# Rapformer: Conditional Rap Lyrics Generation with Denoising Autoencoders

**Nikola I. Nikolov[†], Eric Malmi[‡], Curtis G. Northcutt[§], Loreto Parisi[◇]**
[†]Institute of Neuroinformatics, University of Zurich and ETH Zurich
[‡]Google [§]MIT [◇]Musixmatch
niniko@ini.ethz.ch emalmi@google.com
cgn@mit.edu loreto@musixmatch.com

## Abstract

The ability to combine symbols to generate language is a defining characteristic of human intelligence, particularly in the context of artistic story-telling through lyrics. We develop a method for synthesizing a rap verse based on the content of any text (e.g., a news article), or for augmenting pre-existing rap lyrics. Our method, called RAPFORMER, is based on training a Transformer-based denoising autoencoder to reconstruct rap lyrics from content words extracted from the lyrics, trying to preserve the essential meaning, while matching the target style. RAPFORMER features a novel BERT-based paraphrasing scheme for rhyme enhancement which increases the average rhyme density of output lyrics by 10%. Experimental results on three diverse input domains show that RAPFORMER is capable of generating technically fluent verses that offer a good trade-off between content preservation and style transfer. Furthermore, a Turing-test-like experiment reveals that RAPFORMER fools human lyrics experts 25% of the time.[1]

## 1 Introduction

Automatic lyrics generation is a challenging language generation task for any musical genre, requiring story development and creativity while adhering to the structural constraints of song lyrics. Here we focus on the generation of *rap lyrics*, which poses three additional challenges specific to the rap genre: ($i$) a verse in rap lyrics often comprises multiple rhyme structures which may change throughout a verse (Bradley, 2017), ($ii$) the number of words in a typical rap verse is significantly larger when compared to other music genres (Mayer et al., 2008), requiring modeling of long-term dependencies, and ($iii$) the presence of many slang words.

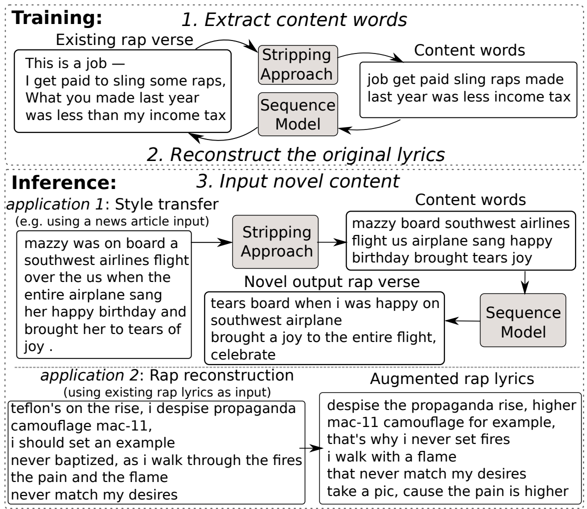


Figure 1: Overview of our approach to *conditional rap lyrics generation*. **Training**: (1) extract content words from existing rap verses, then (2) train sequence models to guess the original verses conditioned on the content words. **Inference**: (3) Input content from non-rap texts to produce *content-controlled* rap verses; or input existing rap verses to augment them.

Prior approaches to rap generation typically use *unconditional* generation (Potash et al., 2015; Malmi et al., 2016). That approach synthesizes lyrics without providing any context that could be useful to guide the narrative development into a coherent direction (Dathathri et al., 2020). For example, generating rap lyrics on a specific topic, e.g., "cooking," is not possible with unconditional generation. Motivated by this, in this paper, we propose a novel approach for *conditional* generation of rap verses, where the generator is provided a source text and tasked with transferring the style of the text into rap lyrics. Compared to unconditional generation, this task can support the human creative process more effectively as it allows a human writer to engage with the generator by providing the content of the lyrics while receiving automatic suggestions on how to improve the style of the lyrics to resemble the rap domain.

[1]We created a song with lyrics generated by RAPFORMER using the abstract of this paper as input, available in the supplementary material, and at https://bit.ly/3kXGItD.



*Proceedings of The 13th International Conference on Natural Language Generation*, pages 360–373,
Dublin, Ireland, 15-18 December, 2020. 

Our approach to conditional generation is to train sequence-to-sequence models (Vaswani et al., 2017) to reconstruct existing rap verses conditioned on a list of content words extracted from the verses (Figure 1). By learning a mapping from content words to complete verses, we implicitly learn the latent structure of rap verses given content, while preserving the target output style of the rap lyrics. Model outputs are enhanced by a post-processing step (Section 3.2) that substitutes non-rhyming end-of-line words with suitable rhyming alternatives.

We test our method on three diverse input domains: short summaries of news articles, movie plot summaries, and existing rap lyrics. Automatic and human evaluations (Sections 5 and 6) suggest that our method provides a trade-off between content preservation and style compared to a strong information retrieval baseline.

## 2 Background

### 2.1 Rap Lyrics Generation

Prior work on rap lyrics generation often focuses on unconditional generation, either using language models (Potash et al., 2015) or by stitching together lines from existing rap lyrics using information retrieval methods (Malmi et al., 2016). There are two main drawbacks of unconditional generation of rap lyrics. First, the open-ended nature of the task is too unconstrained for generating lyrics with more specific content: ideally, we may want to have control over at least some aspects of the model during inference, such as the topic of the lyrics, or their sentiment. Second, although frequent rhyming is an essential feature of fluent rap verses (Malmi et al., 2016), language models have no built-in incentive to learn to consistently generate rhymes at the end of each line, prompting researchers to invent techniques to promote rhyming in their models separately (Hopkins and Kiela, 2017).

More recently, Manjavacas et al. (2019) propose a conditional approach to rap lyrics generation, which extracts high-level features from the lyrics, such as their sentiment, mood, or tense, to provide a template during generation. Although their approach allows for some control during generation, it is limited in terms of generating lyrics with more specific content. The work that is closest to ours is (Lee et al., 2019) who propose an approach to sentence style transfer based on text denoising, and test their approach on style transfer from pop to rap lyrics. In contrast to these works, we condition the model on longer input text and also introduce a novel method for enhancing the rhymes of our output verses. We also perform extensive automatic and human evaluations on style transfer from diverse input domains to rap lyrics.

### 2.2 Text Rewriting and Style Transfer

Recent work on style transfer of text (Fu et al., 2018; Shen et al., 2017; Prabhumoye et al., 2018; Lample et al., 2019; Liu et al., 2019), focuses on transfer from one text attribute to another, such as gender or political inclination. The main difference between such studies and our work is that our setting is more lenient with respect to meaning preservation: our focus here is on generating creative and fluent verses that match the overall topic of the input and also preserve *some* of the content. Our conditional lyrics generation based on denoising autoencoders is also related to recent work on self-supervised pre-training objectives for text-to-text generation tasks, which have been beneficial for many NLP tasks, such as automatic text summarization (Zhang et al., 2020), question answering (Lewis et al., 2020; Raffel et al., 2019), and data-to-text generation (Freitag and Roy, 2018).

## 3 Conditional Generation of Lyrics

Our approach to conditional generation of rap verses consists of three steps (Figure 1).

1. Given a dataset of rap verses, we apply a stripping approach to extract from each verse a set of *content words* that aim to resemble the main content of the original text, omitting any specific stylistic information.

2. We train a Transformer model (Vaswani et al., 2017) to reconstruct the original rap verses conditioned on the content words. The model learns to generate the original verse, filling in missing stylistic information.

3. At inference time, we can input content words extracted from a text written in any style, such as a news article, resulting in novel output rhyme verses. After generation, we optionally apply a rhyme enhancement step (Section 3.2).

### 3.1 Stripping Approach

Given a dataset of original rap verses, our base approach to extracting content words involves preprocessing each verse to remove all stop words[2], numbers, and punctuation. To promote greater novelty[3] and variability in the outputs produced by our models, we additionally apply one of three noise types to the stripped content words:

**Shuffle.** We shuffle all of the content words on the sentence level (line level for rap verses). This type of noise forces our models to learn to rearrange the location of the input content words when generating the output rap lyric, rather than to merely copy words from the input in an identical order. A similar noising approach has been recently employed by Raffel et al. (2019).

**Drop.** We randomly remove 20% of the input content words for the purpose of promoting generation of novel words, rather than only copying content words from the input.

**Synonym.** We replace 20% of the content words with synonyms obtained from WordNet (Miller, 1995). We pick words randomly and replace them with a random synonym. This type of noise promotes our models to learn to replace content words with synonyms, which might fit better in the context or style of the current output rap verse.

### 3.2 Rhyme Enhancement with BERT

To improve the rhyming fluency of our models, we implement a post-processing step for *rhyme enhancement (RE)* which modifies a generated verse to introduce additional end-of-line rhymes. Given two lines from a generated verse, such as:

> *where were* ***you****?*
> *last year i was paid in a drought with no* ***beginners***

RE iterates over each of the lines in the verse, replacing the ending words with a *MASK* token. The verse is then passed through a BERT model[4] (Devlin et al., 2019) which predicts the $K = 200$ most likely replacement candidates for *MASK*. For exam ple, the replacement candidates for *you* might be {*they, we, I, it*}, and for *beginners* might be {*food, fruit, you, rules*}. We pick the candidate that leads to the highest increase in rhyming, determined by the length of the longest overlapping vowels in the two words (Malmi et al., 2016). In the example above, replacing *beginners* with *food* maximizes the rhyme length, and the example becomes:

> *where were you?*
> *last year i was paid in a drought with no* ***food***

Algorithm 1 contains a detailed implementation of our approach.

**Algorithm 1:** Bert Rhyme Enhancement

```
input  : lyrics verse V = {l_0, ..., l_N} consisting of
         N tokenized lines; number of BERT
         predictions K to consider.
output : modified V with enhanced rhyming.

Function get_rhyming_replacement (V,
  src_idx, tgt_idx, mask) :
    src ← V [src_idx][-1] // get last word
    tgt ← V [tgt_idx][-1]
    // Predict most likely words.
    preds ← bert_predictions (mask, K)
    // Compute original rhyme
       length.
    rl_orig ← rhyme_length (src, tgt)
    for pred ∈ preds do
        rl_new ← rhyme_length (pred, tgt)
        if rl_new > rl_orig then
            // return replacement
            return pred, rl_new
    return target, rl_orig // return
       original

for i ← 1, 3, ..., N // for each odd line
 do
    // Create two masks for the two
       consecutive lines.
    mask_1 ← mask_text (V, i)
    mask_2 ← mask_text (V, i + 1)
    // Generate replacement
       candidates.
    cand_1, rl_1 ←
       get_rhyming_replacement (V, i + 1,
       i, mask_1) // replace last word
       at i
    cand_2, rl_2 ←
       get_rhyming_replacement (V, i, i +
       1, mask_2) // replace last word
       at i + 1
    if rl_2 ≥ rl_1 // update lines in V
     then
        V [i + 1][-1] ← cand_2
    else
        V [i ][-1] ← cand_1
return V
```

## 4 Experimental Setup

**Datasets.** We conduct experiments using three datasets. As our rap dataset, we use 60k English rap lyrics provided by Musixmatch.[5]

We split each lyric into verses (in the dataset, each verse is separated by a blank line), remove

[2] We use the list of English stopwords defined in NLTK.

[3] In early experiments, we tested training models using only this base approach. The models performed very well at reconstructing existing rap lyrics, however when the input was from a different domain, we observed very conservative outputs.

[4] We finetune a BERT base model on our rap verse dataset for 20 epochs.

[5] https://www.musixmatch.com/

| | News | Movies | Rap |
|---|---|---|---|
| *# Pairs* | 287k/11k/11k | - / - /12k | 165k/1k/1k |
| *Sent. p.d.* | $3.7 \pm 1.2$ | $3.9 \pm 1.6$ | $10.5 \pm 4.5$ |
| *Tok. p.d.* | $57.9 \pm 24.3$ | $90 \pm 27.6$ | $91.8 \pm 49.1$ |
| *Tok. p.s.* | $15.1 \pm 4.7$ | $22.4 \pm 11$ | $9.5 \pm 4.25$ |

Table 1: Statistics of our datasets. *# Pairs* denotes the number of pairs used for training/validation/testing; *p.d.* is per document; *p.s.* is per sentence.

verses shorter than 4 lines in order to filter for song choruses and intros, and reserve 2k song lyrics for validation and testing. We use two datasets as our out-of-domain inputs: (1) the summaries from the CNN/DailyMail news summarization dataset (Hermann et al., 2015) and (2) a subset of the CMU movie plot summary corpus (Bamman et al., 2013). Since some of the movie summaries are very long, for this dataset, we filter summaries longer than 140 tokens and shorter than 40 tokens. Table 1 contains detailed statistics of the datasets used for training/validation/testing in our experiments.

**Model details.** As our sequence transducer, we use a 6-layer Transformer encoder-decoder model (Vaswani et al., 2017). We initially train our models on the source domain (e.g., news articles) for 20 epochs, after which we finetune them on rap verses for an additional 20 epochs, using the same stripping approach for both. We train all of our models on the subword level (Sennrich et al., 2016), extracting a common vocabulary of 50k tokens from a joint collection of news summaries and rap lyrics. We use the same vocabulary for both our encoders and decoders and use the Fairseq library.[6] We train all of our models on a single GTX 1080 Ti card.

**Generation details.** During inference, we generate outputs using diverse beam search (Vijayakumar et al., 2018) to promote greater diversity across the hypothesis space. We use a beam with a size of 24 and 6 diverse beam groups. Furthermore, we limit the maximum output sequence length to two times the length of the input content words and penalize repetitions of bigrams in the outputs.

To select our final output, we additionally implement a simple hypothesis reranking method. For each of the 24 final predictions on the beam, we compute two scores: the rhyme density ($RD$) of the text, following (Malmi et al., 2016), as well as its repetition score:

$$rep(\mathbf{s}) = \frac{\sum_i overlap(\overline{\mathbf{s}_i}, s_i)}{|\mathbf{s}|}. \qquad (1)$$

$rep$ measures the average unigram overlap (see Section 5.1) of each sentence $s_i$ in the text $\mathbf{s}$ with all other sentences of the text concatenated into a single string (denoted as $\overline{\mathbf{s}_i}$). We pick the hypothesis that maximizes: $score(\mathbf{s}) = RD(\mathbf{s}) - rep(\mathbf{s})$. Afterwards, we optionally apply our rhyme enhancement step, to further increase the frequency of rhymes in our outputs.

**Bias mitigation** Rap lyrics, like other human-produced texts, may contain harmful biases and offensive content which text generation models should not propagate further. Our conditional lyrics generation setup is less susceptible to this issue since the user provides the content, and the generator is supposed to modify only the style of the text. Yet, the model may learn to use inappropriate individual terms that are common in rap lyrics. To alleviate this, we maintain a deny list of words that the model is not able to generate.

## 5 Automatic Evaluation

We conduct an automatic evaluation of RAPFORMER, using the test sets of each of our three datasets. Our focus is on measuring two components that are important for generating fluent conditional rap verses: preserving content from the input text to the output, and maintaining rhyming fluency during generation.

### 5.1 Evaluation Metrics

**Content preservation.** We test the capacity of our models to preserve content words from the input by computing a unigram overlap score:

$$overlap(\mathbf{x}, \mathbf{y}) = \frac{|\{\mathbf{y}\} \cap \{\mathbf{x}\}|}{|\{\mathbf{y}\}|} \qquad (2)$$

between unique unigrams from an input text $\mathbf{x}$ and the generated output rap verse $\mathbf{y}$. We also report the BLEU score (Papineni et al., 2002) when training a model to reconstruct original lyrics.

**Rhyming fluency.** We measure the technical quality of our rap verses using the rhyme density (RD) metric (Malmi et al., 2016).[7] The metric is based on computing a phonetic transcription of the

[6] https://github.com/pytorch/fairseq

[7] https://github.com/ekQ/raplysaattori

| | Model | Rap reconstruction | | | Style transfer from movies | | Style transfer from news | |
|---|---|---|---|---|---|---|---|---|
| | | BLEU | Overlap | RD | Overlap | RD | Overlap | RD |
| | INPUTS | - | - | 0.84 ± 0.38 | - | 0.73 ± 0.2 | - | 0.72 ± 0.21 |
| | IR NEWS | - | - | - | - | - | 0.29 ± 0.09 | 0.74 ± 0.19 |
| | IR RAP | - | - | - | 0.19 ± 0.06 | **1.02 ± 0.23** | 0.17 ± 0.06 | 1.01 ± 0.24 |
| RAPFORMER | SHUFFLE | 10.27 | **0.63 ± 0.13** | 1.01 ± 0.31 | **0.51 ± 0.11** | 0.90 ± 0.23 | **0.45 ± 0.12** | 0.89 ± 0.26 |
| | SHUFFLE + RE | 12.72 | 0.60 ± 0.12 | 1.10 ± 0.32 | 0.49 ± 0.10 | 0.96 ± 0.27 | 0.43 ± 0.11 | 0.98 ± 0.27 |
| | DROP | 11.06 | 0.52 ± 0.11 | 1.03 ± 0.32 | 0.43 ± 0.10 | 0.90 ± 0.24 | 0.38 ± 0.10 | 0.93 ± 0.25 |
| | DROP + RE | 09.81 | 0.50 ± 0.11 | **1.13 ± 0.33** | 0.40 ± 0.09 | 0.99 ± 0.27 | 0.36 ± 0.10 | 1.03 ± 0.26 |
| | REPLACE | **14.30** | 0.57 ± 0.15 | 1.00 ± 0.30 | 0.43 ± 0.14 | 0.86 ± 0.28 | 0.34 ± 0.13 | 0.95 ± 0.27 |
| | REPLACE + RE | 12.72 | 0.54 ± 0.15 | 1.10 ± 0.31 | 0.40 ± 0.13 | 0.98 ± 0.24 | 0.31 ± 0.12 | **1.05 ± 0.28** |

Table 2: Automatic metric results of RAPFORMER, using three alternative stripping approaches: SHUFFLE, DROP and REPLACE. Model names ending in * + RE denote use of the additional rhyme enhancement step (see Section 3.2). INPUT measures the result of the original input texts, for each of the three inputs (rap/movies/news). **Overlap** is the content preservation score, **RD** is the rhyme density metric. The highest results for each column are in bold.

lyrics and finding the average length of matching vowel sound sequences which resemble multisyllabic assonance rhymes. As a reference, RD values above 1 can be considered high, with some rap artists reaching up to 1.2.

### 5.2 Baselines

For reference, we report the result of an information retrieval baseline, which retrieves the closest text from our training dataset given input from the news or movies test sets, using sentence embedding similarity.[8] We report two variants of the IR baseline. First, we retrieve the closest summary from the CNN/DailyMail news training set (IR NEWS), which resembles a lower bound for our target task of style transfer from news to rap lyrics. Second, we retrieve the closest verse from our rap training set (IR RAP). The outputs of the strong IR Rap baseline perfectly match the style of original rap verses, giving us an upper bound for rap style, while maintaining some degree of lexical and semantic overlap with the input texts.

### 5.3 Results

Our results are shown in Table 2, where we include all of our stripping approaches (Shuffle, Drop, Replace). We report the results of applying the additional rhyme enhancement step separately (model names ending with "+ RE").

**Rap reconstruction.** In the left part of Table 2, we evaluate our model's capacity to reliably regenerate original rap lyrics given extracted content words from them. RAPFORMER performed well on this task, generating fluent lyrics that incorporate a large part of the input content words and surpassing the average rhyme density observed in the training dataset (INPUTS). When using our rhyme enhancement step, we observe a slight decrease in overlap due to the potential replacement of content words. However, RD increases by 10% on average.

**Style transfer.** In the right part of Table 2, we evaluate the capacity of our model to generate rap lyrics using content words extracted from movie plot summaries or news article summaries. For these inputs, our model generated outputs with lower overlap on average than for rap reconstruction, with movies retaining slightly more content than news. This gap is potentially due to the large differences in style, vocabulary, and topic of the inputs, prompting our models to ignore some of the content words to better match the target rap style. Still, our generation methods manage to achieve similar RD scores while considerably outperforming the strong IR baseline in terms of overlap.

## 6 Human Evaluation

Due to the limitations of automatic metrics for text generation, we also perform four human evaluation experiments using three raters, who are trained to translate lyrics. Due to limited resources, we evaluate only the RAPFORMER variant with the SHUFFLE stripping approach and rhyme enhancement, which achieved the highest content overlap in our automatic evaluation.

The first two human experiments (in Table 3) focus on style transfer using news articles as input. Each rater inspected 100 verses produced by either the RAPFORMER, or the two IR baselines, answering the following three questions:

1. *How much do the lyrics presented resemble rap lyrics? On a scale from 1 (not at all),*

[8] We use a 600-dimensional Sent2Vec model (Pagliardini et al., 2018), which is pretrained on Wikipedia.

| Method | Style | Meaning | Familiarity |
|---|---|---|---|
| IR NEWS | 1.18 | 2.01 | 1% |
| IR RAP | 4.27 | 1.33 | 31% |
| RAPFORMER | 2.03 | 2.55 | 8% |

Table 3: Human evaluation results of RAPFORMER (using the SHUFFLE stripping approach, and news articles as input). The average inter-rater agreement for **Style** is 0.3, and for **Meaning** is −0.1, measured using Cohen's Kappa (Cohen, 1960).

*to 5 (this could be from existing rap lyrics)*, which measures the capacity of our models to preserve the **Style**.

2. *How well do the lyrics preserve the content of the original news article on a scale from 1 (not at all) to 5 (very well)?* This question measures the meaning preservation of our models (**Meaning**).

3. *Do these lyrics look like a song you know (yes or no)?* For IR RAP, this question measures the **Familiarity** of the raters with the lyrics; for the other two methods, it measures the capacity to fool the raters.

| Method | Side-by-Side | Random |
|---|---|---|
| RAPFORMER | 7% | 25% |

Table 4: Turing-like evaluation, reporting the percentage of lyrics generated by RAPFORMER (using the SHUFFLE stripping approach, and rap lyrics as input) that human experts incorrectly label as existing rap lyrics. The average inter-rater agreement for **Side-by-Side** is 0.8, and for **Random** is 0.4, measured using Cohen's Kappa (Cohen, 1960).

The other two human experiments (in Table 4) focus on our rap reconstruction task, performing two Turing-test-like comparisons between 100 real and synthetic verses:

1. **Side-by-Side**: the original rap lyrics and RAPFORMER lyrics are presented side-by-side, in a random order, and a rater is asked, *Which of these lyrics was written by a human?* (see the Appendix for examples).

2. **Random**: a verse is shown and the rater is asked, *"Do you think these rap lyrics are: (a) AI-generated or (b) human-created?"*.

In terms of style (Table 3), we outperform IR NEWS, demonstrating that there is a change in style

INPUT (RAP): before the dough came , my whole aim , was blow like propane
control the whole domain , and then show no shame
make rappers go ? and they so lame , playing with no game
put em on the lil plane til they can 't claim no fame
i got , the range , better , stay in the slow lane
i make the flow change from hurricanes to a slow rain
your thoughts are so plane , i rearrange your whole frame
until my whole name grow out your brain like rogaine
...

SHUFFLE: aim dough like propane came with a whole blow
shame how you control the whole show
lame rappers playin' the game, make your domain go
can't claim em til you put a lil fame on a plane
slow down, stay in your lane, got a better range
make it rain, change your flow, slow down
rearrange your whole frame, plane thoughts
grow like rogaine, put my whole name on your brain
...

SHUFFLE + RE: aim dough like propane came with a whole blow
shame how you control the whole show
lame rappers playin' the game, make your domain go
can't claim em til you put a lil fame on a **show**
slow down, stay in your lane, got a better range
make it rain, change your flow, slow **rain**
rearrange your whole frame, plane **change**
grow like rogaine, put my whole name on your brain
...

Table 5: Example model output for rap reconstruction. Words replaced by our rhyme enhancement step are in bold. The input lyrics are from the song *How I Get Down* by Rakim.

towards rap verses. There is still a large gap to reach the fluency of original rap verses retrieved by IR RAP. However, it is worth noting that the content preservation of IR RAP is considerably lower, as shown in Tables 2 and 3, and simply the fact that the content of the generated lyrics is closer to the news domain might encourage the raters to rate the generated lyrics as having a lower rap resemblance score. In other words, the style score of IR RAP might be unrealistic to attain even with a perfect conditional generator.

Overall, the results indicate that our method provides a trade-off between the two baselines in terms of style while outperforming them in terms of content overlap. Furthermore, 8% of the time, our conditional generation model fooled experienced raters to think that our synthetic rap lyrics generated from news articles originate from real rap songs. Our rap lyrics augmentation approach also proved to be robust in the Turing-style evaluation of rap reconstruction (Table 4), where RAPFORMER fooled the raters 25% of the time when lyrics from a random source are presented one-by-one, and 7%

of the time when lyrics are presented side-by-side.

| |
|---|
| INPUT (MOVIES): the film follows the lives of several west point cadet classmates who find themselves on opposite sides of the war . the film also follows the adventures of lucius the slave escaping via the underground railroad to freedom with the film cutting between the first battle of bull run and the birth of a lucius ' child born in slavery . |
| SHUFFLE: this is the opposite of war follows lives on both sides<br>several point film from the west to the wrong<br>find a child born escaping via film<br>film the underground cutting off the film of all the complications<br>slave, run from lucius slavery<br>battle of freedom and birth<br>also the first bull follows luc-up! |
| SHUFFLE + RE: this is the opposite of war follows lives on both sides<br>several point film from the west to the **light**<br>find a child born escaping via **immigration**<br>film the underground cutting off the film of all the complications<br>slave, run from lucius slavery<br>battle of freedom and **liberty**<br>also the first bull follows luc-up! |

Table 6: Example model outputs for style transfer from movie plot summaries. Words replaced by our rhyme enhancement step are in bold.

| |
|---|
| INPUT (NEWS): temperatures dipped into the mid-30s during 4 days man lay in woods of philadelphia park . mom told police son was with her in maryland , but he was found friday with blanket , bible . victim being treated for malnutrition , dehydration ; mother faces host of charges after extradition . |
| SHUFFLE: man i was dipped up in a lay up with some of them from an old<br>mid-30s days in the park<br>mom told me to be in michigan woods<br>police blanket friday<br>i found my son a bible<br>he was a host for the charges<br>my mother treated him as an age<br>a victim of faces |
| SHUFFLE + RE: man i was dipped up in a lay up with some of them from an old<br>mid-30s days in the **home**<br>mom told me to be in michigan anyway<br>police blanket **friday**<br>i found my son a bible<br>he was a host for the **trial**<br>my mother treated him as an **alien**<br>a victim of faces |

Table 7: Example model outputs for style transfer from news articles. Words replaced by our rhyme enhancement step are in bold.

## 7 Example Model Outputs

In Tables 5, 6 and 7, we also display a few manually selected example model outputs (additional examples are available in the Appendix) produced after inputting content words extracted from each of our input text styles (existing rap lyrics, movie plot summaries and news article summaries). When using existing rap lyrics as input, many outputs seem coherent and of higher quality in comparison to outputs produced using news/movie inputs. For news/movie inputs, the models are still capable of integrating the input content words into a rhyming verse that preserves some of the overall meaning of the original text (e.g., "the film also follows the adventures of lucius the slave escaping via the underground railroad to freedom" → "slave, run from lucius slavery; battle of freedom and liberty").

Furthermore, in Table 8 we present examples from our side-by-side Turing test, where we asked raters to choose which of two lyrics was generated (augmented) by RAPFORMER, and which was written by a human. For the selected outputs, two of the three raters incorrectly guessed that the lyrics generated by RAPFORMER were actually human-created.

## 8 Conclusion

We have proposed a novel approach to generation of rap verses conditioned on a list of content words. We showed that our method is capable of generating coherent and technically fluent synthetic verses using diverse text types as input, including news articles, movie plot summaries, or original rap verses. The fluency of our outputs is further improved through a novel rhyme enhancement step. Our approach is particularly effective when rephrasing the content of existing rap lyrics in novel ways, making it a potentially useful tool for creative writers wishing to explore alternative expressions of their ideas.

The generality of our approach to conditional text generation makes it applicable to generation of creative texts in other domains, such as poetry or short stories. Future work could explore other approaches to extracting content words, including combining several stripping approaches, and could explore the utility of large-scale pretrained models (e.g., (Raffel et al., 2019; Lewis et al., 2020)) for this task. Another direction is to extend our

| | |
|---|---|
| *Question 45 of 100*<br>LYRICS (A)<br>waka waka:<br>they say na blind eye, take it far<br>i've got it on my own, my own<br>oche num, oda du, doka dum so<br>if anybody ever try go shoot the almighty<br>blazing so amazing | <br>LYRICS (B)<br>i say na correct eye i take waka this waka<br>but after i've got you i blind pata pata<br>oche du no dum no oda du num doka<br>anybody try you i go shoot the murderfker<br>ever blazing you amazing |
| *Which of these lyrics was written by a human?* | *Correct answer:* (B) |
| *Question 72 of 100*<br>LYRICS (A)<br>vegas on the third floor, like lamar with the cardio<br>fascinated by the cars smokin' dope in the casino<br>despise the propaganda rise, higher<br>mac-11 camouflage for example, that's why i never set fires<br>i walk with a flame that never match my desires<br>take a pic, cause the pain is higher<br>i'm rich as a coupe, light it up with kelly<br>phone sucker, my friend, it's a blessing<br>benz, plaques, wall, and g6's<br>- 'em all, hustler say the victim<br>ciroc and bel air -<br>april -'s -, her name so | <br>LYRICS (B)<br>out in vegas like lamar, third floor tropicana<br>fascinated with the cars, smokin' dope in the phantom<br>teflon's on the rise, i despise propaganda<br>camouflage mac-11, i should set an example<br>never baptized, as i walk through the fires<br>the pain and the flame never match my desires<br>crucified cause i'm rich, in the coupe, take a pic<br>on the phone at the light, kelly rowland's a friend<br>catfish in the benz, manti teo's a sucker<br>plaques on the wall, hustler so i can say "- 'em"<br>bel air for the -, ciroc in the pool<br>my - is a -, her name is april's a fool |
| *Which of these lyrics was written by a human?* | *Correct answer:* (B) |
| *Question 74 of 100*<br>LYRICS (A)<br>she cut the call when she was on ma phone<br>when you picked up the line<br>you got so mad and asked me who's the girl<br>i'm sleeping with behind<br>baby, i had no words to say<br>so i guess i will try<br>not to lie... it's the time... | <br>LYRICS (B)<br>i picked up the phone and cut the line and call<br>i asked what's up girl, why you got so long<br>i'm sleeping behind you<br>baby, i guess i try to say the truth<br>but... it's time to lie... |
| *Which of these lyrics was written by a human?* | *Correct answer:* (A) |

Table 8: Examples of lyrics generated by RAPFORMER that fooled the majority (at least two out of three) human raters in a side-by-side comparison with human created lyrics. Inappropriate words are replaced by a single dash.

work to end-to-end generation with an integrated rhyming loss function, which could potentially be tackled using reinforcement learning (Luo et al., 2019). Moreover, the task of generating coherent lyrics from a set of content words could be naturally modeled as a text-editing task (Dong et al., 2019; Mallinson et al., 2020; Malmi et al., 2019) instead of a sequence-to-sequence task.

## Acknowledgements

We are thankful to Alessandro Calmanovici, Scott Roy, Aliaksei Severyn, and Ada Wan for useful discussions. We also thank Simone Francia and Maria Stella Tavella from Musixmatch, for technical help, and the three raters, for participating in the human evaluation.

## References

David Bamman, Brendan O'Connor, and Noah A. Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria. Association for Computational Linguistics.

Adam Bradley. 2017. *Book of rhymes: The poetics of hip hop*. Civitas Books.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. 2020. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*.

Yue Dong, Zichao Li, Mehdi Rezagholizadeh, and Jackie Chi Kit Cheung. 2019. Editnts: An neural programmer-interpreter model for sentence simplification through explicit editing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3393–3402.

Markus Freitag and Scott Roy. 2018. Unsupervised natural language generation with denoising autoencoders. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Jack Hopkins and Douwe Kiela. 2017. Automatically generating rhythmic verse with neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 168–178.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In *ICLR 2019*.

Joseph Lee, Ziang Xie, Cindy Wang, Max Drach, Dan Jurafsky, and Andrew Y Ng. 2019. Neural text style transfer via denoising and reranking. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 74–81.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Dayiheng Liu, Jie Fu, Yidan Zhang, Chris Pal, and Jiancheng Lv. 2019. Revision in continuous space: Unsupervised text style transfer without adversarial learning. *arXiv preprint arXiv:1905.12304*.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. A dual reinforcement learning framework for unsupervised text style transfer. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019*.

Jonathan Mallinson, Aliaksei Severyn, Eric Malmi, and Guillermo Garrido. 2020. Felix: Flexible text editing through tagging and insertion. *arXiv preprint arXiv:2003.10687*.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5057–5068.

Eric Malmi, Pyry Takala, Hannu Toivonen, Tapani Raiko, and Aristides Gionis. 2016. Dopelearning: A computational approach to rap lyrics generation. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 195–204. ACM.

Enrique Manjavacas, Mike Kestemont, and Folgert Karsdorp. 2019. Generation of hip-hop lyrics with hierarchical modeling and conditional templates. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 301–310.

Rudolf Mayer, Robert Neumayer, and Andreas Rauber. 2008. Rhyme and style features for musical genre classification by song lyrics. In *Proceedings of the 9th International Conference on Music Information Retrieval*, pages 337–342.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. In *NAACL 2018 - Conference of the North American Chapter of the Association for Computational Linguistics*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2015. Ghostwriter: Using an lstm for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876, Melbourne, Australia. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, pages 6830–6841.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search: Decoding diverse solutions from neural sequence models. *AAAI 2018*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *arXiv preprint arXiv:1912.08777, ICML 2020*.

# A Additional Model Outputs

In Tables 9, 10 and 11 we display a few additional manually selected model outputs for each of our input domains (rap lyrics, movie summaries and news article summaries) and each of our stripping approaches (SHUFFLE (RAPFORMER), DROP, and SYNONYM).

# B Demo Song

We generated lyrics for a demo song by using the abstract of this paper as the input to RAPFORMER. We generated multiple samples, by reshuffling the content words of the abstract multiple times. We sent all sample lyrics to a rap artist, and asked them to record a song using a subset of those lyrics. We allowed for re-arranging and deletion, but no addition of human-created lyrics. The resulting audio file is included in the supplementary material [9], while the final lyrics of the song are in Table 12.

We also tested the recently released Jukebox algorithm (Dhariwal et al., 2020) for end-to-end synthesis of a rap song conditioned on the abstract lyrics. However, our preliminary results were unsatisfactory since it was impossible to tell individual words apart from the generated audio.

[9] Also available at `https://bit.ly/3kXGItD`.

| |
|---|
| INPUT: control the whole domain , and then show no shame<br>make rappers go ? and they so lame , playing with no game<br>put em on the lil plane til they can 't claim no fame<br>i got , the range , better , stay in the slow lane<br>i make the flow change from hurricanes to a slow rain<br>your thoughts are so plane , i rearrange your whole frame<br>until my whole name grow out your brain like rogaine<br>... |
| DROP + RE:<br>like propane in control, my whole objective show<br>no shame, i can make it playin' the game<br>put em in a lil' game, til they don't claim fame range<br>you better stay in your lane, and make the flow change<br>i be on a plane to rearrange your own frame<br>your whole name grow in my brain, rogaine letter |
| SYNONYM + RE: control the whole domain just to show it no shame<br>make these rappers go lame, they playin' the game<br>put em on a lil' plane til they can't title the fame<br>if you got a range, you better stay in a slow lane<br>name run change hurricane slow rain<br>thoughts on the ground and rearrange your whole frame<br>whole name grow brain like a rogaine cane |
| SHUFFLE + RE (RAPFORMER): aim dough like propane came with a whole blow<br>shame how you control the whole show<br>lame rappers playin' the game, make your domain go<br>can't claim em til you put a lil fame on a show<br>slow down, stay in your lane, got a better range<br>make it rain, change your flow, slow rain<br>rearrange your whole frame, plane change<br>grow like rogaine, put my whole name on your brain<br>... |
| INPUT: now comes our turn<br>we step in, you burn<br>you got something to learn<br>and bring a breath of fresh air<br>this is how we go<br>you check this mode<br>ride da best you know<br>and still want more |
| DROP + RE: when it comes to turn to step<br>i got something to bring in my breath<br>go check my mode<br>ride with da so<br>you know i still want to be your own |
| SYNONYM + RE: when it comes to turn my step to burn<br>i got something to learn<br>bring your breath to new light<br>go check the way i ride<br>da best i know<br>you still want so |
| SHUFFLE + RE (RAPFORMER): when it comes to my turn<br>step up and learn something to burn<br>i got fresh air on my breath<br>go check da best mode, bring it yeah<br>you know i still want a ride |

Table 9: Additional model outputs for rap reconstruction.

| |
|---|
| INPUT: hoping to improve his financial lot , petty thief hawk chovinski hires a dancing instructor to teach him how to bear himself like a gentleman . his lessons completed , hawk then poses as a european nobleman , intending to trap a wealthy wife . yolande cowles sees through hawk ’ s pose but falls in love with him anyway . |
| DROP + RE: i improve a grizzly lot of petty thief times<br>dancing in the middle of the night<br>i am the man who can teach you how to bear it<br>like a gentleman with diamonds<br>i’m a superheroic, i can be your wife<br>yolande cowles tonight<br>falls in love anyway |
| SYNONYM + RE: hoping that you can improve<br>a financial lot of petty use<br>mortarboard chovinski engage<br>dancing with the snake<br>teach her how to settle<br>like a gentleman<br>lessons are shackled by a bullet<br>sit in european<br>imagine<br>in the trap with a wealthy wife<br>yolande hood sees the sky<br>when the pose falls in line<br>anyway, no, not me |
| SHUFFLE + RE (RAPFORMER): you teach me petty dancing like bear thief<br>chovinski, intersect, be<br>a lot of financial gentleman hoping he can improve somebody<br>wife, nobleman, the trap is so polished<br>wealthy hawk lessons, european hawk lessons<br>yolande cowles anyway, sees him pose when he says<br>hawk love! |
| INPUT: the film follows the lives of several west point cadet classmates who find themselves on opposite sides of the war . the film also follows the adventures of lucius the slave escaping via the underground railroad to freedom with the film cutting between the first battle of bull run and the birth of a lucius ’ child born in slavery . |
| DROP + RE: film of the west point where they can find the opposite sides of ours<br>film also and they will be a slave escaping me from the underground,<br>and we will not be the same if we are not the maker<br>this is a film cutting first bull from birth to child’s slaver. |
| SYNONYM + RE: film to succeed our lives in several zones<br>our head is the most likely to find our own<br>we are not the same as the other side of ever<br>film also follows adventure<br>the lucius slave, the escaping via underground<br>motorical, freedom, film out<br>first battle bull, then feed him birth<br>golden child, born in order |
| SHUFFLE + RE (RAPFORMER): this is the opposite of war follows lives on both sides<br>several point film from the west to the light<br>find a child born escaping via immigration<br>film the underground cutting off the film of all the complications<br>slave, run from lucius slavery<br>battle of freedom and liberty<br>also the first bull follows luc-up! |

Table 10: Additional model outputs for style transfer from movie plot summaries to rap lyrics.

| |
|---|
| Input (news): temperatures dipped into the mid-30s during 4 days man lay in woods of philadelphia park . mom told police son was with her in maryland , but he was found friday with blanket , bible . victim being treated for malnutrition , dehydration ; mother faces host of charges after extradition . |
| Drop + RE: i've been dipped for days, lay in woods<br>in the park with the crook<br>my son from pittsburgh found friday blanket, bible victim<br>i was born to be a victim of my reality<br>with no faces<br>host charges, i had to do it everything |
| Synonym + RE: dipped in mid-a.t. days, man i dwell in ways<br>UNK 3: gs, park mom told the police son that he was from illegal<br>found him on friday, he had a blanket with a bible<br>a victim of how he treated him as a |
| Shuffle + RE (Rapformer): man i was dipped up in a lay up with some of them from an old<br>mid-30s days in the home<br>mom told me to be in michigan anyway<br>police blanket friday<br>i found my son a bible<br>he was a host for the trial<br>my mother treated him as an alien<br>a victim of faces |
| Input (news): jordan spieth won the 2015 masters by four shots on sunday . the 21-year-old american led all week at the augusta national golf club . he shot final-round 70 to finish on 18 under par and take the green jacket . |
| Drop + RE: to all of my masters,<br>four sunday morning, american led alert<br>national golf club, final-round time<br>take a green jacket |
| Synonym + RE: jordan, we are not the same, no masters!<br>four shots of the sun, the laughter<br>we were the most likely american led in a week<br>at the first club shot last finish, hey<br>get the green cap |
| Shuffle + RE (Rapformer): masters, four shots on sunday<br>jordan, led me to the national club, the american way<br>golf week, green dine, par<br>finish my jacket, take my final-round start |
| Input (news): the dallas native will play alongside justin rose in the final pairing . has set a scoring record for the first 54 holes of 16 under par . finished runner-up last year and is now determined to win . is first player since greg norman in 1996 to have lead after each round . |
| Drop + RE: dallas native play<br>i was born to be a slave<br>but now i'm on my own<br>and i'll be the first so<br>justin final scoring holes in par<br>last year determined to start<br>been a player, since greg<br>the only way to tell |
| Synonym + RE: dallas, c4, i play with the same<br>g6, justin rose to the place<br>c1, ready to scoring the record<br>first holes in the firearm, then i remember<br>this is the first year<br>determined to win, first player, since marc ellen went here |
| Shuffle + RE (Rapformer): justin rose, native gold<br>final par, scoring holes, set it off, play it again, justin rose<br>determined to win the first record, last year i was finished<br>greg player, he was a player from the beginning<br>since first i lead the worldball. |

Table 11: Additional model outputs for style transfer from news articles to rap lyrics.

| |
|---|
| [intro]<br>i am the oldest<br>the lyrics they just follow orders.<br>i am the oldest<br>the lyrics they just follow orders.<br>good trade-off of your style.<br>i am the oldest<br>the lyrics they just follow orders.<br>i rhyme more rhymes and moreover<br>move over I'm recording<br><br>[verse 1]<br>another verse written on the news of rap methods,<br>given to me in the form of an autoencoder<br>to develop the words that i rap, and i will be denoting<br>in my text, i am the only content,<br>i can be the same as an automatist,<br>i train rap lyrics to study different meaning when i approach words as i am,<br>I train lyrics that are the most definitive,<br>more essential than a scheme of three<br>more untouchable than an underflow<br>move over. pirana, the founder, moreover.<br>my rhyme lyrics are more than the rhyme over<br>(when i develop a verse)<br><br>[verse 2]<br>when i develop a verse i form a text from an art that is written on the news of an autoencoder rap<br>another method given to a train that i have been through and i am not the only thing to do with<br>this is my reality<br>i will not be content with rap lyrics i approach with the meaning oh<br>my words are based on my attack.<br>my lyrics are essential as I generate rap.<br>my average rhyme scheme is to show you different content<br>in other words, i can't study my own admirations.<br>my raps are so amazing<br>the rhyme is paraphrasing.<br><br>[bridge]<br>my results are very good like I'm a human being<br>my rap is in the convoy.<br>your lyrics will be so pre-dated.<br>(when i develop a verse)<br><br>[outro]<br>I'm a human being<br>I'm a human being |

Table 12: Lyrics of our demo song, described in Appendix B.