusion impact performance?

6. EVALUATION RESULTS

Our discussion of the results is divided per research question, our results are displayed in Figures 6, 7 and 8 as well as Tables 2, 3, 4 and 5.

RQ1: What model architectures perform the best on our music dataset?

The first aim of this work is to determine what model architecture is the best for music recommendation. Table 2 shows the performance of the aforementioned baseline models on track recommendation as defined in section 5.3 over the pre-defined recommendations metrics from section 5.5. From Table 2, we see that the MacridVAE architecture is the best-performing model. This model presents with performance twice as strong as the second-place model in NDCG@10 and far greater than a similar model in MultiVAE. The major difference between the MultiVAE and MacridVAE approaches is the modelling of user intent by the MacridVAE model. This supports the point made in section 2.1 that user intent is a highly important aspect of the consumption of music. We clearly see from our results in 2 that the ability to understand why users listen to a selection of tracks allows for greater recommendations.

Model	NDCG	MRR	Recall	HIT
BPR-MF	.1872	.3964	.0911	.6960
LightGCN	*.1625*	*.3477*	*.0804*	*.6475*
MultiVAE	.0816	.1933	.0393	.4374
MacridVAE	**.3245**	**.5567**	**.1532**	**.8187**

Table 2: Baseline model results where **bold** indicates the best result and *italics* the second-best result. All metrics @10.

RQ2: How does the use of multi-modal information impact performance?

Now, we test and evaluate the impact of incorporating multi-modal information into our suitable model architectures. From Table 3, we can see that the incorporation of multi-modal information leads to a marked improvement in the performance of the standard MultiVAE approach. However, this performance improvement does not persist with the MacridVAE architecture. The incorporation of multi-modal information in MacridVAE is done by extending the latent vector for each user by the length of the multi-modal user information. Thus, we are learning user intent with help from the user's multi-modal information which could be conflicting information and cause weaker performance when incorporating multi-modal information in the MacridVAE architecture.

Additionally, we study the effectiveness of accounting for multi-modal information as opposed to uni-modal. We also look at how effective our method of fusing multi-modal information is against a naïve method in concatenation. These tests were performed using the MultiVAE approach as it saw improvement in performance from conditioning and thus serves as the suitable architecture for this evaluation. Our results in Figure 6 and Table 4 show that our method of multi-modal fusion performs best, achieving the greatest score across all metrics. We also note our method of multi-modal fusion outperforms the naïve fusion method of concatenation. Additionally, lyrics alone serve as better conditioning than a concatenation of audio and lyrics. This demonstrates the importance of combining multi-modal information correctly as poor fusion can lead to worse results than uni-modal information and even no conditioning. In comparison to the baseline model, with no conditioning, we see that conditioning alone does not lead to better performance as audio conditioning performs far worse than the baseline model. This shows that, while additional information can result in increased performance, the information must also effectively convey aspects of the tracks that influence users, otherwise, it can hinder our model from achieving strong performance.

	Without Conditioning		*With Conditioning*		
Model	**NDCG**	**Recall**	**NDCG**	**Recall**	**NDCG Change**
MultiVAE	.0816	.0393	.0980	.0461	+15.2%
MacridVAE	.3245	.1532	.2837	.1342	−13.6%

Table 3: Model performance measured with and without multi-modal conditioning. All metrics @10.

Conditioning	**NDCG**	**MRR**	**Recall**	**HIT**
Audio	.0176	.0527	.0059	.1320
Lyric	*.0860*	*.2118*	*.0372*	*.4551*
Concatenation	.0739	.1737	.0323	.4013
Tensor Fusion	**.0980**	**.2227**	**.0461**	**.4829**

Table 4: Performance of conditioned MultiVAE model using different modal conditioning where **bold** indicates the best result and *italics* the second-best result. All metrics @10.

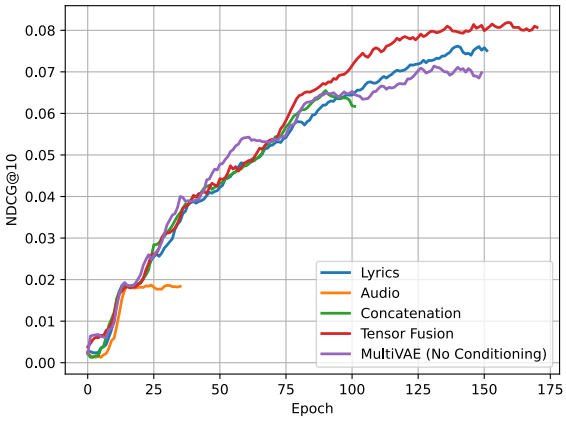


Figure 6: Training curves of the MultiVAE model using different conditioning.

RQ3: How does the use of denoising diffusion impact performance?

Based on the outcomes presented in Table 5, we observe that the incorporation of denoising diffusion does result in a marked improvement to the MultiVAE model. However, we do not see the same result of marked improvement in the performance of the MacridVAE model architecture. The underlying reason for this disparity requires further investigation. However, we speculate that the disentanglement aspect of modelling user intent and the objective of the MacridVAE model may not combine efficiently with denoising diffusion.

Based on the analysis of Figures 7 and 8, we observe a marked improvement in the initialisation of our models in terms of time and epochs by utilising denoising diffusion. Specifically, for the MacridVAE-based implementation, after approximately thirty minutes of training, the model attains an NDCG@10 score of approximately 0.24, which takes the baseline architecture two hours of training to achieve. Moreover, with regards to epochs, the MacridVAE-based implementation attains an NDCG@10 score of approximately 0.25 within five epochs, whereas the baseline implementation requires over one-hundred epochs to reach a similar NDCG@10 score.

Our results exhibit an irregularity in the MultiVAE-based implementation in that while employing denoising diffusion, the model's performance shows a promising start but deteriorates rapidly. This implies that the use of denoising diffusion disrupts the model's training objective. Consequently, more research is necessary to identify an appropriate training objective for this architecture.

Based on our experimental findings regarding Research Question 3 (RQ3), we propose the hypothesis that the strong initialisation of our implementation may suggest a favourable performance in addressing the cold start problem. The cold start problem pertains to the challenge of recommending items for users who are new and for whom we possess limited information on historical item interactions. We base our hypothesis on the observation, from our experimentation, that our implementations exhibit strong performance after a single training epoch, which suggests their ability to function effectively with reduced information.

7. CONCLUSION

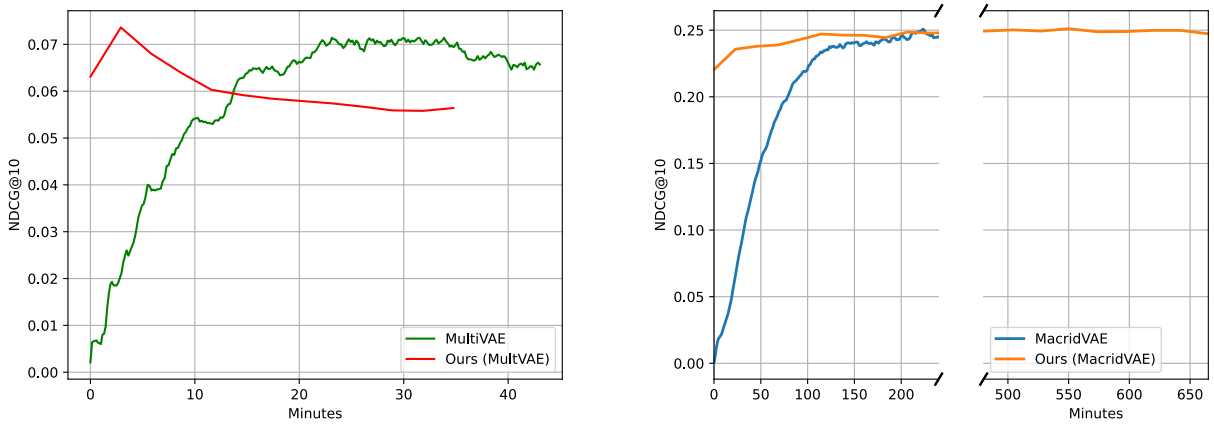
In this work, we proposed a general recommendation architecture that incorporates multi-modal information and denoising diffusion, which can markedly improve performance and initialisation, respectively. We also propose a multi-modal conditioning component that can be integrated for music recommendation. Furthermore, we investigated and declared why music is an important and difficult medium to recommend.

Overall, the evaluation section presents the results of the study concerning three research questions. The first question examines the best model architecture for music recommendation, and the study finds that MacridVAE performs the best due to its ability to model user intent. The second question focuses on the impact of incorporating multi-modal information, and the study reveals that while incorporating multi-modal information improves the performance of the standard MultiVAE approach, it does not persist with the MacridVAE architecture. The study also shows that the fusion of lyrical and audio information using tensor fusion leads to better performance than audio or lyrical information alone or using a naïve fusion method in concatenation. The third question investigates the impact of denoising diffusion, and the study finds that its incorporation has a positive impact on the initialisation of the models in terms of time and epochs, but its effectiveness in improving performance is not consistent across all model architectures. The study proposes a hypothesis that the strong initialisation of the implementation may suggest a favourable performance in addressing the cold start problem.

We plan to expand on our current research by exploring more advanced and diverse multi-modal fusion techniques to investigate whether there are further improvements that can be made in the performance of conditioning and multi-modal information usage. Additionally, we aim to evaluate our current implementation against more complex methodologies to further verify our findings. As part of our future

Model	NDCG	MRR	Recall	HIT
BPR-MF	.1872	.3964	.0911	.6960
LightGCN	.1625	.3477	.0804	.6475
MultiVAE	.0816	.1933	.0393	.4374
Ours (MultiVAE)	.08774	.2066	.0439	.4399
MacridVAE	**.3245**	*.5567*	**.1532**	**.8187**
Ours (MacridVAE)	*.3242*	**.5642**	*.1484*	*.8151*

Table 5: Baseline model results with our denoising diffusion models, where **bold** indicates the best result and *italics* the second-best result. All metrics @10.



(a) MultiVAE (b) MacridVAE

Figure 7: NDCG value in minutes comparing VAE baseline models and our diffusion-based approaches.

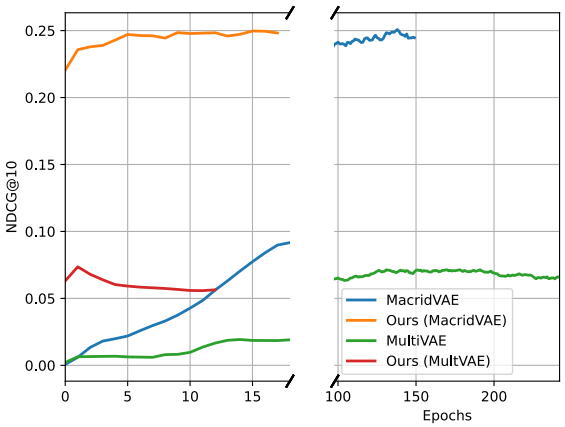


Figure 8: NDCG value in epochs comparing VAE baseline models and our diffusion-based approaches.

research, we also intend to investigate the potential applications of the strong initialisation of our denoising diffusion method. We believe that the strong performance of denoising diffusion after a small amount of training could be a promising solution to address the cold-start problem.

Acknowledgements

I would like to express my deepest gratitude to my advisor Zaiqiao Meng for their support, guidance, and encouragement throughout the project. Their expertise and insights have been invaluable in shaping my research and helping me navigate through the challenges.

I would also like to extend my heartfelt appreciation to my family for their unconditional support throughout my academic journey. Their unwavering belief in me has been a constant source of inspiration and motivation.

I express my appreciation to the faculty members at the University of Glasgow's School of Computing and School of Mathematics, who have provided valuable guidance and expert advice on the topics that I had inquiries about.

8. REFERENCES

[1] Music streaming app revenue and usage statistics (2022), Sep 2022.

[2] M. Besson, F. Faita, I. Peretz, A.-M. Bonnel, and J. Requin. Singing in the brain: Independence of lyrics and tunes. *Psychological Science*, 9(6):494–498, 1998.

[3] B. Brost, R. Mehrotra, and T. Jehan. The music streaming sessions dataset. In *The World Wide Web Conference*, pages 2594–2600, 2019.

[4] C.-W. Chen, P. Lamere, M. Schedl, and H. Zamani. Recsys challenge 2018: Automatic music playlist continuation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 527–528, 2018.

[5] L. Chen, G. Chen, and F. Wang. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25:99–154, 2015.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[7] X. Du, X. Wang, X. He, Z. Li, J. Tang, and T.-S. Chua. How to learn item representation for cold-start multimedia recommendation? In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3469–3477, 2020.

[8] H. Eghbal-Zadeh, B. Lehner, M. Schedl, and G. Widmer. I-vectors for timbre-based music similarity and music artist classification. In *ISMIR*, pages 554–560, 2015.

[9] M. Fell, E. Cabrio, E. Korfed, M. Buffa, and F. Gandon. Love me, love me, say (and write!) that you love me: Enriching the wasabi song corpus with lyrics annotations. *arXiv preprint arXiv:1912.02477*, 2019.

[10] Y. Ge, Z. Yang, Z. Huang, and F. Ye. A multi-level feature fusion method based on pooling and similarity for hrrs image retrieval. *Remote Sensing Letters*, 12(11):1090–1099, 2021.

[11] J. S. Gómez Cañón, H. Boyer, E. Gómez Gutiérrez, E. Cano, et al. The emotions that we perceive in music: the influence of language and lyrics comprehension on agreement. 2019.

[12] J. S. Gómez Cañón, E. Cano, H. Boyer, E. Gómez Gutiérrez, et al. Joyful for you and tender for us: The influence of individual characteristics and language on emotion labeling and classification. In *Cumming J, Ha Lee J, McFee B, Schedl M, Devaney J, McKay C, Zagerle E, de Reuse T, editors. Proceedings of the 21st International Society for Music Information Retrieval Conference; 2020 Oct 11-16; Montréal, Canada.[Canada]: ISMIR; 2020.* International Society for Music Information Retrieval (ISMIR), 2020.

[13] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part I 13*, pages 213–228. Springer, 2017.

[14] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.

[15] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

[16] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[17] J. Kim, J. Urbano, C. Liem, and A. Hanjalic. One deep music representation to rule them all? a comparative analysis of different representation learning strategies. *Neural Computing and Applications*, 32(4):1067–1093, 2020.

[18] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[19] C. Laurier, J. Grivolla, and P. Herrera. Multimodal music mood classification using audio and lyrics. In *2008 seventh international conference on machine learning and applications*, pages 688–693. IEEE, 2008.

[20] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*, pages 689–698, 2018.

[21] Q. Lin, Y. Niu, Y. Zhu, H. Lu, K. Z. Mushonga, and Z. Niu. Heterogeneous knowledge-based attentive neural networks for short-term music recommendations. *IEEE Access*, 6:58990–59000, 2018.

[22] J. Ma, C. Zhou, P. Cui, H. Yang, and W. Zhu. Learning disentangled representations for recommendation. *Advances in neural information processing systems*, 32, 2019.

[23] M. Moscati, E. Parada-Cabaleiro, Y. Deldjoo, E. Zangerle, and M. Schedl. Music4all-onion—a large-scale multi-faceted content-centric music recommendation dataset. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 4339–4343, 2022.

[24] S. Naseri, S. Reddy, J. Correia, J. Karlgren, and R. Jones. The contribution of lyrics and acoustics to collaborative understanding of mood. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 16, pages 687–698, 2022.

[25] S. Oramas, O. Nieto, M. Sordo, and X. Serra. A deep multimodal approach for cold-start music recommendation. In *Proceedings of the 2nd workshop on deep learning for recommender systems*, pages 32–37, 2017.

[26] H. Pálmason, B. ■. Jónsson, M. Schedl, and P. Knees. Music genre classification revisited: An in-depth examination guided by music experts. In *Music Technology with Swing: 13th International Symposium, CMMR 2017, Matosinhos, Portugal, September 25-28, 2017, Revised Selected Papers 13*, pages 49–62. Springer, 2018.

[27] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[28] S. Raschka. Musicmood: Predicting the mood of music from song lyrics using machine learning. *arXiv preprint arXiv:1611.00138*, 2016.

[29] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.

[30] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[31] N. Sachdeva, K. Gupta, and V. Pudi. Attentive neural architecture incorporating song features for music recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 417–421,

2018.

[32] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295, 2001.

[33] M. Schedl. Deep learning in music recommendation systems. *Frontiers in Applied Mathematics and Statistics*, page 44, 2019.

[34] M. Schedl, S. Brandl, O. Lesota, E. Parada-Cabaleiro, D. Penz, and N. Rekabsaz. Lfm-2b: A dataset of enriched music listening events for recommender systems research and fairness analysis. In *ACM SIGIR Conference on Human Information Interaction and Retrieval*, pages 337–341, 2022.

[35] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[36] K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

[37] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[38] A. Vall, M. Dorfer, H. Eghbal-Zadeh, M. Schedl, K. Burjorjee, and G. Widmer. Feature-combination hybrid recommender systems for automated music playlist continuation. *User Modeling and User-Adapted Interaction*, 29(2):527–572, 2019.

[39] A. Vall, M. Quadrana, M. Schedl, and G. Widmer. The importance of song context and song order in automated music playlist generation. *arXiv preprint arXiv:1807.04690*, 2018.

[40] A. Van Den Oord, O. Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[41] K. Vaswani, Y. Agrawal, and V. Alluri. Multimodal fusion based attentive networks for sequential music recommendation. In *2021 IEEE Seventh International Conference on Multimedia Big Data (BigMM)*, pages 25–32. IEEE, 2021.

[42] M. Vystrčilová and L. Peška. Lyrics or audio for music recommendation? In *Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics*, pages 190–194, 2020.

[43] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019.

[44] Y. Wang, W. Huang, F. Sun, T. Xu, Y. Rong, and J. Huang. Deep multimodal fusion by channel exchanging. *Advances in neural information processing systems*, 33:4835–4845, 2020.

[45] L. Weng. From autoencoder to beta-vae. *lilianweng.github.io*, 2018.

[46] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[47] J. Wu, X. Wang, F. Feng, X. He, L. Chen, J. Lian, and X. Xie. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 726–735, 2021.

[48] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022.

[49] Y. Wu, C. Macdonald, and I. Ounis. Multi-modal dialog state tracking for interactive fashion recommendation. In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 124–133, 2022.

[50] Z. Yi, X. Wang, I. Ounis, and C. Macdonald. Multi-modal graph contrastive learning for micro-video recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1807–1811, 2022.

[51] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.-P. Morency. Tensor fusion network for multimodal sentiment analysis. *arXiv preprint arXiv:1707.07250*, 2017.

[52] J. Zeng, Y. Tong, Y. Huang, Q. Yan, W. Sun, J. Chen, and Y. Wang. Deep surface normal estimation with hierarchical rgb-d fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6153–6162, 2019.

[53] S. Zhang, L. Yao, A. Sun, and Y. Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)*, 52(1):1–38, 2019.