Unsupervised Melody-Guided Lyrics Generation

Yufei Tian[1*], **Anjali Narayan-Chen**[2], **Shereen Oraby**[2], **Alessandra Cervone**[2], **Gunnar Sigurdsson**[2], **Chenyang Tao**[2], **Wenbo Zhao**[2], **Tagyoung Chung**[2], **Jing Huang**[2], **Nanyun Peng**[1,2]

[1] University of California, Los Angeles, [2] Amazon Alexa AI

yufeit@g.ucla.edu, {naraanja,orabys,cervon,gsig,chenyt,wenbzhao,tagyoung,jhuangz}@amazon.com, violetpeng@cs.ucla.edu

Abstract

Automatic song writing is a topic of significant practical interest. However, its research is largely hindered by the lack of training data due to copyright concerns and challenged by its creative nature. Most noticeably, prior works often fall short of modeling the cross-modal correlation between melody and lyrics due to limited parallel data, hence generating lyrics that are less singable. Existing works also lack effective mechanisms for content control, a much desired feature for democratizing song creation for people with limited music background. In this work, we propose to generate pleasantly listenable lyrics without training on melody-lyric aligned data. Instead, we design a hierarchical lyric generation framework that disentangles training (based purely on text) from inference (melody-guided text generation). At inference time, we leverage the crucial alignments between melody and lyrics and compile the given melody into constraints to guide the generation process. Evaluation results show that our model can generate high-quality lyrics that are more singable, intelligible, coherent, and in rhyme than strong baselines including those supervised on parallel data.

1 Introduction

Music is ubiquitous and an indispensable part of humanity (Edensor 2020). Self-served songwriting has thus become an emerging task and has attracted interest in the AI community (Watanabe et al. 2018; Lee, Fang, and Ma 2019; Yu, Srivastava, and Canales 2021; Sheng et al. 2021; Tan and Li 2021; Zhang et al. 2022). However, the task of melody-to-lyric (M2L) generation, in which lyrics are generated based on a given melody, faces three major challenges. First, there is a limited amount of melody-lyric aligned data. The process of collecting and annotating paired data is not only labor-intensive but also requires strong domain expertise and careful consideration of copyrighted source material. Previous works either manually collect a small amount (usually a thousand) of melody-lyrics pairs (Watanabe et al. 2018; Lee, Fang, and Ma 2019), or (Sheng et al. 2021) use the recently publicized data (Yu, Srivastava, and Canales 2021) in which the lyrics are pre-tokenized at the syllable level leading to nonsensical syllables in the outputs.

Another challenge lies in melody-lyric modeling. Compared with uni-modal sequence-to-sequence tasks such as machine translation, the latent correlation between lyrics and melody is difficult to learn. For example, Watanabe et al. (2018); Lee, Fang, and Ma (2019); Chen and Lerch (2020) train RNNs, LSTMs, or SeqGANs (Yu et al. 2017) where the input is a short melody piece and output is a line of lyrics; Sheng et al. (2021) train two transformers with separate encoder-decoders for each modality (lyrics or melody) and adopt cross attention (Vaswani et al. 2017) to learn the melody-lyrics mapping. However, these methods may generate less singable lyrics when they violate too often a superficial yet crucial alignment: one word in a lyric always strictly aligns with one or more notes in the melody (Nichols et al. 2009). In addition, their outputs are not fluent enough because training neural models from scratch cannot compete with massive pre-trained language models (PTLMs).

Besides, current M2L methods, all modelled as sequence-to-sequence, can only produce arbitrary lyrics based on the input melody sequence and have little control over the content to be generated (Chen and Lerch 2020; Xue et al. 2021; Sheng et al. 2021). Such a problem setting is far away from real-world application scenarios where users will likely have an intended topic (e.g., an intended title and a few keywords) to write about. Hence, additional mechanisms to endow a lyric generator with topic relevance are urgently needed.

A promising direction to overcome the first two difficulties is to look into music theory studies, and pose melody alignment as decoding constraints for lyrics (Guo et al. 2022). One interesting discovery is that the *durations* of music notes, not the pitch values, play a significant role in melody-lyric correlation (Dzhambazov et al. 2017). Intuitively, the segmentation of lyrics should match the segmentation of music phrases for breathability (Watanabe et al. 2018). Nichols et al. (2009) also find that shorter note durations tend to associate with unstressed syllables, while longer durations tend to associate with stressed syllables. We find that, even when equipped with state-of-the-art neural architectures, existing lyric generators which are already supervised on melody-lyrics aligned data still fail to capture these simple yet fundamental rules.

With all this in mind, we propose **LYRA**, an unsupervised melody-conditioned *LYRics generAtor* that can generate high-quality lyrics related to a provided topic without

[*] Work was done when the author interned at Amazon.

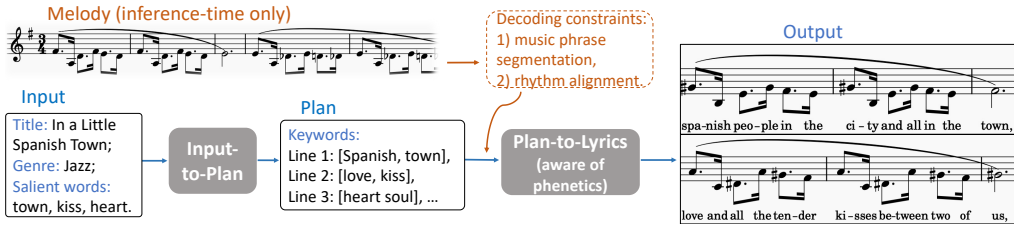

Figure 1: An overview of our approach that disentangles training from inference. During training, our input-to-plan model learns to predict the sentence-level plan (i.e., keywords) given the title, genre, and salient words as input. Then, the plan-to-lyrics model generates the lyrics while being aware of word phonetic information and syllable counts. At inference time, we compile the given melody into 1) music phrase segments and 2) rhythm constraints to guide the generation. Examples of lyrics generated by the complete pipeline can be found here.

training on melody-lyric data. To circumvent the shortage of paired data and comply with a given topic, we train a text-based hierarchical framework (Fan, Lewis, and Dauphin 2018; Yao et al. 2019) that first predicts the content outline and then generates the lyrics. At inference time, we compile a melody into constraints to guide the above text-based generation model. We summarize our contributions as follow:

- To the best of our knowledge, we are among the first to generate melody-constrained lyrics without training on parallel data. To this end, we design a hierarchical framework that *disentangles training from inference*.
- During training, our input-to-plan model learns to generate a plan of lyrics based on the input title and salient words, then the plan-to-lyrics model generates the complete lyrics. To fit in the characteristics of lyrics and melody, we equip the latter model with the ability to generate a sentence with a predefined count of syllables through multi-task learning.
- Inspired by the aforementioned music theories, we design an inference-time decoding algorithm and incorporate two melody constraints: segment and rhythm alignment. Without losing flexibility, we introduce a factor to control the strength of the constraints.
- Preliminary results show that our unsupervised model **LYRA** can beat fully supervised baselines in terms of both text quality and musicality with a large margin.

2 Methodology

2.1 Problem Setup

We aim to generate lyrics that comply with both the provided topic and the melody. The input topic can be further decomposed into an intended title T and a few salient words S to be included in the generated lyrics. Following the settings of previous work (Chen and Lerch 2020; Sheng et al. 2021), we also assume that the input melody $\mathcal{M}$ is predefined. $\mathcal{M}$ can be denoted by $\mathcal{M} = \{p_1, p_2, ...p_M\}$, where each p_i $(i \in 1, 2, ..., M)$ is a music phrase. The music phrase can be further decomposed into timed music notes $(p_i = \{n_{i1}, n_{i2}, ...n_{iN_i}\})$, where each music note n_{ij} $(j \in \{1, 2, ..., N_i\})$ comes with a duration and is associated with or without a pitch value.[1] The output is a piece of lyrics $\mathcal{L}$ that aligns with the music notes: $\mathcal{L} = \{w_{11}, w_{12}, ..., w_{MN}\}$. Here w_{ij} is a word or a syllable of a word that aligns with the music note n_{ij}.

[1]When a music note comes without a pitch value, it is a rest.

Model	(Input: syllable count) Output: generated lyric
Ours, naïve	(7) Cause the Christmas gift was for
Chen and Lerch (2020)	(9) Hey now that's what you ever do my
Sheng et al. (2021)	(6) And then I saw you my
Ours, multitask	(12) Someday the tree is grown with other memories

Table 1: Examples of the desired syllable counts and the corresponding lyrics generated by different models. Our model with multitask learning is the only system that successfully generates a complete line of lyric. The supervised models (Chen and Lerch 2020; Sheng et al. 2021) which are already trained with the melody-lyrics paired data still generate dangling or cropped lyrics.

2.2 Training

Overview and Data An overview of our hierarchical generation method is shown in Figure 1. We finetune two modules in our purely text-based pipeline: 1) an input-to-plan generator that generates a keyword-based intermediate plan, and 2) a plan-to-lyrics generator which is aware of word phonetics and syllable counts. Our training data consists of 38,000 English lyrics and their corresponding genres processed from the Lyrics for Genre Classification dataset.[2]

Input-to-Plan The input contains the song title, the music genre, and three salient words extracted from ground truth lyrics using the YAKE algorithm (Campos et al. 2020). Our input-to-plan model is then trained to generate a line-by-line keyword plan of the song. Considering that in real time we might need different numbers of keywords for different expected output lengths, the number of planned keywords is not fixed. Specifically, we include a placeholder (the <MASK> token) in the input for every keyword to be generated in the intermediate plan. In this way, we have control over how many keywords we would like per line. We finetune BART-large (Lewis et al. 2020) as our input-to-plan generator with format control.

Plan-to-Lyrics Our plan-to-lyrics module takes in the planned keywords as input and generates the lyrics. This module encounters an added challenge. To match the music notes of a given melody at inference time, this module should be capable of generating lyrics with a desired syllable count. If we naïvely force the generation to stop once it reaches the desired number of syllables, the outputs are usually cropped or dangling sentences. Surprisingly, existing lyric generators which are already trained on melody-to-lyrics aligned data also face the same issue. We list their

[2]https://www.kaggle.com/datasets/mateibejan/multilingual-lyrics-for-genre-classification

example outputs of our naive model and two recent works in Table 1. We hence propose to study an under-explored task: generating a line of lyric that 1) has the desired number of syllables, and 2) is not cropped. To solve this, we propose to equip the plan-to-lyrics module with the word phonetics information and the ability to count syllables. To this end, we adopt multi-task learning to include the aforementioned external knowledge during training. Specifically, we study the collective effect of the following tasks:

- Task 1: Plan to lyrics generation
- Task 2: Syllable count: sentence $\rightarrow$ number of syllables
- Task 3: Plan to unnatural lyrics generation where we specify the syllable counts of each word in the output
- Task 4: Word to phoneme translation

We finetune the GPT-2 large model (Radford et al. 2019) on different combinations of the four tasks, as our preliminary experimental results showed that BART could not learn these multi-tasks as well as GPT-2. We show our model's success rate of generating complete chunks of words with a predefined number of syllables in Section 3.1.

2.3 Inference

In this subsection, we discuss how to compile a given melody into constraints to guide the decoding. We start with the most straightforward constraints introduced in Section 1: 1) segmentation alignment and 2) rhythm alignment. Note that such melody constraints can be updated without needing to retrain the model.

Segment Alignment The segmentation of music phrases should align with the segmentation of lyrics (Watanabe et al. 2018). Given a melody, we first parse the melody into music phrases, then compute the number of music notes within each music phrase. For example, the first melody segment in Figure 1 consists of 13 music notes, which should be equal to the number of syllables in the corresponding lyric chunk. Without losing generality, we also add variations to this constraint where multiple notes can correspond to one single syllable when we observe such variations in the gold lyrics.

Rhythm Alignment According to Nichols et al., the stress-duration alignment rule hypothesizes that music rhythm should align with lyrics meter. Namely, shorter note durations are more likely to be associated with unstressed syllables. At inference time, we 'translate' a music note to a stressed syllable (denoted by 1) or an unstressed syllable (denoted by 0) by comparing its duration to the average note duration. For example, based on the note durations, the first music phrase in Figure 1 is translated into alternating 1s and 0s to guide the inference decoding.

Inference Decoding At each decoding step, we ask the plan-to-lyrics model to generate candidate complete words, instead of subwords, which is the default word piece unit for GPT-2 models. This enables us to retrieve the word phonemes from the CMU pronunciation dictionary (Weide et al. 1998) and identify the resulting syllable stresses. For example, since the phoneme of the word 'Spanish' is 'S PAE1 NIH0 SH', we can derive that it consists of 2 syllables that are stressed and unstressed.

Task Name				Success Rate	
T1 Lyrics	T2 Count	T3 Unnatural	T4 Phoneme	Greedy	Sample
✓				23.14%	19.87%
✓	✓			50.14%	44.64%
	✓	✓		55.01%	49.70%
✓	✓	✓		**93.60%**	**89.13%**
✓	✓	✓	✓	91.37%	87.65%

Table 2: Success rate for variants of our plan-to-lyrics model on generating sentences with desired number of syllables.

Next, we check if the candidate words satisfy the stress-duration alignment rule. Given a candidate word w_i and the original logit $p(w_i)$ predicted by the plan-to-lyrics model, we introduce a factor α to control the strength:

$$p'(w_i) = \begin{cases} p(w_i) & \text{if } w_i \text{ satisfies the rhythm alignment,} \\ \alpha p(w_i) & \text{otherwise.} \end{cases} \tag{1}$$

We can either impose a **hard constraint**, where we reject all those candidates that do not satisfy the rhythm rules ($\alpha = 0$), or impose a **soft constraint**, where we would reduce their sampling probabilities ($0 < \alpha < 1$).

3 Experimental Results

3.1 Generating the correct number of syllables

Recall that we trained the plan-to-lyrics generator on four tasks in order to equip it with the ability to generate a sentence with a pre-defined number of syllables. To test this feature, we computed the success rate on a held-out test set that contains 168 songs with 672 lines of lyrics. We experimented with both greedy decoding and sampling and report the result in Table 2. We can see that without multi-task learning, the model success rate is around 20%, which is far from ideal. By gradually training with additional phonetics information, the success rate increases, reaching over 90%. We also notice that the phoneme translation task is not helpful for our goal, so we disregard the last task and only keep the remaining three tasks in our final implementation.

3.2 Baseline Models for Lyrics Generation

We compared the following models. **1. SongMASS** (Sheng et al. 2021) is a state-of-the-art song writing system which leverages masked sequence to sequence pre-training and attention based alignment for M2L generation. It requires melody-lyrics aligned training data while our model does not. **2. GPT-2 finetuned on lyrics** is a uni-modal, melody-unaware GPT-2 large model that is finetuned end-to-end on the same lyrics data as our model. The input prompt contains a song title but not salient words. This serves as ablation of **LYRA w/o rhythm** to test the efficacy of our plan-and-write pipeline without inference-time constraints. **3. LYRA w/o rhythm** is our base model consisting of the input-to-plan and plan-to-lyrics modules with syllable control, but without the rhythm alignment. **4. LYRA w/ (soft/hard) rhythm** is our multi-modal model with music segmentation and soft or hard rhythm constraints. For the soft constraints setting, the strength controlling hyperparameter $\alpha = 0.01$.

Model Name	Topic Relevance			Diversity		Fluency		Music Alignment
	Salient Word Coverage↑	Sent Bleu↑	Corpus Bleu↑	Dist-1↑	Dist-2↑	PPL ↓	Cropped Sentence↓	Stress-Duration
SongMASS	/	0.045	0.006	**0.17**	**0.57**	518	34.51%	58.8%
GPT-2 finetuned	/	0.026	0.020	0.09	0.31	**82**	/	53.6%
LYRA w/o rhythm	**91.8%**	0.074	0.046	0.12	0.45	85	**3.65%**	63.1%
LYRA w/ soft rhythm	89.4%	**0.075**	**0.047**	0.11	0.46	85	8.96%	68.4%
LYRA w/ hard rhythm	88.7%	0.071	0.042	0.12	0.45	108	10.26%	**89.5%**
Ground Truth	100%	1.000	1.000	0.14	0.58	93	3.92%	73.3%

Table 3: Automatic evaluation results on 120 songs. The gold performance is highlighted in a grey background. Among all machine performances, we highlight the best scores in boldface and underline the second best.

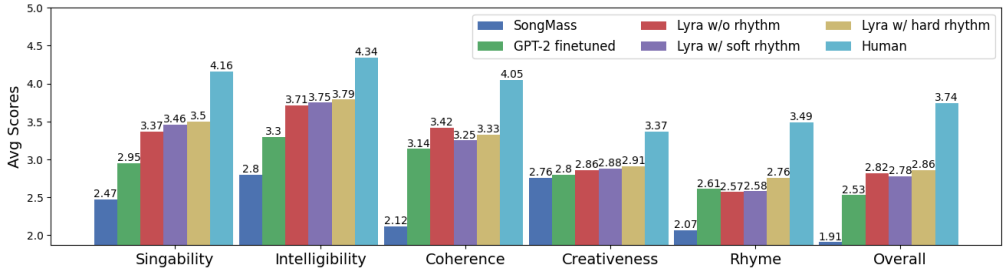

Figure 2: Average human annotator Likert scores for 5 songs on musicality, quality of lyrics alone, and overall. Considering the small sample size, the current results may be inaccurate for the comparison between variations of our own LYRA models, as sometimes the differences are not significant enough. We plan to launch a large-scale human evaluation after recruiting qualified annotators to reduce the randomness.

3.3 Automatic Evaluation

We automatically assessed the generated lyrics on two aspects: the quality of text, and music alignment. For **text quality**, we divided it into 3 aspects: 1) **Topic Relevance**, measured by input salient word coverage ratio, and sentence- or corpus-level BLEU (Papineni et al. 2002); 2) **Diversity**, measured by distinct unigrams and bigrams (Li et al. 2016); 3) **Fluency**, measured by the perplexity computed using Huggingface's pretrained GPT-2 small. We also computed the ratio of cropped sentences among all sentences to assess how well they fit music phrase segments. For **music alignment**, we computed the percentage where the stress-duration rule holds.

We report the results in Table 3. Our LYRA models significantly outperform the baselines and generate the most on-topic and fluent lyrics. In addition, adding rhythm constraints to the base LYRA noticeably increases the music alignment quality without sacrificing too much text quality. It is also noteworthy that humans do not consistently follow stress-duration alignment, meaning that higher is not necessarily better for music alignment percentage. Since the baseline model SongMASS has no control over the content, it has lowest topic relevance scores. Moreover, although the SongMASS baseline seems to achieve the best diversity, it tends to produce non-sensical sentences that consist of a few gibberish words (e.g. 'for hanwn to stay with him when, he got to faney he alone'). Such degeneration is also reflected by the extremely high perplexity and cropped sentence ratio.

3.4 Human Study

We conducted a human evaluation on 5 randomly selected songs. We first used an online singing voice synthesizer (Hono et al. 2021) to generate the singing voice audio clips. We then asked 20 Mechanical Turkers to evaluate the lyrics on six dimensions on a scale from 1 to 5, where a higher score means better quality. For musicality, we asked them to rate singability (whether the melody's rhythm aligns well with the lyric's rhythm) and intelligibility (whether the content of the lyrics is easy to understand without looking at the lyrics). For the lyric quality, we asked them to rate coherence, creativeness, and rhyme. Finally, we asked annotators to rate how much they like the song overall. The average inter annotator agreement (Spearman correlation) is 0.61. The preliminary results are shown in Figure 2.

Generated examples can be found in this anonymous demo website. Surprisingly, SongMASS performs even worse than the finetuned GPT-2 baseline in terms of musicality. Upon further inspection, we posit that SongMASS too often deviates from common singing habits: it either assigns two or more syllables to one music note, or matches one syllable with three or more consecutive music notes. It is clear that LYRA w/o rhythm greatly outperforms both baseline models in terms of singability, intelligibility, coherence, and overall quality, which demonstrates the efficacy of our plan-and-write pipeline with syllable control. The addition of rhythm alignment leads to further improvements in musicality, creativeness, and rhyme, with some sacrifices in coherence and overall quality. Considering the small sample size, it is still early to draw a final conclusion. We plan to launch a large-scale human evaluation after recruiting qualified annotators for greater statistical significance.

4 Conclusion

Our work explores the potential of lyrics generation without training on lyrics-melody aligned data. To this end, we design a hierarchical plan-and-write framework that disentangles training from inference. At inference time, we compile the given melody into 1) music phrase segments, and 2) rhythm constraints. Such melody constraints can be updated without needing to retrain the model. Our preliminary results show that our model can generate high-quality lyrics that are singable, intelligible, and coherent.

Acknowledgements

The authors would like to thank Yiwen Chen for helping to design and providing valuable insights to the human evaluation. We also thank the anonymous reviewers for the helpful comments.

References

Campos, R.; Mangaravite, V.; Pasquali, A.; Jorge, A.; Nunes, C.; and Jatowt, A. 2020. YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences*, 509: 257–289.

Chen, Y.; and Lerch, A. 2020. Melody-conditioned lyrics generation with seqgans. In *2020 IEEE International Symposium on Multimedia (ISM)*, 189–196. IEEE.

Dzhambazov, G.; et al. 2017. *Knowledge-based probabilistic modeling for tracking lyrics in music audio signals*. Ph.D. thesis, Universitat Pompeu Fabra.

Edensor, T. 2020. *National identity, popular culture and everyday life*. Routledge.

Fan, A.; Lewis, M.; and Dauphin, Y. 2018. Hierarchical neural story generation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Guo, F.; Zhang, C.; Zhang, Z.; He, Q.; Zhang, K.; Xie, J.; and Boyd-Graber, J. 2022. Automatic Song Translation for Tonal Languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, 729–743. Dublin, Ireland: Association for Computational Linguistics.

Hono, Y.; Hashimoto, K.; Oura, K.; Nankaku, Y.; and Tokuda, K. 2021. Sinsy: A deep neural network-based singing voice synthesis system. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29: 2803–2815.

Lee, H.-P.; Fang, J.-S.; and Ma, W.-Y. 2019. iComposer: An automatic songwriting system for Chinese popular music. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 84–88.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics,*.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2016. A diversity-promoting objective function for neural conversation models. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Nichols, E.; Morris, D.; Basu, S.; and Raphael, C. 2009. Relationships between lyrics and melody in popular music. In *ISMIR 2009-Proceedings of the 11th International Society for Music Information Retrieval Conference*, 471–476.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Sheng, Z.; Song, K.; Tan, X.; Ren, Y.; Ye, W.; Zhang, S.; and Qin, T. 2021. Songmass: Automatic song writing with pre-training and alignment constraint. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 13798–13805.

Tan, X.; and Li, X. 2021. A tutorial on AI music composition. In *Proceedings of the 29th ACM international conference on multimedia*, 5678–5680.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Watanabe, K.; Matsubayashi, Y.; Fukayama, S.; Goto, M.; Inui, K.; and Nakano, T. 2018. A melody-conditioned lyrics language model. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 163–172.

Weide, R.; et al. 1998. The Carnegie Mellon pronouncing dictionary. *release 0.6, www. cs. cmu. edu*.

Xue, L.; Song, K.; Wu, D.; Tan, X.; Zhang, N. L.; Qin, T.; Zhang, W.-Q.; and Liu, T.-Y. 2021. DeepRapper: Neural rap generation with rhyme and rhythm modeling. *arXiv preprint arXiv:2107.01875*.

Yao, L.; Peng, N.; Weischedel, R.; Knight, K.; Zhao, D.; and Yan, R. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 7378–7385.

Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Yu, Y.; Srivastava, A.; and Canales, S. 2021. Conditional lstm-gan for melody generation from lyrics. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 17(1): 1–20.

Zhang, C.; Chang, L.; Wu, S.; Tan, X.; Qin, T.; Liu, T.-Y.; and Zhang, K. 2022. ReLyMe: Improving Lyric-to-Melody Generation by Incorporating Lyric-Melody Relationships. In *Proceedings of the 30th ACM International Conference on Multimedia*, 1047–1056.