Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Christian Ziegenhahn Jensen
Espen Sørhaug

The Perfect Rap Lyrics

AI Generated Rap Lyrics That Are Better Than Lyrics from Existing Popular and Critically Acclaimed Rap Songs

Master's thesis in Computer Science
Supervisor: Gambäck, Björn
June 2021

NTNU
Norwegian University of
Science and Technology

Christian Ziegenhahn Jensen
Espen Sørhaug

The Perfect Rap Lyrics

AI Generated Rap Lyrics That Are Better Than Lyrics from Existing Popular and Critically Acclaimed Rap Songs

Master's thesis in Computer Science
Supervisor: Gambäck, Björn
June 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

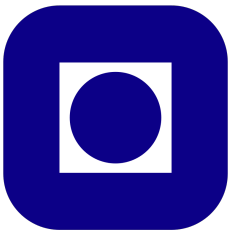
Christian Ziegenhahn Jensen & Espen Sørhaug

The Perfect Rap Lyrics

AI Generated Rap Lyrics That Are Better Than Lyrics from Existing Popular and Critically Acclaimed Rap Songs

Master's Thesis in Computer Science, Spring 2021

Data and Artificial Intelligence Group
Department of Computer Science
Faculty of Information Technology and Electrical Engineering
Norwegian University of Science and Technology



Abstract

The objective of developing computational systems exhibiting creative behaviours has been described as the final frontier in artificial intelligence. With the emergence of ever more sophisticated systems for generation of natural language, the opportunity arises of generating lyrics within a given music genre that is comparable to existing, lyrics written by humans. This thesis offers an exploration of the intersection between rap lyrics and artificial intelligence, with a bipartite focus on research on lyrics analysis and lyrics generation.

On the subject of lyrics analysis, the research lead to a framework for determining rhyme complexity of lyrics. When comparing the calculated rhyme complexity of rap lyrics to the popularity and average score given by critics for the respective songs, a decisive correlation was revealed between rhyme complexity and critics' score, as well as an inverse correlation between rhyme complexity and popularity.

The rap lyrics generation lead to a series of generated rap phrases that were evaluated by quantitative human evaluation as well as the aforementioned framework for assessing rhyme complexity. When assessed by humans, the generated phrases did not score higher than existing lyrics in any of the metrics that were measured; however, in some instances the generated phrases appear to be indistinguishable from human generated lyrics.

As there currently exists no commonly used universal framework for overall rhyme complexity that rewards different types of rhymes, the main contributions of this thesis are the work on the framework for determining rhyme complexity in lyrics, as well as the generation of rap lyrics through artificial intelligence. The validity and potential of this framework is particularly pertinent when comparing results from the calculated rhyme complexity with quantitative human evaluation of perceived rhyme complexity. On the generative side, an artificially intelligent software system that generates rap phrases that are indistinguishable from human written lyrics is regarded as a contribution to the fields of natural language processing and computational creativity.

Sammendrag

Det å utvikle datamaskinelle systemer som utviser en form for kreativitet har lenge vært ansett som en av de største utfordringene innenfor kunstig intelligens. Det kommer stadige nyvinninger innenfor feltet språkbehandling og mer sofistikerte systemer for å emulere naturlig språk. Disse fremskrittene åpner opp for muligheten til å bruke kunstig intelligens til å generere sangtekst innenfor en gitt sjanger, som kan måle seg med sangtekster skrevet av mennesker. Denne oppgaven omhandler således grensesnittet mellom kunstig intelligens og rap-tekster, med et todelt fokus på sangtekstanalyse og sangtekstgenerering.

Hva angår sangtekstanalysen, munnet dette ut i et rammeverk for å vurdere rimkompleksitet i sangtekst. Når man sammenligner denne utregnede rimkompleksiteten til sangtekster med aggregert kritiker-score og populariteten til sangene som teksten kommer fra, ser man en tydelig korrelasjon mellom rimkompleksitet og kritiker-score, så vel som en negativ korrelasjon mellom rimkompleksitet og popularitet.

Tekstgenereringen endte med et sett av genererte rap-strofer som ble evaluert både gjennom kvantitativ menneskelig evaluering og det ovenfor nevnte rammeverket for vurdering av rimkompleksitet. Etter menneskelig vurdering kom det frem at rap-strofene ikke blir rangert høyere enn eksisterende rap-strofer på noen av metrikkene som ble målt, men i noen av tilfellene oppfattes den genererte teksten som uatskillelig fra tekst skrevet av mennesker.

Siden det til dags dato ikke finnes noe universelt rammeverk for vurdering av rimkompleksitet, blir arbeidet som er gjort med rammeverket her ansett som et av hovedbidragene for oppgaven. Gyldigheten og potensialet for rammeverket er av særlig interesse når det sidestilles ved menneskelig evaluering av rimkompleksitet. Videre blir programvaresystemet som bruker kunstig intelligens for å generere rap-strofer som er uatskillelig fra menneskeskrevede sangtekster å anse som et skritt i riktig retning for språkbehandling og datamaskinell kreativitet.

Preface

This thesis was conducted during the spring semester of 2021 as part of our *Master of Science* (MSc) thesis in Computer Science at the *Department of Computer Science* at the *Norwegian University of Technology and Science* (NTNU). It was supervised by Professor Björn Gambäck, and we would like to offer our sincere gratitude for his guidance throughout the course of this thesis. A special acknowledgement should also be given to NTNU's High Performance Computing Group and their IDUN system. Lastly, we would like to offer our gratitude to Even Glad Sørhaug for his assistance in proof reading and F. Paupier for creating the rap lyrics dataset used in the thesis.

Christian Ziegenhahn Jensen & Espen Sørhaug
Trondheim, 11th June 2021

"Not until a machine can write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain"

- Geoffrey Jefferson, 1949

Contents

1 Introduction **1**
- 1.1 Background and Motivation . . . 1
- 1.2 Goals and Research Questions . . . 2
- 1.3 Research Method . . . 3
- 1.4 Contributions . . . 4
- 1.5 Thesis Structure . . . 4

2 Background **5**
- 2.1 Hip Hop Theory . . . 5
- 2.2 Text Mining and Lyrics Analysis . . . 7
- 2.3 Natural Text Generation . . . 9
- 2.4 Artificial Intelligence and Neural Networks . . . 10
- 2.5 Computational Creativity . . . 13

3 Related Work **15**
- 3.1 Lyrics Analysis . . . 15
- 3.2 Lyrical Text Generation and the Aspect of Computational Creativity . . . 16
- 3.3 Systems for Generation of Hip Hop Lyrics . . . 19
- 3.4 Emerging Approaches and State-of-the-Art Systems for Text Generation . 21

4 Architecture **23**
- 4.1 Lyrics Analysis . . . 23
- 4.2 Lyrics Generation . . . 29
- 4.3 Dataset . . . 29

5 Experiment 1: Rhyme Complexity in Rap Lyrics **31**
- 5.1 Research Method and Description . . . 31
- 5.2 Results of Experiment 1 - Rhyme Complexity . . . 32

6 Experiment 2: AI Generation of Rap Lyrics **41**
- 6.1 Setup of Experiment and Lyrics Generation . . . 41
- 6.2 Research Method . . . 42
- 6.3 Result of Experiment 2 - Hip Hop Lyrics Generation . . . 44

7 Discussion and Evaluation **53**
- 7.1 Lyrics Analysis . . . 53
- 7.2 Rhyme Metrics and Rhyme Complexity . . . 55

7.3 Lyrics Generation and Evaluation of Generated Phrases 58
7.4 Survey and Findings from Human Evaluation 60
7.5 Inference of Results for Both Experiments 65

8 Conclusion and Future Work **69**
8.1 Conclusion to Lyrics Analysis and Rhyme Complexity 69
8.2 Conclusion to Lyrics Generation 70
8.3 Future Work 73

Bibliography **75**

Appendix A - Term Frequency Lyrics Catalog **81**

Appendix B - Survey for Evaluation of Rap Phrases **83**

Appendix C - Rap Phrases Used in Survey **89**

Appendix D - Participants in Survey **93**

Appendix E - Correlation Between General Quality and Other Metrics **95**

Appendix F - Dispersion of responses **97**

Appendix G - Results of Survey **105**

Appendix H - Perception of Lyrics Being AI Generated **107**

1 Introduction

Hip hop emerged as a musical genre in the Bronx, New York in the 1970s, where it permeated as a musical expression of oppression in a period plagued by unemployment, drugs and poverty [Rivas, 2020]. It rose continuously in popularity over the following decades and in 2017 it had become the most consumed musical genre in the US [Nielsen Music, 2017].

This rapid rise in popularity brings to the surface a myriad of questions; What makes a musical genre ridden with themes of poverty and oppression so appealing to the general public? Is it possible to quantify what makes some rap songs popular, while others are relegated to rap music oblivion? Is it possible to determine what characteristics are prevalent in rap lyrics of varying popularity and critical acclaim? Would it be possible to recreate the success of popular rap lyrics through the use of *Artificial Intelligence* (AI)? Furthermore, in extension of the previous questions, would it be possible to develop a software system that generates *the perfect rap lyrics*?

Over the course of this thesis there will be a bipartite focus on rap lyrics analysis, and rap lyrics generation. Firstly, analysis of rhyme structure and complexity of rap lyrics from different ends of the spectrum with regards to critical reception and popularity will be conducted. In turn, this will aid in generating rap lyrics phrases, and quantitative evaluation will eventually lead to a conclusion on whether or not objective of using AI to generate the perfect rap lyrics was successful.

1.1 Background and Motivation

Generating natural language convincingly has been regarded by many as one of the foremost challenges within the field of machine learning and *Natural Language Processing* (NLP). With the emergence of recent state-of-the-art text generation systems like the *General Pre-Training* systems GPT-2 and GPT-3, it is possible to generate entire paragraphs of text from just a few words of input [Brown et al., 2020, Radford et al., 2019]. Would it in turn be possible to convincingly generate lyrics that are better than lyrics from popular and critically acclaimed rap songs?

If this task is successful, it would offer a brazen demonstration of the capabilities of both NLP and the field of *Computational Creativity* (CC). In the same vain, it would be of interest to be able to determine what separates lyrics of popular and critically acclaimed songs from unpopular and critically despised songs, to better explain what makes rap lyrics *good*.

1.2 Goals and Research Questions

Firstly, the overarching goal of this project concerns the analysis and generation of hip hop lyrics. The main objective is stated below, and is subsequently divided into more concrete *Research Questions* (**RQ**)s.

Goal 1 *Analyze rap lyrics to discern what separates lyrics of popular and critically acclaimed songs from unpopular and critically despised songs.*

Goal 2 *Develop an AI driven software system that generates rap lyrics phrases that are better than lyrics from existing popular and critically acclaimed rap songs.*

The overarching goals, as stated above, are to develop a system that analyzes rap lyrics to discern patterns in lyrics from rap songs, and subsequently develop a system that generates rap lyrics that are better than lyrics from popular and critically acclaimed songs within the genre. While the title of this thesis shamelessly flaunts the word "perfect" with regards to the generated lyrics, the goal specifies this in more tangible terms as being "as good" as existing lyrics. What constitutes good lyrics is not as easy of a definition, so to specify the merits of this elusive measure, two **RQ** are in place, and subsequently divided into more precise sub-questions.

The approach to achieve this goal will be two-fold, as to be able to generate good rap lyrics one must first discern what constitutes good rap lyrics. Therefore, the first part will be dedicated to addressing the *analysis* of existing rap lyrics, while the subsequent part addresses the aspect of *generation* of new phrases of lyrics. This distinction will be relevant to bare in mind, as these distinct parts of the system will be referred to as the *lyrics analysis* and the *lyrics generation* over the course of this thesis. **RQ 1** addresses directly the objective of lyrics analysis:

RQ 1 *Is it possible to determine what separates lyrics of popular and critically acclaimed rap songs from lyrics of unpopular and critically despised songs?*

- **RQ 1.1** *Is it possible to utilize statistics to identify patterns that are used in the lyrics of rap songs with different degrees of popularity?*
- **RQ 1.2** *Is it possible to utilize statistics to identify patterns that are used in the lyrics of rap songs with different degrees of critical acclaim?*

There may be lots of factors that determine what makes a song popular and critically acclaimed. While overlooking, audio, visuals, notoriety and other factors, and simply inspecting qualities of the lyrics will not paint a complete picture of what makes certain hip hop artists succeed and others not, it may still provide some valuable insight into what sort of lyrics people respond to more favorably, critics and consumers alike. For the scope of this thesis the specific aspect of lyrics that will be analyzed is rhyme structure and complexity of rhymes. This will in turn help inform the generative system, and aid in the process of generating better rap lyrics, that is, lyrics that display some of the qualities more common in successful songs with regards to popularity and critical reception. While **RQ 1** addresses this analysis, **RQ 2** concerns the aspect of lyrics generation.

RQ 2 *Is it possible to generate rap phrases using AI that is better than lyrics from existing popular and critically acclaimed rap songs?*

RQ 2.1 *Will the generated lyrics score highly on metrics defined for evaluating rap lyrics, including findings during rap lyrics analysis (**RQ 1**)?*

RQ 2.2 *Will the generated lyrics be perceived as better than lyrics from existing popular and critically acclaimed rap songs in human evaluation?*

RQ 2.3 *Will the generated lyrics be indistinguishable from human generated rap lyrics?*

Trying to answer these questions will culminate in a system combining elements of statistics, hip hop theory and modern state-of-the-art machine learning techniques with endeavors into the field of computational creativity. The results will ultimately take form in textual output and evaluation of this output, to hopefully provide an answer to the stated **RQ**s.

1.3 Research Method

To address the stated **RQ**s, the first step will be to examine the field of NLP and apply analytic methods to a dataset consisting of a comprehensive catalog of rap lyrics. After sufficient analysis of this dataset, a set of linguistic and thematic patterns will have been defined, that can be used to define what characteristics are prevalent in popular and critically acclaimed rap lyrics, as opposed to patterns prevalent in unpopular and critically despised songs. For the scope of this thesis, the linguistic patterns analyzed will be limited to rhyme structure, *i.e.* what characteristics in rhymes can be found in lyrics of different ends of the critical and popularity spectrum. This analysis will result in a large dataset with lyrics and metrics determining qualities of the rhyme structure. This analytic, quantitative experiment will further be referred to as *Experiment 1*, and will be explained in greater detail in Chapter 5.

Subsequently, after implementation of an AI based generative system for rap lyrics, the generated lyrics will need to be evaluated. To answer all sub-questions of **RQ 2**, the lyrics will be run through a framework for evaluating generated text, as well as by quantitative human evaluation, to gauge the perception of the generated lyrics in relation to existing lyrics. This generation and evaluation will be referred to as *Experiment 2*, and is presented in Chapter 6.

It is important to note that when analyzing the lyrics of popular and critically acclaimed songs, the audio to which the lyrics belong will not be analyzed in any detail, although this will certainly be of some significance to critical and commercial success. This distinction between success of a song and the quality of the lyrics is acknowledged throughout the work of this thesis and will be addressed when appropriate.

1.4 Contributions

The main contributions of this thesis will be two-fold, as with the objectives and **RQ**s. Through the analysis of rap lyrics, a framework needs to be developed to identify characteristics in rhyme structure for rap songs. For the time being, there exists no widely used, universal frameworks for evaluating rhyme structure in lyrics that rewards multiple different types of rhymes. This may yield valuable information about what makes some rap songs succeed, while others do not.

To the field of NLP, the contribution will be to the specific task of generating hip hop lyrics. Furthermore, by first defining a set of characteristics in rhymes based on analysis of existing lyrics, these characteristics can be structured and applied during the generative phase to improve the quality of the output. The generated lyrics will be evaluated by whether or not they adhere to the findings of the analysis, as well as human perception. Furthermore, by combining analysis of generated lyrics, and quantitative human evaluation, there will be a clear road ahead towards generating ever better rap lyrics, and a step in the right direction in the elusive and complex field of computational creativity.

1.5 Thesis Structure

To establish the necessary background knowledge regarding lyrics within the given genre, the second chapter is dedicated to background theory in the field of *rap lyrics theory*, *lyrics analysis*, *Natural Language Generation* (NLG) and *lyrics generation*. Following this, related research and other work within the field of NLP and lyrical generative systems will be presented. After sufficient theoretical and practical backdrop has been outlined, the architecture for the implemented systems is presented. The two following chapters are dedicated to presentation of coinciding research method and results for *Experiment 1* and *Experiment 2*, respectively. This is promptly followed by a discussion and evaluation of the findings for both experiments, before the final chapter of the thesis, which will be dedicated to a conclusion of the work on the thesis as well as an outline of proposed future work within the field.

2 Background

The object of *Artificial Intelligence* (AI) driven rap lyrics generation spans a vast library of different topics, from rap lyrics theory to the inner workings of machine learning and *Neural Networks* (NN). Over the following sections there will be a presentation of fundamental knowledge about a set of topics to bolster the reader's comprehension of the this thesis.

Firstly, to be better able to understand what is going to be generated, a section aimed at providing introductory knowledge about intricacies of rap lyrics will be presented. Following this, a section will be dedicated to the field of lyrics analysis, particularly as it pertains to term frequency and subject matter. Subsequently there will be a presentation of *Natural Language Processing* (NLP), *Natural Text Generation* (NTG) and the AI methods that are frequently used for generation of text, most notably NNs like *Recurrent Neural Networks* (RNN) and *Long Short-Term Memory* (LSTM) Networks. Lastly, a section will be dedicated to the subject of computational creativty, as this is a key factor to be able to generate convincing creative textual output such as the goal of this thesis states.

2.1 Hip Hop Theory

While the musical genre of hip hop is currently the most consumed genre in the US [Nielsen Music, 2017], the techniques and intricacies of rap lyrics itself is not common knowledge. To successfully be able to generate good lyrics within a given genre, knowledge about the genre itself is a prerequisite. The following section will establish a general comprehension of the genre of Rap. It may be noted for the reader that the terms *rap* music/lyrics and *hip hop* music/lyrics may be used interchangeably over the course of this thesis, and refer to the same thing.

Rap mainly consist of three components, namely *content*, *flow* and *delivery* [Edwards, 2009]. Delivery is how the written lyrics is ultimately performed audibly, and since the goal of this paper is purely text based output, delivery will not be a chief concern. The following subsections will be dedicated to preliminary findings in the field of content and flow.

2.1.1 Content

The content of a rap, what the rap lyrics is about, can be whatever the artist desires. Nevertheless, there are certain themes that are more prevalent in the genre, especially in critically acclaimed and popular rap songs. Many of these themes have shifted in

accordance with the times the raps were written in, but some have also transcended historical context.

The earliest precursor to rap, as with most musical genres, is found in Africa. Griots[1] would tell stories rhythmically, often accompanied by drums and other primitive instruments. These stories were told to preserve the genealogies, the histories and the oral tradition of the griots' people [McKenna, 2020]. They would also serve as advisers and provide social commentary. The themes of the daily hardships and social commentary still stand strong in rap today.

The youngest predecessor of rap is blues. Some music historians have even claimed that rap is the "living form of blues" [Wald, 2004]. In terms of themes, the two genres share a lot in common. Blues can be seen as a direct descendant to the work songs and spirituals of the West African slaves in the US. These themes of oppression and hard times are deeply rooted in the blues, and subsequently in rap.

The blues also has a more provocative side, the dirty blues. The themes in this sub-genre were more humorous in nature and often included taboo topics such as sex and drug use, themes that have been, and still are, prevalent in rap.

The last topic that it is important to look at in terms of the content of rap today is "The Dozens". The dozens is an African American traditional verbal and rhythmical combat, based on rhyming schemes and insults between the duellists [Wald, 2014]. The dozens have been important in all parts of rap, in terms of delivery it gave rise to the concept of "attitude"[2], it was instrumental in the evolution of flow, and the battling nature and insults are a big part of contemporary rap as well.

2.1.2 Flow

In his book *How To Rap*, a book made up of interviews of 104 notable rappers, Edwards [2009] states: "If an artist takes his or her time to craft phrases that rhyme in intricate ways but still gets across the message of the song, that is usually seen as the mark of a highly skilled MC [rapper].". In other words, it is not solely the message *or* the structure of a song that determines the quality, true craftsmanship is to be found in the intersection between the two. This brings us on to flow, which can be broken down to three main components; rhyme, rhyme schemes and rhythm.

Rhymes are often seen as the most important part of a rap. In his book, Edwards concludes that rhymes are what give rap its musicality. Popular rhyming techniques used in rap are end rhymes (perfect rhymes), internal rhymes, multi-syllabic rhymes and sections with consistently rhyming words. This rich diversity of techniques led music scholar Adam Bradley [2009] to claim "It [rap] has done more than any other art form in recent history to expand rhyme's formal range and expressive possibilities.".

[1] A griot is a West African historian, storyteller, praise singer, poet, or musician, often seen as a leader.
[2] Attitude is a concept pertaining to a rapper or a rappers performance and is simplest translated to mean street cool.

Word	Perfect Rhyme	Vowel Assonance	Consonant Assonance
Gang	Sl**ang**	B**a**d	**G**one
Skylight	H**ighlight**	Hi**g**h li**f**e	**S**kater

Table 2.1: Displaying different styles of rhymes with the words *gang* and *skylight*.

Recent explorations into the field of rhyme and rap have emphasized the importance of assonance rhymes in the genre [Edwards, 2013]. Assonance rhymes, in contrast to perfect rhymes, do not necessarily share identical phonetic endings. Instead they appear when two words share some similar sounds, that is, they share vowel or consonant *phonemes*. Phonemes are the second smallest unit of which audible language is constructed. Breaking a phrase in a rap song down to a sequence of phonemes, and analyzing these phonemes might reveal valuable information about successful rhyming schemes. Example of different types of rhymes can be seen in Table 2.1.

Adam Krims [2000] divided flow into three categories as they relate to rhythm in his book *Rap Music and the Poetics of Identity*; ”sung”, ”percussion-effusive” and ”speech-effusive”. The ”sung” category is categorized by rhythms closely resembling those of sung pop, with rhythmic repetition, on-beat accents, regular on-beat pauses and strict couplet groupings. The other two categories are both effusive and violates the meter in some way. In the ”percussion-effusive” category the voice is used as an additional percussion instrument, with sharp staccato attacks, and in the ”speech-effusive” category the rhythms closely resemble the natural rhythms of speech.

Building on the work of Krims, Kyle Adams went on to explore what parameters rappers manipulate to create their flow. He argued that flow should be thought of as the rappers version of an instrumentalist’s technique, and went on to define the seven techniques of flow presented in Table 2.2 [Adams, 2009].

Metrical Techniques	Articulative Techniques
1. The placement of rhyming syllables. 2. The placement of accented syllables. 3. The degree of correspondence between syntactic units and measures. 4. The number of syllables per beat.	1. The amount of legato or staccato used. 2. The degree of articulation of consonants. 3. The extent to which the onset of any syllables is earlier or later than the beat.

Table 2.2: Adams’ metrical and articulative techniques of flow.

2.2 Text Mining and Lyrics Analysis

To be able to generate good and convincing lyrics one must first understand how lyrics within the given genre is constructed and structured. Text mining is a useful tool to gather as much information as possible about a corpus of text. Simply put, text mining is the

art of extracting information and uncovering insights into unstructured, semi-structured or fully structured textual data.

A common technique used to gather information about a body of text, *(i.e. the song lyrics)*, is keyword extraction, where the objective is to determine the most frequently used words and the most important words in a body of text. Popular means to achieve both these goals is to look at the *Term Frequency* (TF) and *Inverse Document Frequency* (IDF) of words within the given corpus.

2.2.1 Term Frequency and Inverse Document Frequency

TF is expressed through the equation $tf = f/d$, where f is the number of occurrences of a given word within a document with d total words. This is used to extract the most frequently used words in a text, yielding valuable information about what topics that are appearing more often in a corpus of text. In any semi-strutured or fully structured text, some words will naturally appear frequently, such as *"the"*, *"and"*, *"as"*, *etc.* which do not necessarily yield any information about the text. These types of words are called *stop words*, and are usually ignored when counting the frequency of terms.

On the other hand, the IDF of terms t within a corpus of documents D, displays a measure of how much information this term provides within a given document. This provides information about how important a given word is within a given document, the formula for which can be seen in Equation 2.1, where N is the total number of documents in corpus $|D|$..

$$idf(t, D) = log\frac{N}{\{d \epsilon D : t \epsilon d\}} \tag{2.1}$$

Despite the inherent simplicity of these algorithms, both can be very powerful tools when it comes to gathering information about a text [Qaiser and Ali, 2018]. They do however have their limitations and shortcomings, as they use bag-of-words techniques, which discards word order and ignores context. Thus, to be able to paint a more complete picture of the content, it can be helpful to also analyze the general sentiment of the text.

2.2.2 Phonemes and Rhymes

As mentioned in Section 2.1 regarding hip hop theory, rhyming and rhyme scheme is an integral part in the world of hip hop lyrics. Particularly non-perfect rhymes, such as phoneme rhymes are very frequently used. Over the recent years, attempts have been conducted breaking down lyrics into phonemes to better investigate rhyme structure and assonance rhymes [Savery et al., 2020], that is, words that share similar sounds. This is in direct accordance with analysis of phonemes and assonance rhymes as an essential tool for flow in rap music [Edwards, 2009, 2013].

There are two types of assonance rhymes, with one concerning vowel sounds and the other concerning consonants. The latter of the two also being referred to as *consonance rhymes* and some times *slant rhymes*. Unlike conventional end rhymes or perfect rhymes, where the end of the word sounds identical, assonance rhymes occur when two words

Line	IPA Conversion	Vowel Phoneme Sequence
Don't tempt me	dəʊnt tɛmpt mi	əʊ-ɛ-i
So empty	səʊ ɛmpti	əʊ-ɛ-i

Table 2.3: Example of vowel assonance rhyme between the lines "Don't tempt me" and "So empty".

share one or more identical phonemes. This is more clearly illustrated when converting lines of lyrics into *International Phonetic Alphabet* (IPA), as can be seen in Table 2.3, where two different lines, although spelt completely different contain the exact same sequence of vowel phonemes.

An example of consonance, or so called slant rhymes, can be seen in Table 2.4, where the "dʒ" sound occurs four times within the same phrase.

Line	IPA Conversion
Johnny my gentle man	dʒɒni maɪ dʒ*E*ntl maen
Join the magic band	dʒʉɪn ðə maedʒɪk baend

Table 2.4: Exampel of consonance rhyme or slant rhymes between the lines "Johnny my gentle man" and "Join the magic band".

As far as analyzing the structure of rap lyrics, it may provide valuable insight in knowing the length of the lines with regards to words or syllables [Malmi et al., 2016]. More interestingly, it may be valuable to see how these lengths differ between lyrics of rap songs with different degree of popularity or critical reception.

2.3 Natural Text Generation

While historically, there have been conceived many different approaches to natural text generation, not all have withstood the test of time. Over this section a few of the most common method for natural text generation will be described.

2.3.1 Retrieval Based Text Generation

Information retrieval is a powerful tool that has become a standard function of people's everyday life [Baeza-Yates et al., 1999]. It is the task of obtaining relevant information system resources, and this is what happens in the background every time someone perform a Google search. Text generation can essentially be viewed as a problem of information retrieval, as it boils down to retrieving the required set of words and punctuation in the correct order. field

2.3.2 Template Based Text Generation

Template based models for text generation has been around since the emergence of natural text generation as a field of interest in the 1960s. This is often viewed as a simplistic and limited approach, where labeled data is displaced to fill slots in existing templates. Some still argue that this is an unfair reputation, and that template based text generation still offers untapped potential, and particularly in combination with emerging technologies in text generation [Deemter et al., 2005].

2.3.3 Text Editing as Text Generation

Other approaches utilize an encode-tagging approach in which existing sequences of text is being encoded, tagged and edited to realize new sequences of text. This can essentially be viewed as a way of utilizing text editing as a tool for text generation.

None of the approaches described above, however, have displayed as much prowess in flexibility as neural sequence-to-sequence driven natural language generation. Which brings us to the subject of neural networks.

2.4 Artificial Intelligence and Neural Networks

As the emergence of digital computers erupted in the 1950s, the field of AI arose along with it. AI is a branch of computer science that can generally be viewed as intelligent machines which can behave like a human, think like humans, and be able to make decisions on their own [Dhankar and Walia, 2020]. This process of machines emulating human intelligence can be utilized to emulate the process of generating textual output. This branch of AI is called *Natural Text Generation* (NTG).

2.4.1 Natural Language Generation and Natural Text Generation

Natural Language Generation (NLG) systems have been around since the mid 1960s and have steadily evolved over the following decades. The process of generating natural language has long been considered one of the most challenging computational tasks [Lu et al., 2018]. The reason for this is the ambiguity of natural language, which, as opposed to artificial language has evolved naturally over time and is inherently ridden with subtext and ambiguity. *Neural Networks* (NN) can be used to generate textual output and in practice emulate natural language through machine learning. This process is called NTG.

A traditional approach to text generation with machine learning is probabilistic or likelihood based language models, like *Maximum Likelihood Estimation* (MLE). MLE models calculate the likelihood of a given word appearing, using N-grams to determine the number of words that are to be taken into consideration during the estimation. These maximum likelihood optimizations can be used to train NN language models. Generally speaking, they work by aggregating through a corpus of text and calculating the likelihood of a word w appearing given the previous words in a sequence. To calculate

the probability of x_n appearing at the end of sequence *(x_1, x_2 ..., x_{n-1})*, Bayes rule can be used to state that:

$$P(x_1, x_2, ..., x_n) = P(x_n|x_1, x_2, ..., x_{n-1})P(x_1, x_2, ..., x_{n-1})) \tag{2.2}$$

and in extension of this:

$$P(x_1, x_2, ..., x_n) = \prod_{i=1}^{n} P(x_1, x_2, ..., x_i) \tag{2.3}$$

Although MLE has shown to be effective at training systems for the purpose of generating general sentences based on a large corpus of training data [Lu et al., 2018], they do have some limitations in the fact that they have been proven to be prone to accumulating errors over time. This is because larger bodies of text generated on likelihood models are limited by their training, and are not particularly well suited for handling long-term dependencies, thus yielding unsatisfactory output over time [Bengio et al., 2015]. This problem of limited exposure is not as large of a concern when generating short phrases, as oppose to long continuous bodies of text.

The process of generating output solely based on maximum likelihood models using N-grams is also limiting in the fact that the system does not take into consideration that it is modelling language, it might as well be a string of arbitrary symbols [Rosenfeld, 2000]. A better approach to emulating natural language is through the use of NNs [Lu et al., 2018], particularly on the aspect of *Recurrent Neural Networks* (RNN) to capture long-term dependencies. To clarify, NN models are also probabilistic models, however they do not always operate on maximum likelihood.

2.4.2 Recurrent Neural Networks and Long Short-Term Memory Networks

The use of NNs for language modelling has been studied extensively since the advent of RNNs in the 1980s. An RNN is a NN that utilizes output from previous steps in the system as input in the current step. A general illustration of this concept can be seen in Figure 2.1. This process helps inform the current step about all previous actions and calculation made by the system, which in turn makes it possible to generate long streams of output with long-term dependencies. Modern text generation techniques through the use of NNs, attempt to solve the problem of ambiguous input by ascribing context to the areas of ambiguity and ironing out grammatical difficulties [Bullinaria, 1995]. Language models utilizing RNNs have succeeded in the task of generating satisfactory text output, by taking advantage of the ability to use output from earlier parts of the system as input later to interpret context in language [Lu et al., 2018].

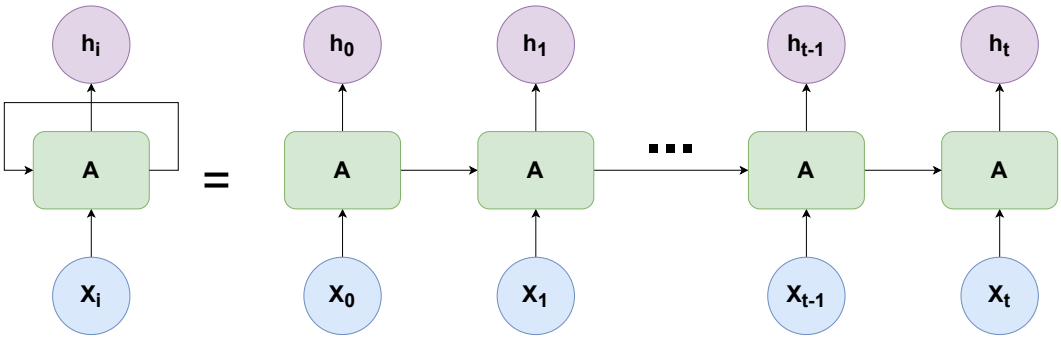


Figure 2.1: Recurrent Neural Network Architecture. Adapted from Colah's blog with permission from the author [Olah, 2015].

By utilizing this architecture, the current state of the model h_t can be calculated by using output from the previous state h_{t-1} along with the current input state x_t. This can be seen in Equation 2.4, where *f()* is some activation function.

$$h_t = f(h_{t-1}, x_t) \tag{2.4}$$

In Figure 2.1 each module represents a neuron in the neural network. Each of the neurons consist of the same fairly simple structure, *i.e.* a simple activation function like a *tanh* layer, as can be seen in Figure 2.2. In that given instance, the formula for calculating the activation function would be as stated in Equation 2.5, where W_{hh} are the weights at current neuron and W_{xh} are the weights of the input neuron.

$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t) \tag{2.5}$$

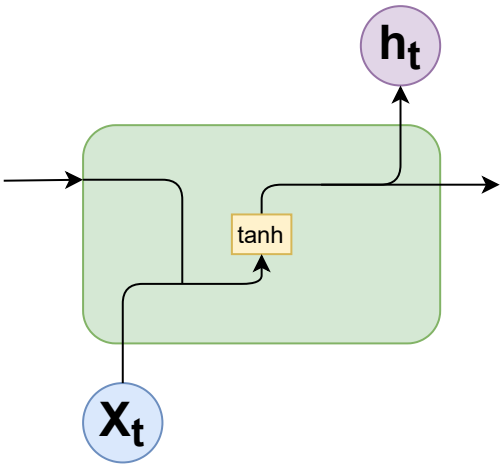


Figure 2.2: RNN Module with a ***tanh*** activation function.

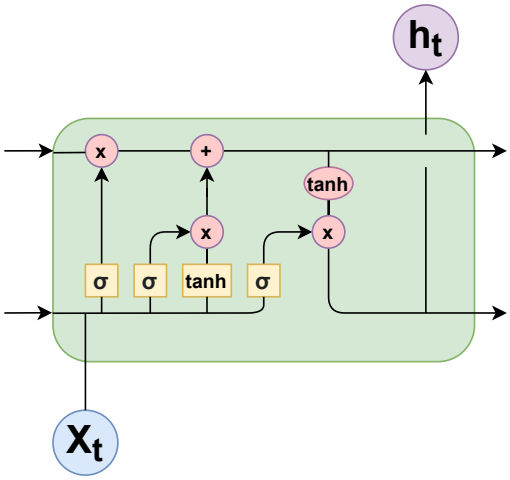


Figure 2.3: Long Short-Term Memory Module Architecture. Figure based on original design from [Hochreiter and Schmidhuber, 1997] with permission from the authors.

Long Short-Term Memory (LSTM) networks are a form of RNN better suited for the task of generating output with long sequences of dependencies [Hochreiter and Schmidhuber, 1997]. In LSTMs, each single module is equipped with a more complex four neuron layer architecture with each layer handling different aspects of the module. These layers are handling three equally important tasks:

1. decide how much previous information should be taken into consideration and calculation
2. decide how much the current input units should add to the current state
3. decide what part of the current state data should be passed to the output

These different computational tasks are being handled different neurons. As can be seen in Figure 2.3, the architecture for each module is more complex than in ordinary RNNs. Each neuron is visualized through four individual functions, namely three *sigmoid* functions and one *tanh* function that interact to execute the different tasks listed above.

2.5 Computational Creativity

Another aspect to be taken into consideration when generating an inherently creative piece of text is the concept of computational creativity. Generally speaking, computational creativity is the art of using computational means to emulate or enhance any aspect of human creativity, from problem solving to visual or audible art. The discipline of computational creativity can be found in the intersection between AI, cognitive sciences and the creative arts [ACC, 2020].

As defined by Margaret Boden [1998, 2004], for a generated work to be considered creative, it needs to be *novel*, *valuable* and *surprising*. This also applies to computationally generated pieces of work. In other words, simply emulating previously existing works will not amount to any creative output.

Some endeavours into the field of computational creativity postulate that the use of AI in creativity offers new ways to improve creativity in people [Oktradiksa et al., 2021] and even new ways to learn about creativity itself [Gobet and Sala, 2019]. Artificially intelligent creative systems are offered little to no knowledge about the world outside of their training scope, and therefore lack outside domain knowledge and discriminatory abilities that might be limiting to humans in their own creative endeavours. An important aspect of CC is the notion that the knowledge about art being partially or fully generated by a computer affects the consumers perception of the piece of art itself [Colton, 2008, Colton and Wiggins, 2012]. This is crucial to keep in mind when examining and evaluating computationally creative endeavors.

Boden also states that creativity can be either *combinational*, *explorational* or *transformational*. Combinational creativity utilizes existing elements and combines them in novel and surprising ways to make something that is valuable, while exploratory creativity utilizes and tweaks the creative space in which it resides to make creative works. Transformational creativity is the more extreme of the three types, in which the creative space itself is transformed, creating an entirely new space for future generations to combine, explore and further transform. In any system where creative output is generated based on limited domain knowledge and little to no knowledge outside this domain, the art of transforming the space substantially will be highly difficult [Colton, 2012], and the system will be limited to combinational and exploratory creativity.

3 Related Work

Over the course of this chapter there will be a presentation of related research into fields of interest, as well as a thorough presentation of some systems that share a lot of similarities with the problem description outlined in the introduction in Chapter 1.

Firstly, a section will be dedicated to the field of lyrics analysis, particularly as it pertains to rhyme and structure of hip hop lyrics, before we move on to the aspect of generation of text and lyrics and detailed outline of a selection of systems designed for the task.

3.1 Lyrics Analysis

To define what common features, patterns and themes can be found in lyrics from popular and critically acclaimed rap songs, it is beneficial to thoroughly examine both the content and the structure of existing rap song lyrics. As mentioned in Chapter 2, *Term Frequency* (TF) and *Inverse Document Frequency* (IDF) may be a good starting point to gather information of a body of text. In the case of rap lyrics analysis, the body of text will be a dataset consisting of a vast catalog of lyrics from existing rap songs. This type of strategy has been implemented in mood classification, for instance by Zaanen and Kanters [2010], where they state categorically that word oriented approaches are a valuable source for classifying the mood of the music. In other words, there is a lot of information to gather about the music in question, even when analyzing lyrics alone.

While, in the above-mentioned instance, this approach has been used to classify mood, in extension, it may be applied to analyze all sorts of thematic classifications. Classifying lyrics from successful rap songs into different thematic classes may be a valuable asset, as knowing the themes that are prevalent in popular or critically acclaimed rap music makes a ground frame for what to include and what not to include in the perfect lyrics [Mahedero et al., 2005]. As mentioned briefly in Chapter 2, however, TF and IDF display some shortcomings when it comes to sentiment analysis, as it commonly uses bag-of-words strategies and therefore lack context.

3.1.1 Rhyme Scheme and Rhyme Structure

While there has been conducted quite some research into the field of rhyming scheme and rhyme structure, there currently exist no universal framework for overall rhyme complexity.

Within the field for rhyme structure in hip hop, the importance of phoneme rhymes have been emphasized in multiple publications *i.e.* Edwards [2009, 2013], Savery et al.

[2020] and Adams [2009]. In a higher degree than traditional lyrics and poetry, hip hop relies on imperfect rhymes, in which words do not contain the same exact phonemes, but sound similar enough to constitute the perception of a rhyme [Holtman, 1996]. One approach to detect imperfect rhymes in lyrics is using methods commonly used in detection of combinations of amino acids, in a similar way to detect combinations of phonemes in lyrics and scoring each pair of phonemes to calculate the probability that this constitutes an imperfect rhyme [Hirjee and Brown, 2009].

3.1.2 Existing Frameworks for Lyrical Rhyme Analysis

While most researchers agree on the general theory of rap lyrics, they may have varying approaches to rhyme detection, which highlights completely different aspects of rhyme scheme and structure. Some examine rhymes, monosyllabic as well as polysyllabic, on a word for word basis while others break lines or even complete phrases into continuous strings of phonemes. One example of the former was conducted in accordance with an *Information Retrieval* (IR) approach to hip hop lyrics generation [Malmi et al., 2016]. In this case the authors defined *Rhyme Density* as a self-defined metric for quantifying the technical quality of the rhyme structure. Rhyme density in this regard is simply put an average of the longest matching number of phonemes per word in a song lyrics and is defined by a single number (float). This system will be further explored in Section 3.3. Approaches like these that analyze rhymes on a word-for-word basis offer their limitations with regards to identifying polysyllabic rhymes spanning multiple words, which are fairly common in in hip hop lyrics.

3.2 Lyrical Text Generation and the Aspect of Computational Creativity

When approaching the aspect of lyrical and creative text generation, the first subject to explore is natural text generation and in turn, tie this to computational creativity. Subsequently this could be tied specifically to the genre of hip hop.

3.2.1 Natural Text Generation

While approaches to text generation have been attempted using IR, like the aforementioned hip hop generaion system DopeLearning, most modern approaches utilize some variation of *Neural Networks* (NN). As the theory behind NN driven *Natural Language Generation* (NLG) systems are described in some detail in Chapter 2, this section will be limited to describing practical applications of NNs in modern text generation systems.

The task of NLG have deep roots in the field of AI, as the main challenges with generating natural language is the implicit nature of communication. According to *Handbook of Natural Language Processing* [Dale et al., 2000], as the field of generating natural language emerged as a legitimate subfield of *Natural Language Processing* (NLP) in the 80s, the field seemed to be of greater concern among scientists than engineers. This

is because if somebody were to successfully create a convincing natural text generation system with all the nuances and subtleties of natural language, the practical applications would be less interesting than the implications it would have on the field of human linguistics. Because of this inherent relationship between text generation and human intelligence, traditionally systems approaching the task of NLG start by emulating some aspects of human intelligence, which is the general idea behind the field of AI.

Historically, hip hop has not been the main focus of musical and lyrical generative systems. There has, nevertheless, been conducted extensive research into the field of NLP and the field of *Natural Text Generation* (NTG), as well as some endeavors into the realm of hip hop lyrics generation. The conventional approach, and most benchmark approaches utilize a sequence-to-sequence neural network approach, for instance *Long Short-Term Memory* (LSTM) networks.

3.2.2 NTG and Hip Hop

There have been several different approaches to lyrical generative systems though the past decades. Some of these systems are directly tasked with generating hip hop lyrics, like Shimon The Rapper [Savery et al., 2020], Ghostwriter [Potash et al., 2015] and DopeLearning [Malmi et al., 2016]. While the two former use a *Recurrent Neural Network* (RNN) approach, namely LSTM networks, the latter uses IR to combine existing lines of rap lyrics to generate longer, novel phrases. The validity of IR as a text generation tool will be further examined in Section 3.2.3 where the field of *Computational Creativity* will be discussed. Other approaches to lyrical text generation include template-based models as well as context-free grammar approaches that use extensive N-gram grammatical analysis to generate phrases consisting of shorter sentences [Pudaruth et al., 2014]. Some of the approaches mentioned above will be described in greater detail in Section 3.3.

3.2.3 Computational Creativity

While the art of generating natural language is no easy task in and of itself, it becomes significantly more difficult when attempting to simultaneously tackle the task of emulating human creativity. Colton [2008, 2012] posed that creative systems in addition to providing us the creative works they generate, has the potential to expand the limits of artificial intelligence while at the same time furthering human creativity in as far as helping us understand what creativity actually is.

3.2.4 Creative Text Generation

Language models (LM) have historically fared well on task-based text generation with both syntactic and semantic representations, however, the main challenge with NTG is that natural language is latent with subtext that is more challenging to emulate successfully [Radford et al., 2019]. There have been made significant advancements in

zero-shot[1] and few-shot[2] LMs like the state-of-the-art *Generative Pre-Training* systems GPT-2 and GPT-3. These language models have shown significant improvements in resolving ambiguity in text input [Brown et al., 2020], and have come closer than any other LMs when it comes to emulating natural language.

These challenges in subtext and ambiguity are only amplified when trying to generate language within the creative realm of lyrics and poetry, which is traditionally riddled with symbolism, metaphors and subtext. As mentioned above, the task of generating lyrical text in and of itself has been approached several different ways from RNN approaches to IR based systems. Although IR has shown to provide exciting opportunities for creative text generation [Veale, 2011] and computational creativity in general [Boden, 2004], however, it may not be the preferred approach for a system pursuing the task of generating one single phrase of lyrics.

A major concern in lyrics generation is the inherent subtext and metaphors that are a part of the genre [Edwards, 2009]. There have been made attempts in the field of self discriminatory systems, such as adversarial networks that produce lyrics [Saeed et al., 2019] and other methods for evaluating generated creative text [Potash et al., 2018]. One such discriminatory generative method is the use of *Generative Adversarial Networks* (GAN) [Goodfellow et al., 2014], which has shown improvements in creative text generation by both human evaluation standards, as well as established language evaluation frameworks like BLEU[3] [Saeed et al., 2019, Yu et al., 2017]. GANs are most commonly used for image generation, and have shown great results in both image and text generation [Denton et al., 2015].

For the purposes of generating lyrical phrases that attempt to achieve a pre-defined level of quality, the system would likely benefit from a method that utilizes context-based learning to help achieve that goal. LSTM approaches have been proven to provide an advantage over context-free N-gram models [Potash et al., 2015], by generating smaller sequences of text that can in turn be put together into well crafted phrases, and overall great prowess in text generation [Graves, 2014]

Generating a system that analyzes these rhyme scheme patterns may be essential to be able to generate good lyrics. This may be challenging, as there is currently no widely used conversion system sufficient enough to capture all the different dialects and pronunciations in hip hop [Savery et al., 2020].

3.2.5 Word-for-Word vs Character-for-Character Generation

When it comes to determining whether to use a word-for-word approach to text generation as opposed to a character-for-character, there are several aspects to take into consideration. Using individual characters for constructing comprehensive text sequence offer the clear advantage of having a small set of variable in the vocabulary, which may significantly

[1]Zero-shot is a problem setup in which the system is classifying input with no labeled training data. This allows the system to solve any number of LM tasks without any task-specific learning.

[2]Few-shot is a problem setup in which the system is classifying input with only a few examples of labeled training data. This limits the need for extensive sets of tagged data during training.

[3]BLEU is an abbreviation of *Bilingual Evaluation Understudy*.

improve training time. Seeing as there are only 26 characters in the Latin alphabet in addition to any punctuation used, the set vocabulary for training would be far smaller when compared to a word-for-word based generation model, which may include hundreds of thousands of unique variables in the training vocabulary. Running through the same training data with the same sequence length does however require far more sequences to be processed, which may negatively impact the time necessary to train the system. The clear advantage of using word-for-word based generation is that the likelihood of producing misspelled words is relegated to the misspelled words already present in the training data. As seeing a misspelled word in a generated sequence of text instantly lowers the credibility of the produced text, this is well worth introducing added complexity, in the form of a larger word-for-word vocabulary, to avoid. Word-for-word models are also better suited for long term linguistic dependencies, as there is more information in a sequence of a set length in relation to a character-for-character sequence of the same length.

3.3 Systems for Generation of Hip Hop Lyrics

There are a couple of systems and research projects that share many similarities with the subject matter of this thesis. For the final section of this chapter a fairly detailed description of these similar systems for future reference during the presentation of this system.

3.3.1 DopeLearning: Information Retrieval and Rhyme Analysis

A section detailing the system presented in [Malmi et al., 2016], particularly as it pertains to IR as an effective tool for coherent text generation and rhyme density as a measure of quality in hip hop lyrics.

In their implementation, Malmi *et al.* utilized IR to generate new phrases of hip hop lyrics employing existing lines of lyrics. As described in Section 2.3.1, IR is an effective tool to generate syntactically and semantically comprehensive bodies of text with the quality in these regards dependent on the domain specific knowledge [Smeaton, 1992]. In the instance of this system, the repository consist of individual lines from existing rap songs, thus the syntactic and semantic quality of each line is constrained by the quality of the original artist's writing, though it would be hard to argue that it is not convincing within the domain of hip hop lyrics. In this way information retrieval can be seen a a shorthand to generate individually comprehensive lines text, however when it comes to generating composite phrases, the challenge of semantic coherence between the lines becomes evident.

To breach the semantic gap between one line and the following line, each line is being converted into a high dimensional vectors that capture semantic and grammatical features. Such a model is described in Pennington et al. [2014]. After the initial line of text has been retrieved by the system, the best next line can be predicted utilizing these vector-space representations of each line. In addition to crafting coherent phrases,

their paper proposes a single metric for the quality of rhyme in a rap lyrics. Their *rhyme density* metric is defined as the average length of matching sequences of phonemes between each word and the following words. This metric utilizes phoneme matching and phoneme rhymes, as has previously been pointed out as a key attribute in hip hop lyrics, however, it does not account for other types of rhymes as well as phoneme rhymes that span multiple words in combination.

3.3.2 Shimon the Rapper: LSTM and Real-Time Interaction

A fairly different approach to the generation of hip hop lyrics can be seen in Savery et al. [2020]. Their system is a real-time freestyle rapping robot and concerns topics of speech synthesis, text-to-speech, speech-to-text, robotics, latency for real-time-interaction and of course hip hop lyrics generation. Of these topics, only the former is of relevance for this thesis, and the rest will have to be consumed at the reader's behest.

As their system aims for efficiency, given the interactive aspect, and are only generating short phrases, the mode of text generation is NN based utilizing an encoder/decoder LSTM network to generate multiple short lines of text. After these lines have been generated, each line's rhyme quality is being scored based on the internal rhymes of each line. This includes both perfect rhymes and slant rhymes. Following the selection of the initial line of text given the highest internal rhyme score, as well as connection to a given subject matter, this initial line is being paired with the best next line based on the quality of rhymes between both lines.

3.3.3 Ghostwriter: LSTM for Emulating Artist Styles

Another system utilizing LSTM to generate hip hop lyrics, however with different attention, is Potash et al. [2015]. In this instance the aim is to convincingly emulate the style of specific artists. Their system is also using LSTM to genrate lyrics, however, as opposed to Shimon the Rapper, Ghostwriter is generating entire verses. As the object of the system is to emulate existing artist's styles, the length of these verses will vary. The vocabulary used in the textual output is confined to the vocabulary in the training data, which will be the existing catalog of lyrics for the respective artist.

The LSTM is trained on the existing catalog of the respective artist, and there are no inherent checks and balances for the rhyme quality or structure of the song. After generation of multiple verses, the verses are matched with the existing catalog to find the proper balance between stylistic correlation and novelty.

3.3.4 Lasertagger: Text Editing as Text Generation

As a response to the neural sequence-to-sequence models becoming the de facto approach to text generation, a novel approach to text generation was introduced that tried to circumvent the need for large amounts of training data and long inference time. The answer was a system that uses tagging of existing text sequences to be able to generate satisfactory text output faster and with significantly less training [Malmi et al., 2019].

This system works by encoding text sequences, tagging each element (word, punctuation) in the sequence and generating new text by performing one of a set of operations (add, delete, keep) on each element in the sequence. This approach of essentially viewing text generation as an extension of text editing performs at benchmark on several tasks when compared to neural sequence-to-sequence models when the training data is large, and outperform them outright when training data is limited.

3.4 Emerging Approaches and State-of-the-Art Systems for Text Generation

Although NLP and NTG have been subjects of interest for many a decade, new and exciting approaches emerge every couple of years. Here will be presented a couple of the more exciting and promising approaches, achieving state of the art benchmarks in many different tasks.

To forgo the need for recurrence in sequential output, Vaswani et al. [2017] proposed the Transformer, a model architecture capable of drawing global dependencies relying entirely on attention mechanisms. Systems like the Google OpenAI *Generative Pre-Training* (GPT) models [Brown et al., 2020, Radford and Narasimhan, 2018, Radford et al., 2019] utilize this transformer architecture and achieve state-of-the-art benchmarks in a multitude of tasks like machine translation, on-the-fly reasoning and arithmetic. By abandoning the recurrence, the system is more prone for parallelization and generally require less training than RNN models.

Other systems, like the one proposed in Malmi et al. [2019] attempt to look at text generation as a text editing task. This is realized by reconstructing target sentences with three edit operations, *delete*, *keep* and *add*. The edit operation is calculated through the combination of an encoder and a transformer. Given a limited training data, this approach outperforms baseline sequence-to-sequence encoder-decoders on tasks like sentence fusion, text summarization and grammar correction.

4 Architecture

The *Research Questions* (**RQ**)s presented in Chapter 1 pose two related yet ultimately distinct problems. Thus, to be able to answer all sub-questions of **RQ 1** and **RQ 2**, two software systems needed to be developed. One system for the analysis of hip hop lyrics with regards to rhyme scheme and rhyme complexity, and a separate system for the generation of hip hop phrases.

This chapter describes the architecture and technical details of both the system for the lyrics analysis and for lyrics generation, as well as a presentation of the dataset that was used for each of the systems. The architecture presented in Section 4.1 relates to what will be presented in Experiment 1 in Chapter 5, while Section 4.2 present the system used in Experiment 2 elaborated upon in Chapter 6.

4.1 Lyrics Analysis

For the lyrics analysis system, the goal was to identify different types of rhymes to determine how complex the rhyming structure of the lyrics is, and how this relate to critical acclaim and popularity. The first challenge in developing such a system is to establish a framework for complexity of rhymes. The components of the rhyme complexity framework developed for this thesis is detailed in Sections 4.1.1 and 4.1.2. Subsequently, the architecture and workflow of the system analyzing the lyrics is presented in Section 4.1.3.

4.1.1 Rhyme Metrics and Rhyme Complexity

Over Sections 3.1.1 and 3.1.2, quite a few methods and metrics for identifying and quantifying rhyme scheme and structure are presented and elaborated upon. Each of these alternatives display different approaches to detection and focus on different aspects of rhymes. However, none of them capture the entire spectrum of rhyming within the genre of hip hop.

To be better able to capture a multitude of different types of rhymes, a framework was designed to detect ten distinct rhyme metrics, each representing different types of rhymes. These metrics can be seen under the *Rhyme Metric* column in Table 4.4, and all these rhyme metrics can in turn be aggregated to showcase an overall rhyme complexity score for each individual song's lyrics. This single aggregated score is called the ***rhyme complexity*** of the song, and is ultimately what is used to characterize the complexity and intricacy of a lyrics' rhyme scheme.

Lyrics	Phonemes	Vowel Sequence	Matching	Score
I feel so empty	aɪ fil səʊ ɛmpti	aɪ-i-əʊ-ɛ-i	əʊ-ɛ-i	3
So don't tempt me	səʊ dəʊnt tɛmpt mi	əʊ-əʊ-ɛ-i	əʊ-ɛ-i	3

Table 4.1: Example of two lines being run through longest assonance rhyme matching algorithm ultimately receiving a score of 3 for exhibiting 3 consecutive matching vowel phonemes.

The rhyme complexity of a song is calculated by adding up all ten distinct rhyme metric scores. The intention of gathering all these individual metrics and adding them together is to reward different styles of rhymes. The rhyme complexity accounts for five different styles of rhymes, and to be able to reward both short term rhyme complexity and overall rhyme complexity for an entire song's lyrics. Each of these five types of rhymes are therefore divided into highest score for one single line in a song and the average score for each line in the entirety of one song. The more detailed description of the different types of rhymes can be found in Section 2.1.

A specific description of the different kinds of rhymes used in the rhyme complexity framework, and how they are calculated in the system can be seen below.

Assonance Rhymes

Assonance rhymes are here defined as a sequence of matching vowel phonemes, without all adjacent consonant phonemes matching, between two lines of lyrics. An example can be seen in Table 4.1. This operation is performed on every line of a song's lyrics, each line is compared with the three subsequent lines, and ultimately each line receives a longest assonance rhyme score that represent the length of the longest matching sequence of vowel phonemes (without all adjacent consonant phonemes matching) between this line and either of the three subsequent lines. To reward complexity in a single section of the lyrics, the longest assonance rhyme for any line in the entire lyrics is being represented. This is the metric *Longest Assonance Rhyme* metric. To reward consistently high complexity in assonance rhymes for a song, the average length of the longest assonance rhymes for each line is also calculated. This is the *Average Assonance Rhyme*.

Internal Rhyme

Internal rhymes are here defined as a string of phonemes within one word that matches another word within the same line of lyrics. For each word in a line, the phoneme sequence of this word is being compared with the phoneme sequence of every other word within the same line. If there is a matching phoneme sequence within one word and another within the same line of lyric, this yields an internal rhyme. All the internal rhymes for each word in the line is counted and this results in an internal rhyme score. For an example, see Table 4.2. To reward complexity in rhymes in a single line of a song, the highest number of internal rhymes within one single line is represented. This is the

Highest Internal Rhyme metric. To account for overall complexity in internal rhymes, the average number of internal rhymes within every line of the lyrics is also calculated.

Lyrics	Phoneme Lyrics	Longest Match	Score
We live to fight the night	wi liv to faɪt ðə naɪt	[(f-**aɪ**-**t**, n-**aɪ**-**t**), (w-i, l-i-v)]	2

Table 4.2: Example of internal rhymes. This method is also used for identifying word rhymes. Word rhymes as opposed to internal rhymes do not take place on one single line.

Word Rhymes

Whereas internal rhymes compare and match words within single lines of lyrics, word rhymes are defined as a sequence of phonemes within a word that matches a sequence of phonemes within another word over the following three lines, and not internally in the same line. This operation is being executed on every line of a song's lyrics, and the resulting number represents for each line the number of words that share matching phoneme sequences with another word over the three following lines. To reward high complexity in word rhymes over a single line, the highest score of word rhyme for any single line in the lyrics is represented. This is the *Highest Word Rhyme* metric. To account for consistently high complexity in word rhymes, the average of word rhymes for each line in a song is calculated as well.

Alliteration

Alliteration in the context of this metric is the highest number of one single letter within one line of lyrics. This can be seen in Table 4.3. To reward high score in alliteration for a single section of a song, the highest alliteration for one line in the entire song is set as the highest alliteration. This is executed for both vowels and consonants. These are the *Highest Vowel Alliteration* and *Highest Consonant Alliteration* metrics. To reward consistently high vowel and consonant alliteration, the average of highest vowel and consonant alliteration for each line in a song is calculated. These are the *Average Vowel Alliteration* and *Average Consonant Alliteration* metrics respectively.

Alliteration Type	Lyrics	Most Used	Score
Vowel	See, we live to fight the night	e	5
Consonant	See, we live to fight the night	t	4

Table 4.3: Example of highest vowel and consonant alliteration for one line of lyrics.

4.1.2 Rhyme Complexity

In this thesis the complexity and intricacy of lyrical writing is ranked on one single score, the *rhyme complexity*; a combination of the 10 rhyme metrics listed above. This metric is designed to reward different types of rhymes and to account for short term complexity in an individual song's lyrics as well as consistently high complexity over the entire lyrics.

After all the individual rhyme metrics are calculated, each metric is given a scale from the lowest calculated value within the dataset to the highest calculated value, as seen in Equation 4.1. After the highest and lowest value has been calculated, each metric of each song can be placed on this relative scale and given a number between 0 and 1, as seen in Equation 4.2, which combined produce one single number between 0 and 10 that represent the complexity of rhymes relative to the entirety of the dataset. A complete example of calculation of rhyme complexity can be seen in Table 4.4.

$$\textit{rhyme metric scale} = \max|\textit{rhyme metric}| - \min|\textit{rhyme metric}| \tag{4.1}$$

$$\textit{song rhyme metric score} = \frac{\textit{song rhyme metric score}}{\textit{rhyme metric scale}} \tag{4.2}$$

4.1.3 Technical Description of Rhyme Analysis

The lyrics analysis dataset is split into lyrics from individual songs. The system iterates through each lyrics and performs a set of tasks as follows:

Firstly, the lyrics is split into a nested list containing each individual line of the lyrics which in turn contains each single word in the line. This list is then iterated through and each word is translated into *International Phonetic Alphabet* (IPA) by Pyhton's builtin nltk library[1], where the lyrics is returned as a list of phonemes separated by each word within each line to be able to detect word rhymes and internal rhymes. This division can be seen in Figure 4.1.

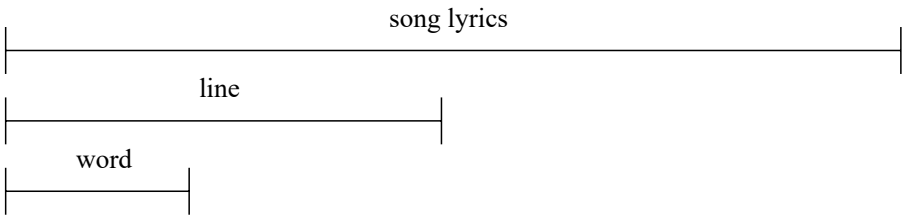
After translation is done, each line is iterated through to detect alliteration by counting occurrences of each unique phoneme, word rhymes and internal rhymes, as well as multi-syllabic phoneme rhymes regardless of word division within each line. After this iteration, each song's lyrics is attached with one single float number for each of the metrics listed above. These metrics are combined to create one single rhyme complexity score for each song. The general architecture for generating rhyme stats for a song's lyrics can be sen in Figure 4.2.

When the entirety of the lyrics analysis dataset has been run through the rhyme complexity framework, the rhyme complexity of the songs can be compared to the Metacritic score, user score on *www.metacritic.com* and popularity metric from Spotify, to identify pattern and correlations between the different metrics.

[1] nltk is Python's Natural Language Toolkit library.

Rhyme Metric	Song's Rhyme Metric Score	Lowest Rhyme Metric Score	Highest Rhyme Metric Score	Song's Relative Rhyme Score
Longest Assonance Rhyme	9	0	14	0.64
Average Assonance Rhyme	1.5	0	4.18	0.36
Highest Internal Rhyme	5	0	7	0.71
Average Internal Rhyme	0.69	0	5.09	0.14
Highest Word Rhyme	5	1	8	0.57
Average Word Rhyme	4.36	0.47	10.18	0.40
Highest Vowel Alliteration	9	2	35	0.21
Average Vowel Alliteration	2.1	1.07	4.93	0.27
Highest Consonant Alliteration	9	2	35	0.21
Average Consonant Alliteration	2.1	1.07	4.93	0.27
Total Relative Rhyme Complexity Score				**2.5**

Table 4.4: Example calculation of rhyme complexity score.



[[[phon, ..., phon], ..., [phon, ..., phon]], ..., [[phon, ..., phon], ..., [phon, ..., phon]]]

Figure 4.1: Nested list of lyrics converted to phonemes. Each song lyrics is split by lines, each line is split by words and in turn each word is split into phonemes for rhyme detection.

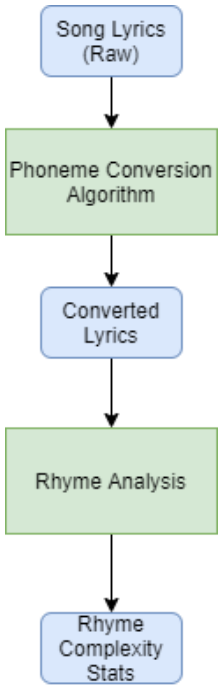


Figure 4.2: General architecture for generating rhyme stats.

4.1.4 Term Frequency and Subject Matter

To be able to generate lyrics comparable to existing lyrics, it is useful to analyse the subject matter of popular and critically acclaimed songs' lyrics. This can be done by running a vast catalog of existing lyrics through a term frequency algorithm and returning the most frequently used words within the catalog. This process returns a list of single words that can be combined to be used as a starting point for the generative system after the lyrics generation module has been trained. To be able to count the frequency of words with essentially the same meaning, the words first need to be stemmed and lemmatized. Stemming is the process of removing the beginning or ending of words, while lemmatization denotes the process of mapping several different words to one single form. Combined they greatly reduce the number of variations of essentially the same word that are being counted as multiple different words, *e.g. study* and *studying*, or *am*, *are* and *is*.

4.2 Lyrics Generation

The generation of hip hop lyrics phrases for this thesis, were so executed with the use of *Recurrent Neural Networks* (RNN), more specifically a *Long Short-Term Memory* (LSTM) network, as described in Sections 2.4.2 and 3.2.1. A large catalog of lyrics, as described in Section 4.3.1, is iterated through to create a long sequence of consecutive words, converted to lower case and stripped of punctuation *(except hyphen "-" and apostrophe "')*. Additionally a vocabulary is produced, consisting of all unique words that are found in the lyrics dataset. This vocabulary is used as the set of variables used during training and generation.

4.2.1 LSTM Model Details

The model that was used in the experiment used a three layer LSTM, for which the specific hyperparameters[2] used seen in Table 4.5. These hyperparameters yielded the most coherent results based on manual inspection. For the training of the LSTM model, we utilized NTNUs *High Performance Computing* (HPC) cluster IDUN/EPIC [Själander et al., 2019].

4.3 Dataset

The dataset used for lyrics analysis and lyrics generation consist of essentially the same data. They are, however, structured quite differently. The lyrics used in both datasets was all gathered from the same rap lyrics scraper repository on Github [Paupier, 2021], and distributed through Kaggle[3].

[2] Hyperparameters are parameters whose value is used to control the learning process for machine learning.

[3] www.kaggle.com is the world's largest data science community.

Parameter	Specification
Hidden layers	512, 256 and 128 neurons.
Sequence Length	16
Embedded Dimensions	64
Activation Function	RELU
Loss Function	Sparse Categorical Crossentropy
Epochs	50

Table 4.5: Specifications for hyperparameter used in the LSTM network for hip hop lyrics generation.

4.3.1 Dataset for Lyrics Generation

For training the AI module of the generative system, a single text file was used with lyrics from 36 different rap artists, consisting of in total ∼2800 songs and ∼1'400'000 words. This is a selection of songs from the music catalog of some of the most popular and critically acclaimed artists, as well as some less known artists and lyrics from less critically acclaimed albums. This is essential as we need both sets of the spectrum with regards to popularity and critical acclaim to be able to detect any differences between across the spectrum.

4.3.2 Dataset for Lyrics Analysis

As the lyrics analysis system was dependent on more information about the lyrics, a more limited set of 838 songs songs were used. This is a sample of the same set of lyrics that was used for the generative system. Each song lyrics was listed individually along with metadata about each lyrics. A full list of metadata associated with each song can be seen in Table 4.6. The purposes of all these fields will be further explained over the previous section regarding lyrics analysis.

ID	Unique ID for each song lyrics.
Artist	Artist delivering (at least parts) of the lyrics.
Song	Name of the song.
Album	Album the song was released on.
Lyrics	The entire lyrics of the song, even parts for featured artists.
Metacritic Score	Average critics score for the album.
User Score	Average user score for album (on Metacritic.com).
Popularity	Spotify Popularity Score for the album the song appears on.

Table 4.6: Data affiliated with each individual song for the object of lyrics analysis.

5 Experiment 1: Rhyme Complexity in Rap Lyrics

The research aimed at answering the *Research Questions* (**RQ**)s presented in Chapter 1, is divided into two distinct experiments, each representing one of the **RQ**s. This is in line with the bipartite focus between lyrics analysis and lyrics generation for the thesis.

For the lyrics analysis, an experiment was conducted with the intent of establishing a universal framework for determining complexity of rhymes in lyrics, and in turn identify whether or not there are any correlations that emerge between rhyme complexity, and Metacritic score, user score on *www.metacritic.com* or popularity. This experiment is referred to as Experiment 1, and the following sections will provide a detailed description of the experiment and method, as well as a presentation of the results from the experiment.

Research related to answering **RQ 2**, regarding the generation and evaluation of rap lyrics will be presented in a similar manner in Chapter 6. Subsequently, a more detailed discussion, evaluation of the research and results from both experiments, as well as conclusion will be presented in Chapters 7 and 8.

5.1 Research Method and Description

The purpose of Experiment 1 was to identify correlations between complexity in rhyme structure, and either of the three metrics; popularity, Metacritic score or user score. These three metrics are all part of the dataset that is used. As such, the premiere challenge for the experiment was to develop a framework for assessing complexity of rhymes.

5.1.1 Setup of Experiment

The metrics used for this experiment are all part of a framework developed expressly for this thesis, with the purpose of being able to evaluate the complexity of rhymes over multiple categories of rhymes. This framework is presented in detail in Section 4.1, and consists of 10 rhyme metrics that culminate in one single metric called the *rhyme complexity*, the composition and calculation of rhyme complexity is explained in detail in Section 4.1.1.

5.1.2 Research Method

This analysis yielded valuable input in the relationship between complexity of rhymes in hip hop lyrics, and people's perception, enjoyment and consumption of said hip hop music.

The experiment is quantitative and utilizes a dataset that has been specially tailored for the task at hand, described below. Analysis is being conducted experimentally by first calculating the rhyme complexity metric and conducting statistical analysis on the findings.

Through human evaluation of lyrics, conducted in Experiment 2 and described in Section 6.2, an attempt was made at ascertaining to some degree, whether or not the result of the rhyme complexity framework coincides with people's perception of rhyme complexity for this lyrics.

5.1.3 Description of Dataset

After running the catalog of all lyrics in the analysis dataset through the rhyme complexity algorithm described in Section 4.1.2, each song is attributed a rhyme complexity score. In addition, each song have their own respective Metacritic score, user score from *www.metacritic.com* (further referred to as simply *user score*) and popularity score from the popularity metric on Spotify. At this point the songs will be ordered after one of these three metrics; popularity, Metacritic score or user score to see if there are any patterns or general trends that emerge when compared with the rhyme complexity.

5.2 Results of Experiment 1 - Rhyme Complexity

As described in Section 5.1, the goal of Experiment 1 was to determine whether or not there are any correlations between rhyme complexity, and popularity, Metacritic score or user score. Through the lyrics analysis, the general patterns that emerged can be summarized as follows:

1. A weak positive correlation was found between Metacritic score and rhyme complexity. In general terms this means that songs that score higher on Metacritic's aggregated reviewer score tend to have a higher rhyme complexity, relative to songs with less favorable reviews.

2. A similar weak positive correlation was found between user score and rhyme complexity, indicating that songs that are rated higher by users on Metacritic.com tend to have a higher rhyme complexity than songs that are rated lower.

3. A weak inverse correlation was found between popularity and rhyme complexity. This indicates that hip hop songs that are popular on Spotify tend to score lower on rhyme complexity, when compared with less popular songs.

These three different finding will be described in more detail individually over the following subsections.

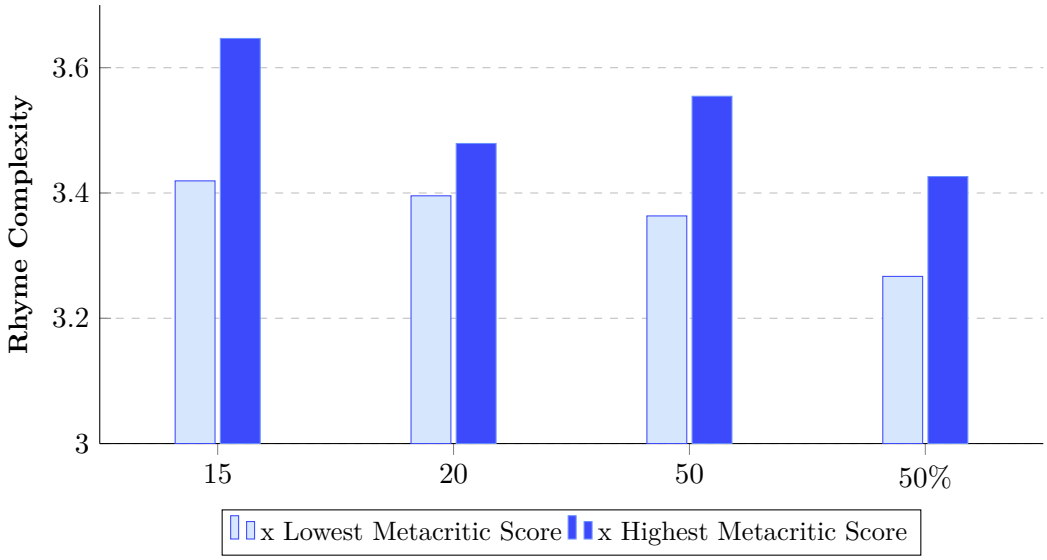


Figure 5.1: Correlation between rhyme complexity and Metacritic score for the x highest and lowest rated songs by user score.

5.2.1 Rhyme Complexity and Metacritic Score

As briefly mentioned above, there was discovered a weak positive correlation between Metacritic score and rhyme complexity. This correlation is evident when comparing the rhyme complexity score of the x highest rated songs by Metacritic score with the x lowest rated songs by Metacritic score, as can be seen in in Figure 5.1. Most notably, when comparing the lowest rated half of the songs with highest rated half (the rightmost columns) it can be observed that there is a disparity of 0.16 points in rhyme complexity, in favor of the higher reviewed albums.

This experiment was executed on a selection of 598 songs with associated Metacritic score for the album the song was released on. A display of all the songs' Metacritic score and rhyme complexity can be seen in Figure 5.2. The trend line is also displayed, showing the correlation with a $y = 0.0085x$ incline, indicating that rhyme complexity increases by 0.0085 points per point increase in Metacritic score. This warrants the distinction of a weak positive correlation.

The trend line has an R^2-value of $R^2 = 0.020$ indicating a high degree of variation within the points around the trend line. As is evident by this fairly low R^2-score and can be seen plainly by simply observing the plot is that the results of the analysis makes for a highly dispersed scatter plot.

A couple of outliers of note can be seen in the song with the highest rhyme complexity score being situated at a meager 70 point on Metacritic score, while the song with the lowest rhyme complexity score displays a Metacritic score of 89. This is in direct contrast with the indication of the correlation between the two. Although outliers like these do

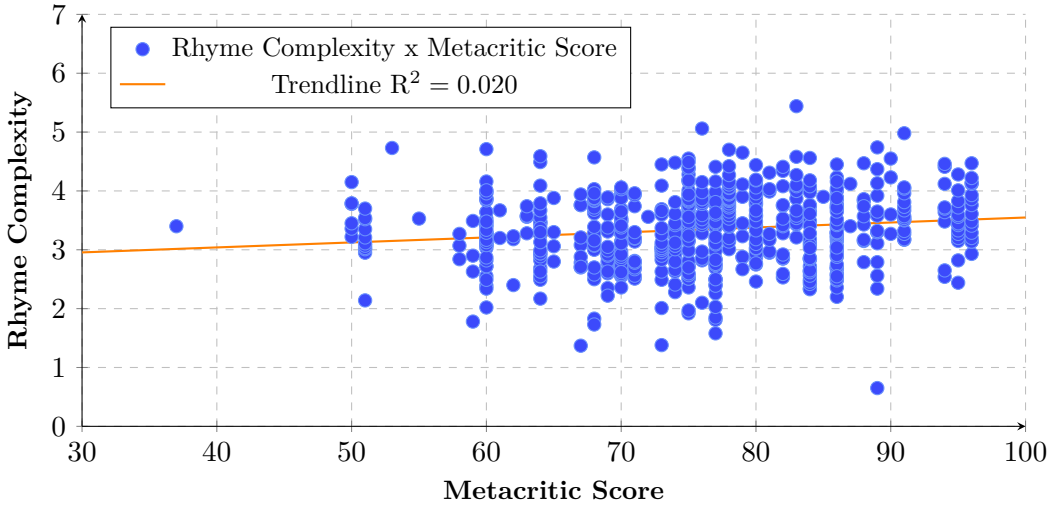


Figure 5.2: Correlation between rhyme complexity and Metacritic score, also showcasing a trend-line for the general trend in this correlation.

not dispute the general trend, they still stand as a demonstration of the large dispersion within the results. Further limitations, concerns and implications of the presented findings will be further elaborated upon in Chapter 7 regarding discussion and evaluation.

The rhyme complexity score that is used to detect the correlation between overall rhyme complexity and Metacritic score consist of a combination of 10 different metrics. An overview of the 10 different relative rhyme metrics scores can be seen in Figure 5.3. The light blue columns represent the average relative rhyme metric score for the 50% lowest rated songs by Metacritic score while the dark blue column represent the 50% highest rated. From this overview we can see that the highest rated half of songs by Metacritic score outperforms the lowest rated half in every category of rhyme, although with small margins in every instance, adding up to the total 0.16 points difference in overall rhyme complexity seen in Figure 5.1.

5.2.2 Rhyme Complexity and User Score

Although Metacritic score and user score on metacritic.com are conducted by independent actors as well as by people of different qualifications, there is a clear correlation between the two scores for the same albums. The similarities between these different scores can be seen in Figure 5.4. The consequence of this is that even though the scoring is done on different merits for these two metrics, it is not surprising to see many of the same correlations between user score and rhyme complexity as we did with Metacritic score and rhyme complexity.

From Figure 5.5 we can see that the relationship between rhyme complexity of the x highest rated songs and the x lowest rated songs with regards to user score is generally

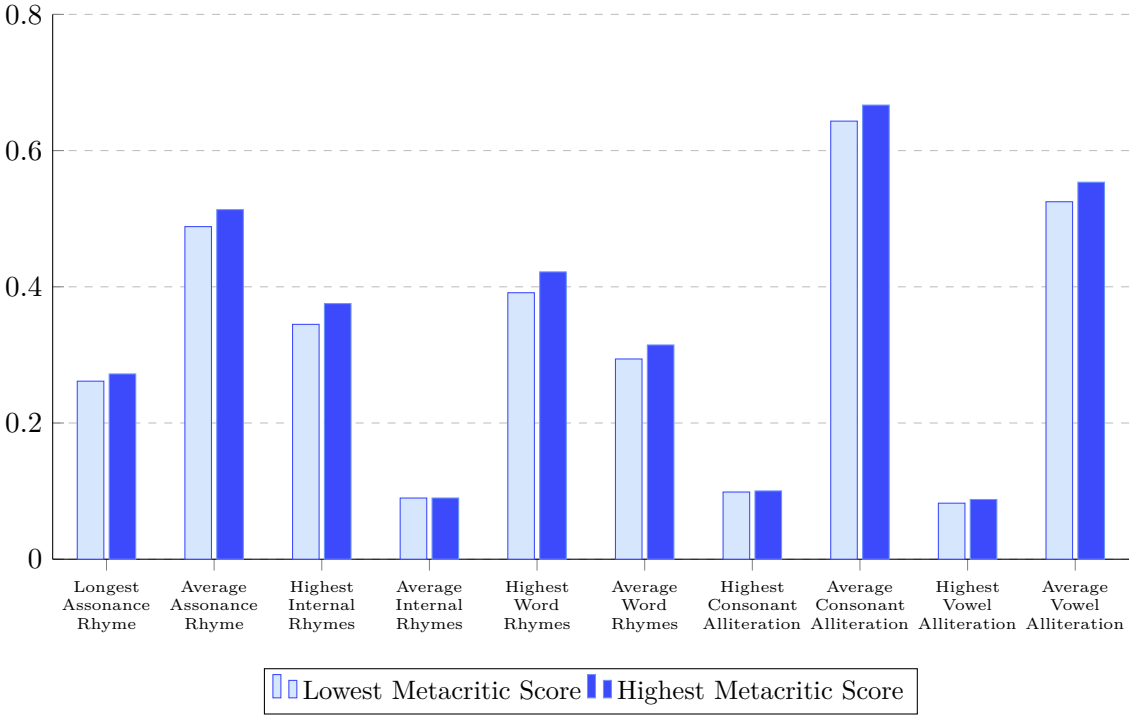


Figure 5.3: Rhyme metric score for each respective rhyme category, sorted by Metacritic score.

fairly similar as with Metacritic score. However, there is an even larger disparity in rhyme complexity when comparing the highest rated half of songs to the lowest rated in user score, displaying a gap of 0.21 points in rhyme complexity score. This indicates that users of *www.metacritic.com* tend to react more favorably towards music with more complex rhyme structure.

The scatter plot displayed in Figure 5.7 show the general trend between user score and rhyme complexity. When examining the trend line, this further emphasizes the indicated weak positive correlation between user score and rhyme complexity. This trend line exhibits the same incline of $y = 0.0085x$ as with Metacritic score, despite these being independent metrics as previously pointed out. This time the R^2-value stands slightly higher at $R^2 = 0.059$, however, this increase bears no significance on the quality of the interpolation or the accuracy of the displayed trend.

Similarly, as with the notable outliers in the Metacritic scatter plot in Figure 5.2, the songs that display the single highest and single lowest rhyme metric score can be found in the opposite side of the user score scale, albeit with a small difference, standing at 82 and 83 user score respectively.

In the same vain as with Metacritic score, the contribution of each individual relative rhyme metric for the highest rated and lowest rated half of songs by user score can be seen in Figure 5.6.

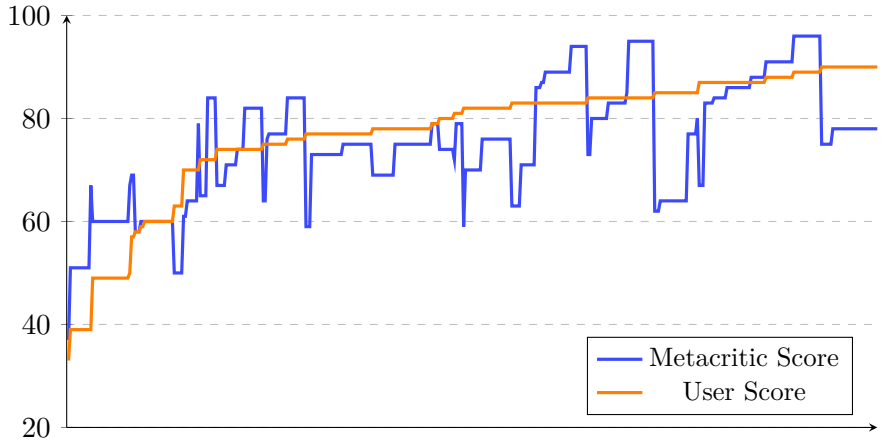


Figure 5.4: Similarities between user score and Metacritic score.

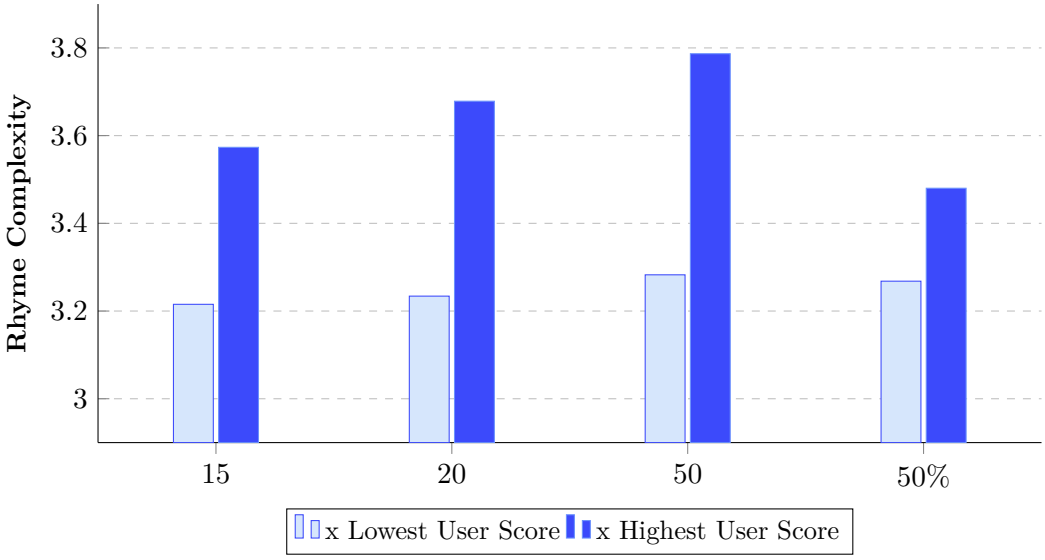


Figure 5.5: Correlation between rhyme complexity and user score for the x highest and lowest rated songs by user score.

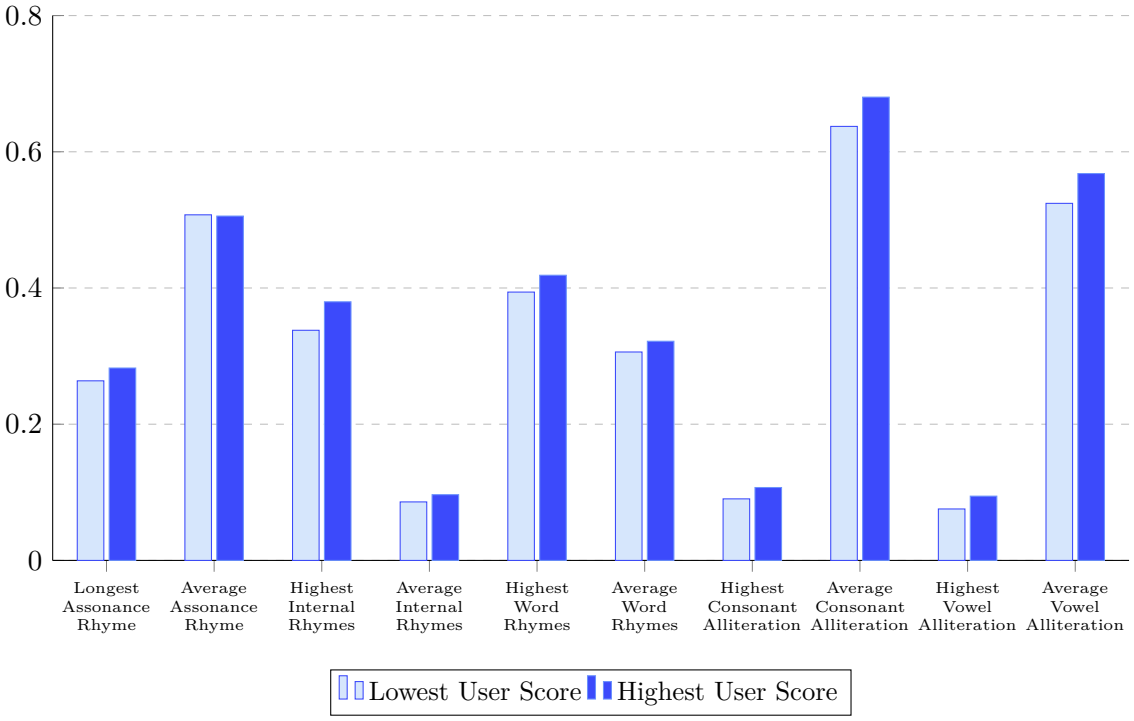


Figure 5.6: Rhyme metric score for each respective rhyme category, sorted by user score on *www.metacritic.com*.

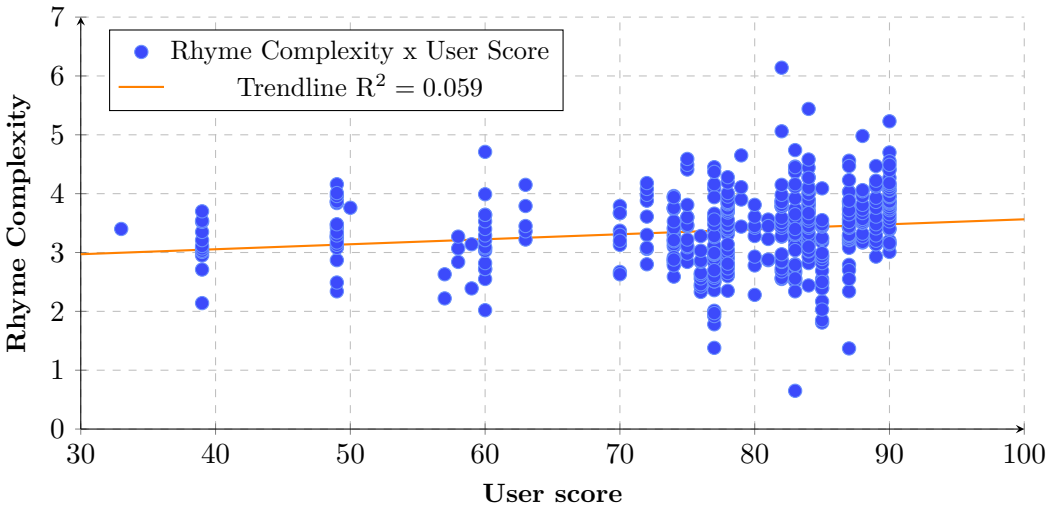


Figure 5.7: Correlation between rhyme complexity and user score, also showcasing a trend-line for the general trend in this correlation.

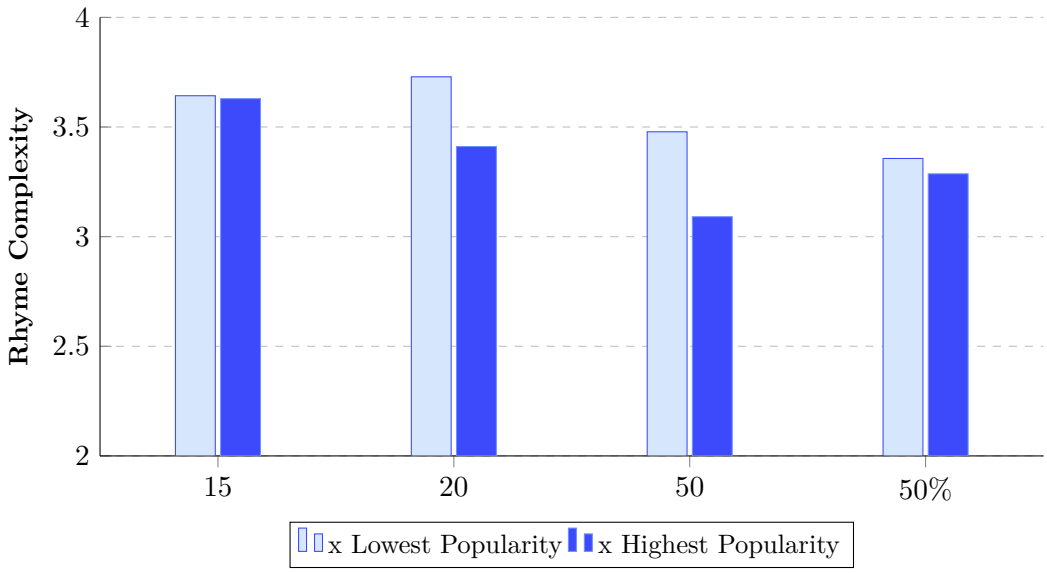


Figure 5.8: Correlation between rhyme complexity and popularity for the x highest and lowest rated songs by user score.

5.2.3 Rhyme Complexity and Popularity

In direct contrast to the correlation seen with Metacritic and user score, the correlation between popularity and rhyme complexity displays a weak negative correlation, indicating that there is a general tendency for more popular songs to display a lower rhyme complexity. The difference in the rhyme complexity of the x highest rated songs and the x lowest rated songs by Spotify's popularity metric (by album) can be seen in Figure 5.8.

In every segment (each pair of columns by a given x) there i a clear disparity, showcasing the tendency of less popular songs being more complex in rhyme structure. This discrepancy is however declining by each successive x, given that a larger x means less polarization in the data and inclusion of more of the middle range of the popularity scale. When comparing the lowest rated half of the song selection with the highest rated half with regards to popularity, the difference in rhyme complexity is a meager 0.08 points, in sharp contrast with the highest and lowest 15 being divergent by 1.02 points.

From the scatter plot in Figure 5.9, it is evident that there are large variations within each popularity segment, with the trend line having an R^2-value of $R^2 = 0.004$. However, the trend line shows definitively that there is a weak negative correlation of $y = -0.0042$. This negative correlation is merely half of the incline of the corresponding trend lines for both user score and Metacritic score with regards to rhyme complexity, indicating that correlation between popularity and rhyme complexity is more limited than that of rhyme complexity, and Metacritic score and user score.

In the case of popularity, each relative rhyme metric does not contribute to the negative correlation. In some instances the rhyme metrics of the lowest rated half of songs by popularity score lower than the highest rated half. This can be seen in Figure 5.10, with

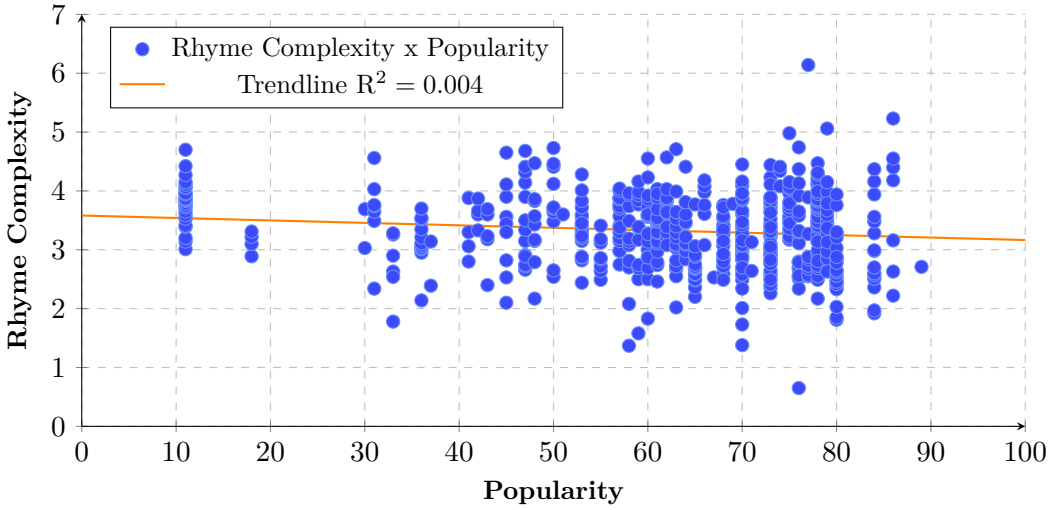


Figure 5.9: Correlation between rhyme complexity and popularity, also showcasing a trend-line for the general trend in this correlation.

for instance the longest assonance rhyme metric scoring higher for higher rated songs by popularity, this is in contrast with the overall rhyme complexity discrepancy of -0.08 seen in Figure 5.8.

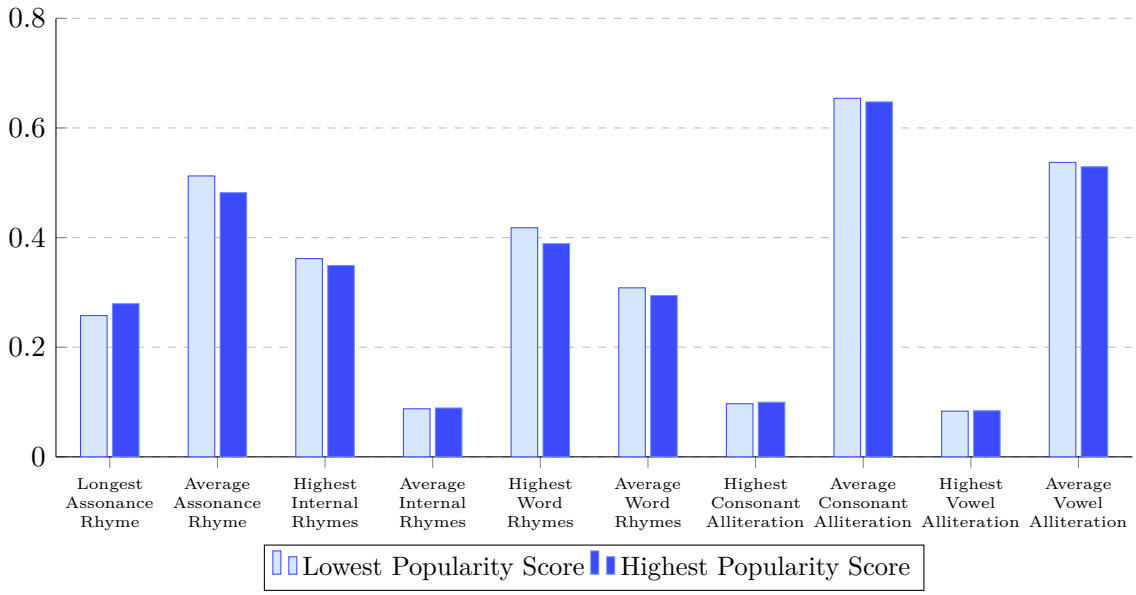


Figure 5.10: Rhyme metric score for each respective rhyme category, sorted by popularity metric on Spotify.

6 Experiment 2: AI Generation of Rap Lyrics

In this chapter, there will be a detailed presentation of the research method and results from Experiment 2, regarding generation and evaluation of hip hop lyrics. During the presentation of the results, some areas of interest will be pointed out and some inference about the implications of the results will be drawn, when deemed appropriate. In turn, a more detailed discussion over the implications and validity of the results will be deferred to Chapters 7 and 8 regarding discussion, evaluation and conclusion of both experiments.

The objective of Experiment 2 was to generate hip hop lyrics phrases using *Artificial Intelligence* (AI), to determine whether or not it is possible to generate these phrases to be as good as existing lyrics from popular and critically acclaimed rap songs. To ascertain how *good* the generated lyrics are in relation to existing lyrics, the quality of both generated phrases as well as phrases from existing songs was assessed by humans through quantitative evaluation. In addition to human evaluation, each phrase was scored on the same rhyme complexity scale that is presented in Section 4.1.1, and utilized in Experiment 1 and described in detail in Sections 4.1.2 and 5.1.

6.1 Setup of Experiment and Lyrics Generation

The lyrics was generated using a *Long Short-Term Memory* (LSTM) Network and trained on a comprehensive catalog of existing rap lyrics, described in more detail in Section 4.2. Subsequently the lyrics underwent some manual editorializing to meet the standardization criteria for all phrases used in the survey for quantitative evaluation, described in more detail below. The generated phrases utilized in the evaluation survey were all generated with the start seed *"ain't shit"*. This start seed is a combination of two of the most frequently used words in the hip hop lyrics dataset, revealed by a term frequency algorithm, the top 100 results of which can be seen in Appendix A, where "ain't" and "shit" appear as number 11 and 10 respectively.

6.1.1 Selection of Generated Phrases

There were essentially generated thousands of phrases over the course of this thesis, where a majority of them displayed little to no cohesion and contained obvious grammatical mistakes and misplaced words that would make it easily discernible to detect that they were generated using AI. Over the course of a couple of weeks with hyperparameter

Raw Output	Standardized Output
aint't shit where my dick too cool my whole hectic when you're trippin' and always bullshittin'	Ain't shit where my dick Too cool my whole hectic When you're trippin' And always bullshittin'
i'm smokin' genius now we live so why do it to this bad ass bitch drank repeat too much flo' of out the dirt try to kill 'em now	I'm smokin' genius, now we live So why do it to this bad ass bitch Drank repeat, too much flo' Out the dirt, try to kill 'em now

Table 6.1: Lyrics displayed as raw output from LSTM module and as standardized output for survey evaluation purposes.

optimization[1], the output gradually became more satisfactory. To avoid the most glaring telltale signs of AI generated lyrics, all phrases selected for the survey were selected manually, with cohesion, rhyme structure and grammar in mind.

6.1.2 Phrase Standardization

As mentioned briefly above, all the phrases; both generated and existing, were standardized for the survey. One instance of phrase standardization can be viewed with a phrase displayed initially in raw output format, and in turn standardized with more reasonably distributed line changes, added punctuation for better readability and capital letters for each new line.

6.2 Research Method

Addressing directly *Research Question* **RQ 2**, regarding generation and evaluation of hip hop lyrics phrases, the second experiment concerns the task of lyrics generation through the use of AI. Goal 2 from Chapter 1 would ultimately see a system that generates rap phrases that are as good as lyrics from existing popular and critically acclaimed rap song. Whether or not this benchmark was achieved was determined through human evaluation, as well as quantitative assessment through the rhyme complexity framework introduced in Sections 4.1.2 and 5.1.

After the lyrics were generated, as described in Sections 4.2 and 6.1, the generated phrases underwent quantitative evaluation through a survey, alongside phrases from existing rap songs. Each of the phrases were evaluated by five different metrics, and the results presented take form in statistical analysis of the evaluation of the phrases. The details and contents of the evaluation survey are described over the following section.

[1]Hyperparameter optimization is the process of tweaking individual parameters for the training phase of your machine learning module.

Type of phrase	Number
Phrases Generated by AI	9
Phrases from popular songs	3
Phrases from unpopular songs	3
Phrases from critically revered songs	3
Phrases from critically despised songs	3

Table 6.2: Number of different types of phrase being used in the survey for evaluating quality of generated lyrics.

6.2.1 Rhyme Complexity of Generated Phrases

Having already conducted research regarding lyrics analysis and rhyme complexity in Experiment 1, the opportunity was also present of calculating rhyme complexity for the generated phrases in comparison with existing rap lyrics and the existing phrases used in the evaluation survey. This opened for the possibility of discerning whether the generated phrases share some of the characteristics more frequently displayed in popular and critically acclaimed songs. This will be a quantitative experimental conduction in the same vein as Experiment 1, however with more emphasis on the characteristics of generated lyrics in comparison with existing lyrics.

6.2.2 Description of Evaluation Survey

To be able to compare the quality of the generated phrases in relation to existing phrases, a survey was created using a selection of phrases as seen in Table 6.2. Here the *popular* phrases are taken from songs with a Spotify popularity score between 79 and 86, unpopular on the other hand are songs with a popularity score between 18 and 42. The critically acclaimed phrases were taken from songs from albums with Metacritic score between 89 and 96, while critically despised phrases were taken from songs from albums with Metacritic score between 50 and 60, which is definitely at the lower end of the critical spectrum. The setup of the survey can be seen in Appendix B, and a complete overview of the phrases included in the survey can be found in Appendix C.

To determine whether or not the lyrics was as good as existing lyrics from popular and critically acclaimed songs, there was devised a set of five metrics to determine overall quality of lyrics phrases. These metrics were *general quality* (first impression), *rhyme complexity* (perceived complexity of rhyme scheme), *cohesion / meaningfulness* (does the phrase convey a meaningful and coherent message), *grammar* and *AI / human generated* (whether the lyrics appear to be written by a human or generated by an AI driven system). Seeing as the knowledge of whether or not art is generated by computers tend to affect people's perception of the art itself, this last criteria of human vs AI generated was a point of particular interest when reviewing the answers to the survey.

To mitigate the difference between generated phrases and existing phrases, all phrases were standardized with capital letters at the start of new lines, and commas were added for better readability.

6.2.3 Participation in Survey

Also included in the survey was fields identifying the age group, relationship with hip hop and background with artificial intelligence of the surveys participants. A detailed overview of this data regarding the 43 participants of the survey can be found in Appendix D. The survey was mostly distributed through fellow students, family members and acquaintances and spans a fairly wide age range, knowledge about AI and relationship with hip hop, although with a slight angling towards people between 18-25 years with a lower to median knowledge base for AI and fairly middle ground relationship to hip hop music. The ramifications and limitations of the stated participation in the survey will be further discussed in Section 7.4.

6.3 Result of Experiment 2 - Hip Hop Lyrics Generation

The hip hop lyrics generation culminated in a selection of nine phrases that were to be evaluated alongside 12 phrases from hip hop songs belonging to one of four categories; *popular*, *unpopular*, *critically revered* or *critically hated*. A complete overview of the selection of phrases can be found in Appendix C. The existing rap songs, from which the 12 existing phrases were taken, were chosen at random from their respective categories. The phrases were then chosen such that they would be individually coherent, and preferably not easily recognizable. To mitigate the possibility of the phrases being recognized, they were all taken from either the second or third verse of the song.

Along side the quantitative evaluation of the phrases, all the phrases used in the survey were run through the rhyme complexity framework used in Experiment 1. A detailed presentation of the results for both these rhyme complexity calculation and findings from the responses to the survey will be presented over the following subsections.

6.3.1 Rhyme Complexity for Generated Phrase

As mentioned in Chapter 4, parts of the objective for the analysis of existing rap lyrics was to be able to evaluate quality of generated phrases by the same merits. The framework designed to evaluate complexity of rhymes provides an opportunity to compare the rhyme complexity of the generated phrases with existing lyrics, particularly the other phrases used in the evaluation survey. When comparing the average score of each rhyme metric for the generated phrases, with the entire song lyrics catalog from Experiment 1 separated into the four distinct categories of existing lyrics; *popular*, *unpopular*, *critically revered* and *critically despised*, a clear pattern emerges.

For every rhyme metric, with the notable exception of average consonant alliteration, the average score for the generated phrases turned out to be lower than the average of each of the four other categories, as can be seen in Figure 6.1. Unsurprisingly, the same relation can be seen for rhyme complexity score which is merely the sum of all other rhyme metrics. Even when comparing the highest value of each rhyme metric in any of the generated phrases to the other categories, the rhyme complexity score is far below the averages of all other categories. This can be seen in Figure 6.2. There are, nevertheless

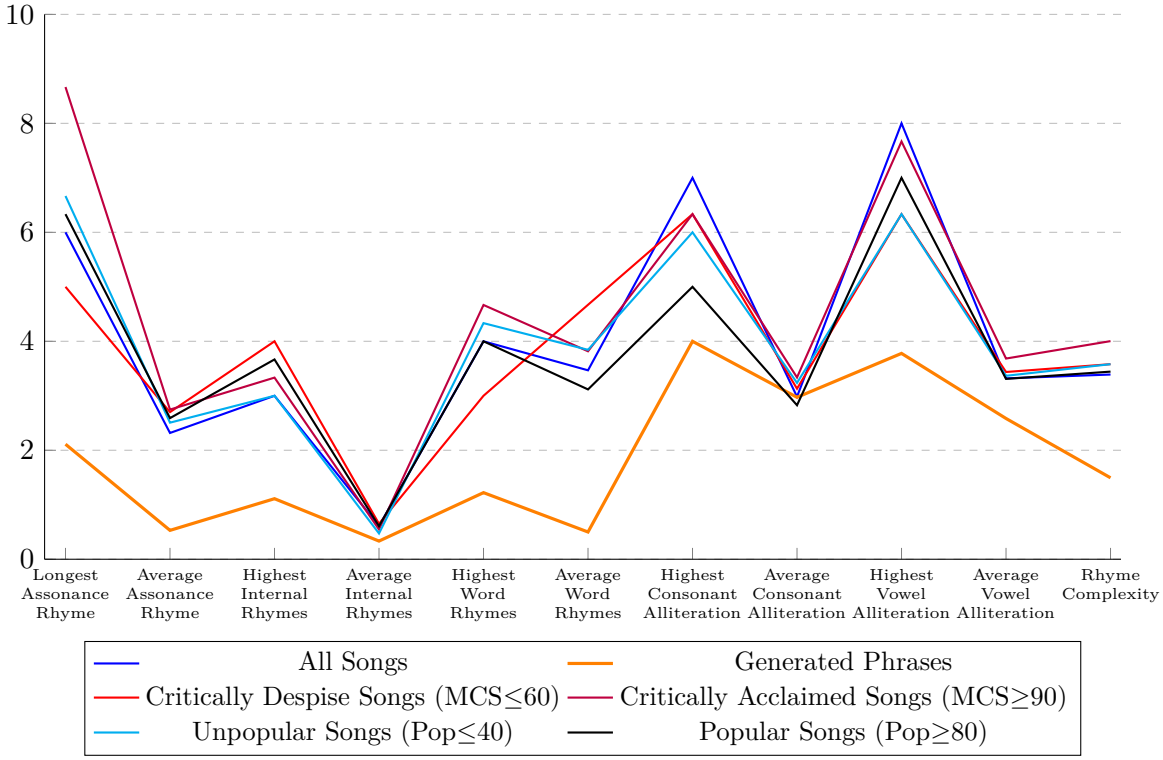


Figure 6.1: Comparing rhyme metric scores from the average of generated phrases with lyrics from popular, unpopular, critically acclaimed and critically despised songs.

multiple metrics in which the highest value of the generated phrases score comparably with the averages of existing lyrics, most notably in all the *alliteration* metrics, as well as with the *average internal rhymes* metric.

What can be deemed from this presentation of rhyme stats for generated phrases and existing song lyrics is that although the rhyme metrics for generated phrases on average score lower than existing lyrics, in some instances the phrases still excel and perform well above average for some categories. The generated phrases have a clear disadvantage in being far shorter than most of the existing song lyrics, which impacts the rhyme complexity score, as the rhyme complexity framework rewards longer lyrics more highly than shorter phrases. This will be further discussed in Section 7.3.4 regarding discussion and evaluation of the results.

When comparing the score of the generated phrases with other individual phrases used in the survey, there appears to be less discrepancy between the rhyme metrics between the five categories of generated phrases, popular phrases, unpopular phrases, critically acclaimed phrases and critically despised phrases. The average score of all songs in the rap lyrics catalog are also displayed, for comparison in Figure 6.3. The all songs category is in all but three metric yields significantly higher averages than any of the

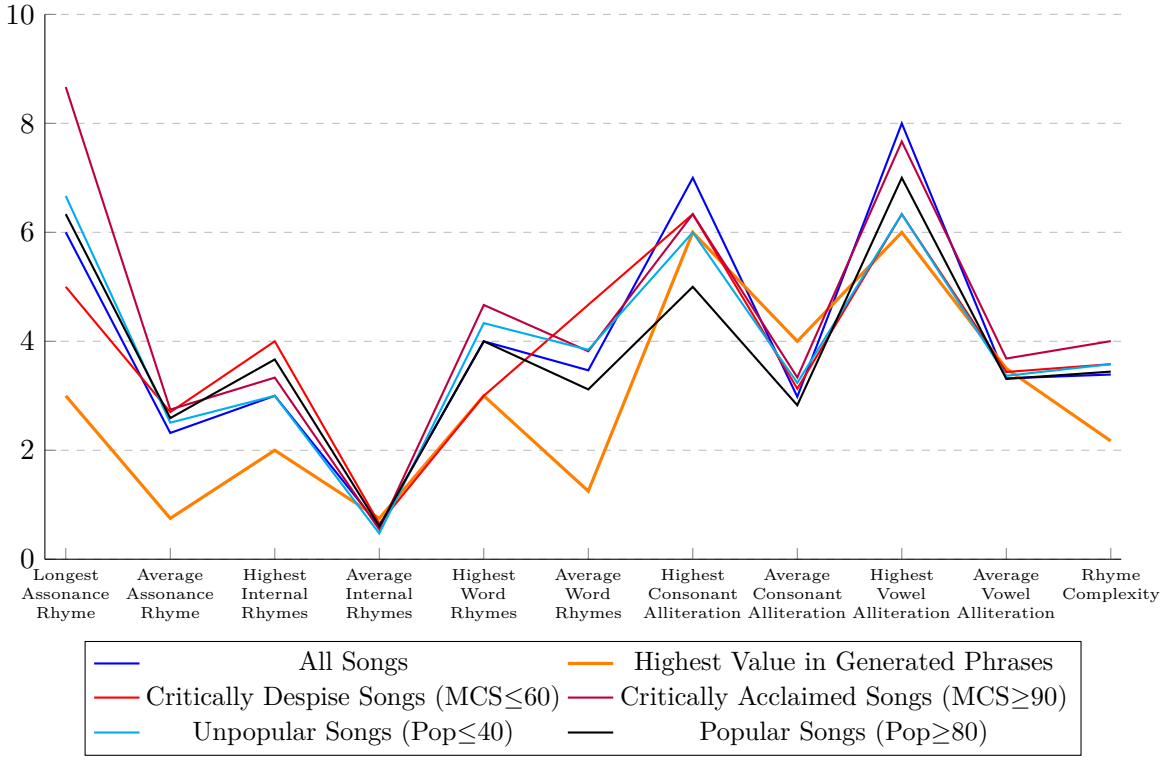


Figure 6.2: Comparing rhyme metric scores from the highest value for ant of the generated phrases with lyrics from popular, unpopular, critically acclaimed and critically despised songs.

five aforementioned categories. However, what can be deemed is that when the existing phrases are all between four and five lines in length, they do not have the clear advantage of significantly more and longer lines when compared to the generated phrases. In this case, the generated phrases outperform every other category in at least one metric, it does however ultimately under perform all of the other categories, as can be seen by this category displaying the lowest overall rhyme complexity score.

What more can be seen is the clear correlation between perceived rhyme complexity though human evaluation and system calculated rhyme complexity through the rhyme complexity framework. The comparison between the two and adjacent correlation can be seen in Figure 6.4, showing a decisive trend line, although with slightly scattered data points. This insight might be valuable when evaluating the validity of the rhyme complexity framework over the following chapters.

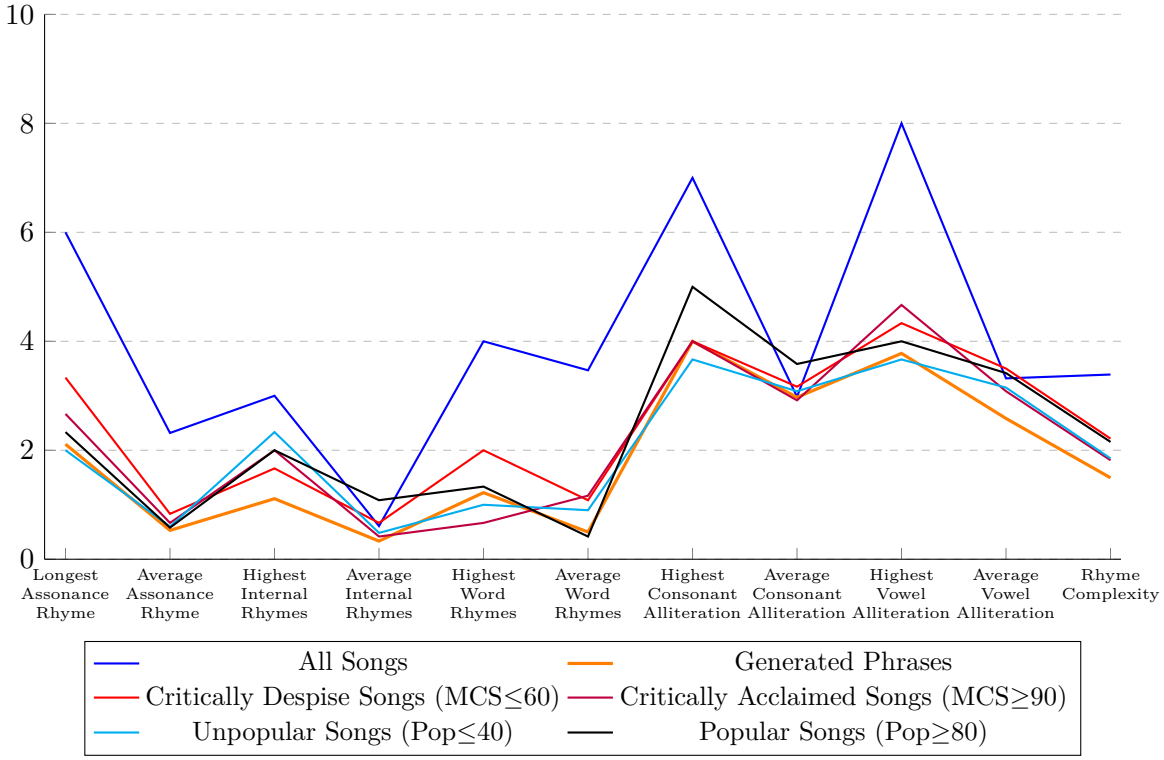


Figure 6.3: Comparing rhyme metric scores from the phrases used in the survey.

6.3.2 Results From Survey

The 21 rap lyrics phrases were to be evaluated on five distinct metrics; general quality, rhyme complexity, grammar, cohesion/meaningfulness and whether the participants believed the phrases to be generated by AI or by a human. The reason for separating the evaluation into these five distinct metrics were to see if any of the metrics would be more consequential for people's general perception of the phrases, as well as determining whether any one of the metrics would be key for the perception of a phrase of lyrics as being generated by AI or a human.

When looking at the results from the survey, there emerged a pretty definite pattern. In Figure 6.6, there is a display of the average score for every metric for each of the five categories of phrases. As it turns out, the highest scoring category for every metric was phrases from *critically despised* songs by quite a large margin. On the other side of the critical spectre, the *critically acclaimed* phrases frequently score lower than the *popular* phrases, and in one instance even *unpopular* phrases. These findings yield no significant insight in people's enjoyment of the songs, as this is a fairly clinical assessment of the rap genre, based solely on a selection of phrases from lyrics.

What is of most interest for this thesis is the consistently low scores for all metrics

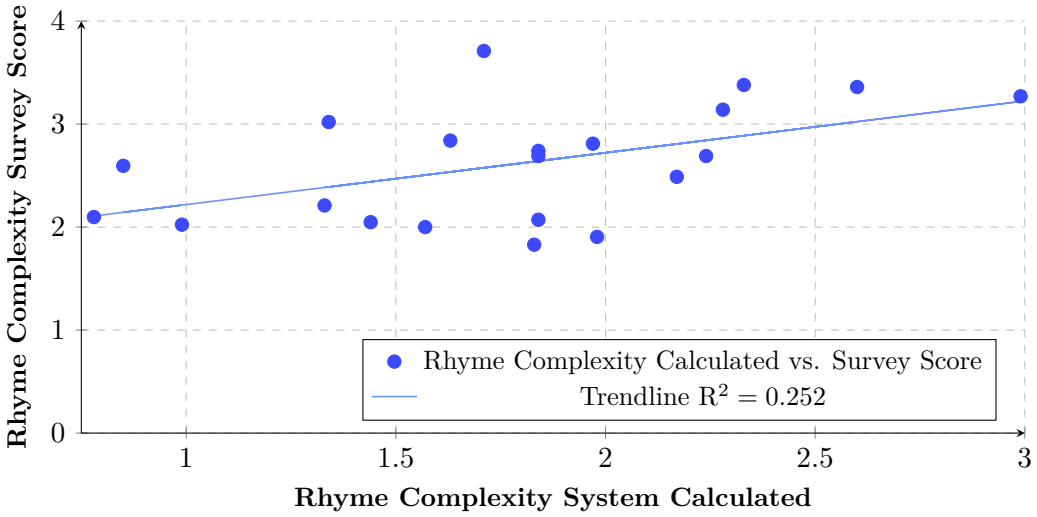


Figure 6.4: System calculated rhyme complexity when compared with human evaluated rhyme complexity.

for the generated phrases, as can be clearly seen in Figure 6.6. This was the case also when separating the responses in segments of all responses with a reported *relationship to hip hop* of 1 or 2, all responses with a reported *knowledge about Artificial Intelligence* of 1 or 2, and all responses. Observe in Figure 6.5 that both the segments of little to no knowledge about AI and relationship to hip hop consistently evaluate the generated phrases higher in every regard, when compared to overall average from all responses. This may imply that people with a self perceived low interest for hip hop or AI tend to evaluate AI generated hip hop lyrics more favorably.

6.3.3 Correlation between Survey Metrics

There is a fairly low variation between the scores of the different metrics for each respective category of lyrics. For instance in the case of *critically despised* songs, where each metric sits at a score of between 3.48 and 3.72. and generated phrases which merely fluctuate between 2.12 and 2.49. While this is the average score of all the phrases in the category, and the individual answers may vary in each metric, this indicates that there is a correlation between each of the metrics, and in fact, when comparing the *general quality* metric with each of the other metrics, a clear correlation emerges.

In Figure 6.7, there can be seen a clear correlation between perceived rhyme complexity and perceived general quality of the phrases. While this might not be all that surprising, it may provide some insight into how rhyme complexity, grammar and meaningfulness impact the overall perception of lyrics, specifically in the rap genre. This notion will be further explored and discussed in Section 7.4.

A similar same correlation can be seen between general quality and all other metrics evaluated, as well ass the overall score when averaging all the metrics for evaluation

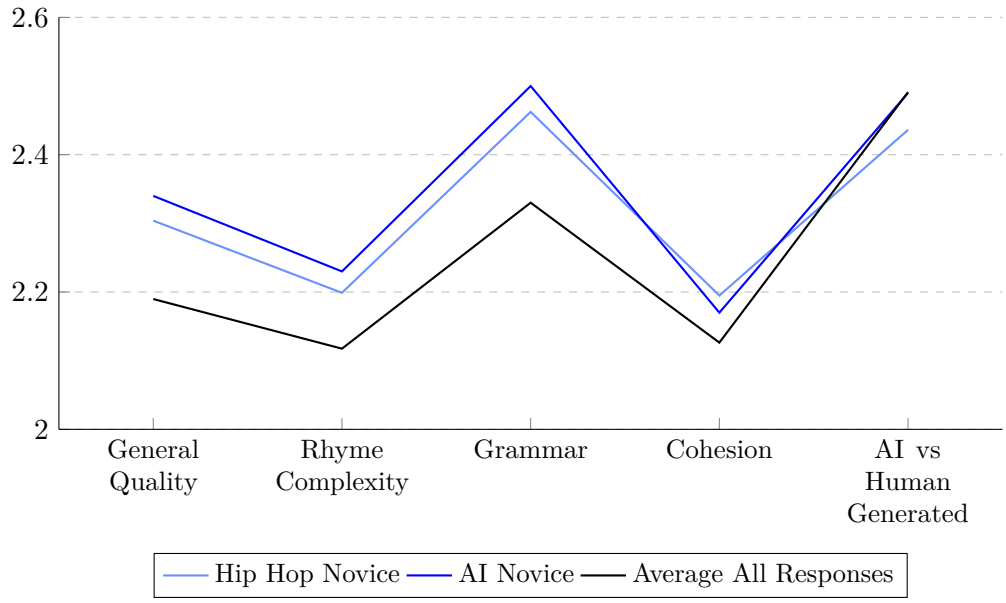


Figure 6.5: Comparing the average score of each metric for all generated phrases, when separating responses from people with little to no relationship with hip hop, and little to no experience with AI, from the average score from all responses.

of phrases. The graphs displaying these correlations can be seen in Appendix E. In short, these strong correlations between general quality and all other metrics suggest that the first metric the participant were asked to evaluate, *general quality*, is either very indicative of what all other metrics mean for the overall quality of the phrases, or being the first metric to evaluate, this simply colors the opinion when answering the rest of the survey. This notion will be further elaborated in Section 7.4.

6.3.4 Dispersion of Responses to Individual Phrases

Below follows three Tables, 6.3, 6.4 and 6.5, showing the dispersion of responses to three of the phrases used in the survey, similar tables for the remaining phrases can be found in Appendix F. The first two are from the two generated phrases that on average scored the highest over all metrics combined among the generated phrases, scoring 2,45 and 2,58 respectively. The last phrase is from the critically acclaimed song *Same Drugs* by *Chance the Rapper*, a phrase that scored towards the middle of the pack among the phrases taken from existing songs, with an *Average All Metrics* (AAM) score of 3,01. The AAM score of the rest of the phrases used in the survey can be found in Appendix G, along with averages for all other metrics for all phrases used in the survey.

What can be seen is a wide fluctuation in the scores given to each phrase, regardless of origin. All three phrases presented here have been scored on both extremes for every

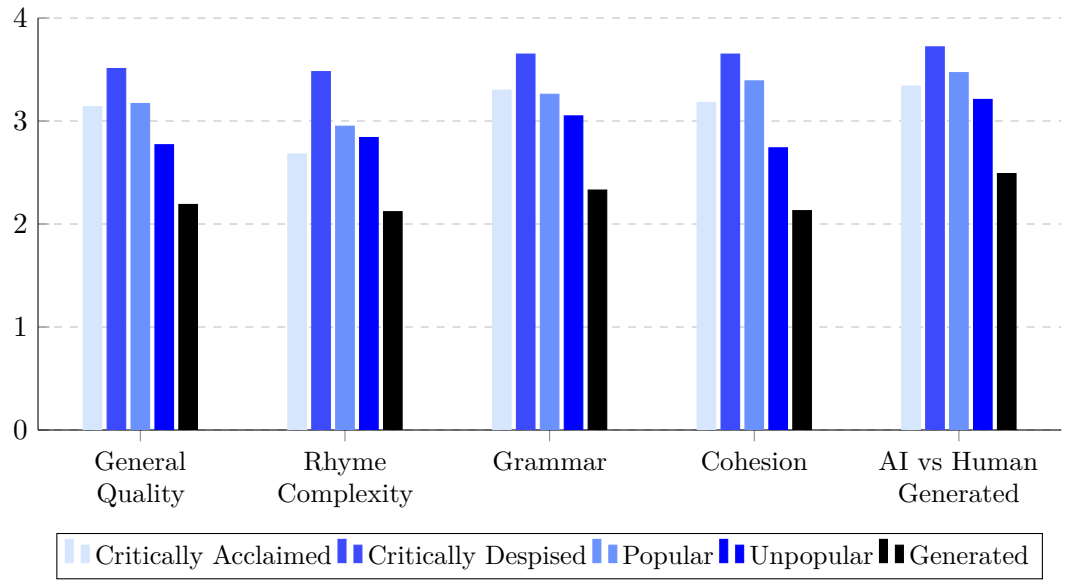


Figure 6.6: Calculated averages for every metric used in the survey for each category of lyrics.

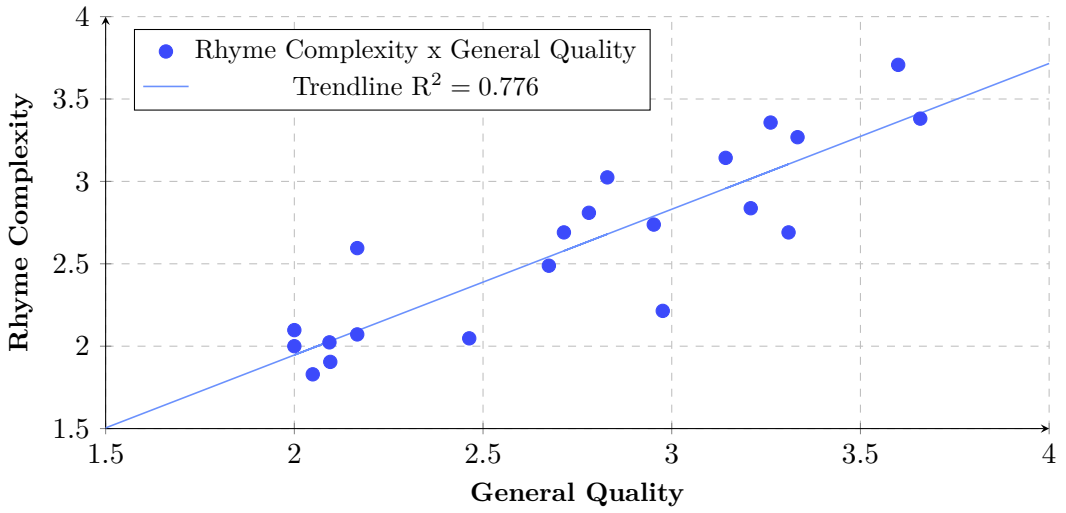


Figure 6.7: Correlation between perceived rhyme complexity and perceived general quality of phrases from survey.

category. Interestingly, every phrase used in the survey have been scored on both extremes in the category *AI / Human Generated*.

I'm smokin' genius

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	12	19	2	1
Rhyme Complexity:	15	12	14	0	1
Grammar:	7	17	14	3	1
Meaningfulness:	9	15	12	5	1
AI/Human Generated:	5	13	5	13	5

Table 6.3: Dispersion of responses to the generated phrase *I'm smokin' genius*.

When I sit through

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	10	18	6	2
Rhyme Complexity:	6	18	13	4	2
Grammar:	7	12	16	5	1
Meaningfulness:	9	10	17	4	2
AI/Human Generated:	10	9	10	10	3

Table 6.4: Dispersion of responses to the generated phrase *When I sit through*.

Chance the Rapper - Same Drugs

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	4	10	12	15	1
Rhyme Complexity:	11	15	13	2	1
Grammar:	2	5	12	15	8
Meaningfulness:	2	8	12	13	7
AI/Human Generated:	9	6	11	9	7

Table 6.5: Dispersion of responses to the phrase from the critically acclaimed song *Same Drugs* by *Chance the Rapper*.

7 Discussion and Evaluation

Over the course of this chapter we will outline the implications of the results from both Experiment 1 and Experiment 2. There will be a discussion about the limitations of the method and conduction of the experiments, as well as the validity of the results. Inherent in both experiments are a number of limitations and points of interest that may affect the validity of the findings and offer additional inference to the initial intention of them, thus to be better suited for drawing a final conclusion, these points will be detailed and discussed.

As Experiment 1 was presented and executed first, Section 7.1 will be primarily dedicated to concerns for this specific experiment. After, there will be a discussion about the merits, validity and potential for the *rhyme complexity framework* utilized in both experiments, before points of interest for Experiment 2 will be discussed. This structure allows for a conversation about the impact of the results of each experiment on the other. Consequently, Section 7.5 will be dedicated to discussion about the implications of the two results upon each other.

7.1 Lyrics Analysis

The findings during experiment 1 paint a clear picture of the correlation between rhyme complexity lyrics and perception and consumption of the songs the lyrics hails from. The results indicate clearly that there is www.a positive correlation between rhyme complexity and Metacritic score, as well as user score on *www.metacritic.com*, and a negative correlation between rhyme complexity and popularity.

There is however, a set of interest points and concerns that warrants some discussion to be able to more successfully conclude the findings of the lyrics analysis. Every aspect from collecting the lyrics dataset to the phoneme conversion and development of mechanisms to identify rhymes might significantly influence the outcome of the experiment.

7.1.1 Dataset for Analysis

For the analysis of the lyrics, one of the premiere points of concern for the results, is the dataset that was used. While the songs used in the dataset are varied and diverse in as far as artist notoriety, popularity, critical acclaim and time of release, it may still misrepresent the vast span in rhyme complexity of the rap music genre. It also warrants to be pointed out that the selection for this experiment is fairly small for the time being, particularly in the lower end of the Metacritic score and the higher end of the popularity metric.

Metric:	Metacritic Score:	User Score:	Popularity:
Number of Songs:	598	437	563
Highest Score:	96	90	89
Median:	76	82	66
Lowest Score:	37	33	11

Table 7.1: Number of songs in the three categories; Metacritic score, User score and popularity, with associated extreme and median values.

The dispersion of songs used for analysis can be seen in Table 7.1. The dataset used for analysis consist of in total 853 songs, of which all are being run through the rhyme complexity framework. However, given the inclusion of artists and albums from different degrees of notoriety, not every song have got an associated critic's score, nor a user score, nor a presence on Spotify and thus not a popularity measure. Therefore, not every song contributes directly to the data presented in Section 5.2. It is reasonable to believe that any substantial expansion or alteration to the selection of lyrics that is being analyzed could yield non-trivial variations in the results.

7.1.2 Phoneme Conversion

The quality of the analysis of rhymes for the lyrics is highly impacted by the quality of the word-to-phoneme conversion. All the lyrics are firstly converted into *International Phonetic Alphabet* (IPA) using python's built-in *Natural Language Tool Kit* (NLTK) library, as described in Section 4.1.3. This method is currently far from perfect, yielding an average phoneme conversion rate of 92% of all words over the entire branded lyrics dataset. When a word is not able to be translated into IPA, the system simply skips this word, which means that standing at 92%, for evey 100 words in a song, 92 is converted properly, while 8 words are disregarded outright.

This may significantly impact the rhyme structure and rhyme scheme of each individual song. To combat this, a specialized dictionary was created, containing the 50 most commonly missed words during the conversion and their IPA translation. This way the system was able to boost phoneme conversion hit rate by 1%, increasing overall word-to-phoneme conversion hit rate to 93%, although, even with this 1% conversion bump, the translation is far from perfect.

7.1.3 Pronunciation and Dialects in Hip Hop

Another aspect that yields a significant impact on the quality of detecting rhymes are all the different pronunciations and dialects used in the genre of hip hop. As the system in this thesis is purely text-based, the ability to detect artistic liberties and interpretations of pronunciation to achieve rhymes is not present. To be better able to detect these sort of audible yet not textual rhymes, a system was in place to give the interpretation of textual lyrics the benefit of the doubt. If there was detected a rhyme, given the definition

Phoneme	NLTK Notation	Alt. Pronunciation	Alt. Pronunciation (NLTK)
oʊ	'OW'	ɑ	'AA'
aʊ	'AW'	ɑ	'AA'
ʉ	'AO'	ɑ	'AA'
i	'IY'	ɪ	'IH'
eɪ	'EY'	ɛ	'EH'

Table 7.2: List of phonemes that are deemed to have a different pronunciation, and that will constitute a rhyme if substituted.

of a rhyme given in Section 4.1.1, except the vowel phonemes were not identical, a list of alternative pronunciations of some choice vowel phonemes was in place. This allows the system to interpret these phonemes in different ways if it is reasonable to believe that words are intended to rhyme audibly, yet they do not rhyme textually. These alternative pronunciations can be seen in Table 7.2.

Even given this list of alternative pronunciations and the added conversion hit rate from specialized hip hop word-to-phoneme dictionary, there may still be significant gaps in rhyme detection. While the same limitations will be present for every lyrics, it is unclear whether or not this affects certain artists more than others. While the ramification of words missed during phoneme conversion do not exclusively impact rhyme scores negatively, it is still worth noting that the rhyme complexity score given does not represent the full extent of the artists intent and audible delivery.

7.1.4 Identifying Rhymes

Despite the inherent weaknesses in text-to-phoneme conversion and textual interpretation of audible output, the methods used for identifying rhymes are deemed to be very thorough and precise. When simply looking at the phoneme representation the system is able to generate from existing textual lyrics, the line-for-line process of identifying rhymes is not prone to missing any of the types of rhymes it is attempting to identify, described in more detail in Section 4.1 and discussed more over the following section.

7.2 Rhyme Metrics and Rhyme Complexity

Given the fact that there currently exists no widely used universal framework for rhyme complexity there is little grounds for comparison. The framework presented in this thesis is an attempt at rewarding different sorts of rhymes as well as a combination of both long term and short term complexity. Although the rhyme metrics chosen here are intended to be a wide selection of rhyme types, there are still areas of rhyme that are not reflected in the rhyme complexity score for this thesis. This will be elaborated upon in Section 7.2.2. What is even more noteworthy is the fact that all rhyme metrics in the rhyme complexity framework are scored relative to the rhyme metric scores of all other songs in

the lyrics analysis dataset. This means that the score presented is not a universal score, but merely relative to the specific domain of the analysis. This and other limitations will be discussed briefly below.

One substantial conundrum that arose over the work on this thesis was with the problem description itself stating that the generated lyrics were to be "better that existing lyrics from popular and critically acclaimed songs". As is evident with the presentation of result from Experiment 1 in Section 5.2, there is no clear correlation between the two with regards to rhyme complexity. In fact there appears to be a directly inverse correlation that makes it more prevalent to restate the question "is it possible to generate rap lyrics phrases that are better than lyrics from existing popular ***or*** critically acclaimed rap songs?".

7.2.1 Limitations of Rhyme Metrics and Rhyme Complexity

Given the fact that the rhyme metrics contained in the rhyme complexity framework are scored relative to other songs in the dataset used, the results of the analysis are very beholden to the dataset used. A natural extension of this framework would be to create a universal framework that could be used to compare rhyme complexity over different domains, *e.g.* between different genres of music.

As can be seen in Figures 5.3, 5.6 and 5.10, the different types of rhymes contribute in varying degree to the overall rhyme complexity score. This is a consequence of the fact that some metrics have a larger span in the scores across the analysis dataset, and have some notable outliers that makes the relative metric score for most songs in the dataset minuscule. For instance the highest vowel alliteration metric for one line in a song ranges from 2 to 70, while 95% of the dataset have a score of 12 or less in this category. This means that almost every song in the dataset score very low on highest vowel alliteration, because the relative scoring mechanism do not account for the ramifications of such outliers. A mitigation strategy for this problem could be to account for length of single lines in the eventual scoring.

All rhyme metrics are scored on a line for line basis, which brings to attention a fundamental point of concern in the current framework. More than anything else, the framework rewards long lines of lyrics, as is evident by the aforementioned example with vowel alliteration. The reason this tendency has been deemed acceptable for this thesis is the justification that writing long lines of lyrics could also be seen as a sign of complexity in lyrical writing. However, this concern of disproportionately rewarding rhyme complexity in longer lines of lyrics could be mitigated by dividing the metrics in some manner by the length of each line.

7.2.2 Types of Rhymes Not Accounted for in Rhyme Complexity Framework

The metrics chosen for the rhyme complexity framework in this thesis include all significant types of rhymes traditionally used in hip hop lyrics, as described in Section 2.1 regarding hip hop theory. There are, however, some ways these types of rhymes could be expanded

upon in the framework to contribute to the rhyme complexity score in more detail and more consequently.

For instance, even though vowel and consonant alliterations are both covered by the rhyme complexity framework, each of these metrics only reflect the highest number of vowel and consonant alliteration for one single line. This means that if there are more instances of alliteration for one single line, these will not affect the overall rhyme score. A metric that reflects the alliteration density could be implemented to account for multiple instances of high value alliteration for one single line. The same goes for assonance rhymes, in which each line only accounts for the single longest assonance rhyme between one line and the subsequent lines. Including a metric reflecting the density of all assonance rhymes for each line may paint a more complete picture of rhyme complexity.

Another metric that could be of interest, yet is not covered in the current version of the framework, are length of internal rhymes and word rhymes. However, these metrics was not a concern of this thesis, but would be interesting to include in the framework in future work.

7.2.3 Outliers and Reasonable Doubt for Correlation

Although the tendencies and correlation found is clear when comparing rhyme complexity with Metacritic score, user score or popularity, there is a considerable dispersion in the data. This is clearly visualized in the scatter plots displayed in Figures 5.2, 5.7 and 5.9. What this ultimately means is that there is reason to believe that although findings indicate positive correlation between rhyme complexity, and Metacritic score and user score, this doe not mean that songs with higher rhyme complexity are necessarily going to be more highly regarded.

7.2.4 Validity of Rhyme Complexity as a Measure of Quality for Rhymes in Lyrics

The results from rhyme complexity analysis yields a clear, although maybe inconclusive, answer to **RQ 1**, addressing whether or not there are any discernible correlation between rhyme complexity of rap lyrics, and popularity, Metacritic score and user score. However, as discussed over the previous subsections there are many points of concern that makes this correlation dubious at best. All this is to say that even though the rhyme complexity framework does to some degree determine the quality of the specific rhymes chosen for this thesis, the overall result may not entirely represent the full and complete picture of the complexity in rhymes for each song. What is nevertheless clear is that given the limitations of the rhyme complexity framework it still provides a comparison of rhyme complexity within the given scope.

A more nuanced discussion of how the calculated rhyme complexity relates to human evaluation of rhyme complexity will be presented in Section 7.5.1 regarding the implications when comparing Experiment 1 and Experiment 2.

7.3 Lyrics Generation and Evaluation of Generated Phrases

The responses of the evaluation survey for the generated hip hop lyrics phrases alongside existing phrases as presented in Section 6.3 indicate that all categories of lyrics; *critically acclaimed*, *critically despised*, *popular* and *unpopular* score higher on average than any of the generated phrases. There are, however, a few key takeaways that warrants further discussion and may challenge the validity of theses results. A multitude of these points of interest will be presented and discussed over the following this and the following section.

7.3.1 Non-Audible Presentation of Phrases

As stated in the introduction in Chapter 1, the scope of this thesis is confined to an in-depth look at textual lyrics only, disregarding the audible aspects of rap outright. While the research conducted is sufficient to see clear patterns in the evaluation of AI generated phrases. in relation to existing, human made phrases, it is reasonable to believe that if the phrases were presented in audible format as well as textual, the results of the evaluation might be substantially different. The presentation of hip hop theory in Chapter 2, indicate that critical acclaim and popularity is inherently intertwined with the audible delivery (flow and rhythm), as well as musical creativity, ease of listening, subject matter and many other factors that may impact people's perception and consumption of the songs.

The critical score and popularity metrics for the lyrics dataset are based on album score. This means that although the phrases that are stated as critically despised rap lyrics are hailed from songs of critically despised albums, this does not necessarily mean that the lyrics of the specific song, and even specific phrase used in the survey, would be despised by critics. In fact, the findings of this survey indicate that people evaluate the lyrics chosen to represent critically despised songs higher than any of the other categories in every metric. This may indicate that lyrics is not as grand of a concern for critics as they are for general consumers. This would, however in some ways contradict the rhyme complexity analysis conducted over the lyrics analysis in experiment one, which indicate that rhyme complexity is correlated with critics score. It may be more pertinent to assume this discrepancy is due to the specific phrases used, the participant's preference and the focus on lyrics alone, rather than some underlying difference in manner of thought between general consumers and critics. This will also be discussed in more detail in Section 7.5.1.

7.3.2 Optimization of Lyrics Generation System

It is reasonable to believe that there are qualities of the current output of the generative system that makes it fairly obvious to objective observers that the generated phrases are in fact computer generated, which may in turn influence their opinion about the quality of the phrases. Stating definitively whether this is mainly due to grammar, cohesion or rhyme complexity, is nevertheless not a trivial task, seeing as each of these metrics

consistently perform lower for generated phrases than existing phrases from both sides of the popularity and critical spectrum.

Spending more time and resources on hyperparameter optimization could substantially improve the quality of output of the generative system. Simply fine-tuning the dataset used for training might also be beneficial, *e.g.* removing non-words that might exist in the lyrics catalog or implementing a system for removing culturally insensitive words that one would not like to have included in the eventual output. The improvement of the module for generating lyrical phrases will accordingly be suggested as a natural basis for future work in Section 8.3.

7.3.3 Selection and Standardization of Phrases

The selection phase was executed manually, as implementing a system that would take into account grammatical quality, rhyme scheme and general cohesion was deemed to be outside the scope of this thesis. As the lyrics of existing songs are merely an interpretation of the audible medium, often executed by AI or independent users of web pages like *Genius*[1] or *Musixmatch*[2], the eventual phrases used in the evaluation survey were standardized.

Over the generation phase, all words were made lower case and punctuation was stripped, except for *hyphen "-"* and *apostrophe "'"*, which were deemed a substantial part of the textual output, not just for phrasal punctuation and flow, but for expressing individual words in individual phrases. The operations of lower casing words and stripping punctuation was done to limit the vocabulary for the training of the AI model, and ultimately reduced the size of of the initial vocabulary by half, from ∼164'000 words to ∼77'000 words. The same goes for the output of the system, *i.e.* the generated hip hop phrases, were created using this smaller vocabulary.

After initial generation of the phrases, the best phrases were selected manually taking into account cohesion rhyme scheme and subject matter. To standardize the generated phrases and better represent the content of the phrases, the lines were sometimes divided manually to better represent the lyrics of existing songs. Each line was capitalized to start with an upper case letter, and punctuation was added in the forms of commas (,) when this was deemed to help the flow of the lyrics. No words were removed, and the order of the words remained unaltered. As the lyrics from existing phrases was also standardized after the same merits, it is reasonable to assume that the integrity of the evaluation is still maintained after standardisation of the textual output from the generative system.

The selection of phrases could in the future be executed through a connected software system. As there appear to be a weak, but tangible correlation between the perceived rhyme complexity through the survey and the calculated rhyme complexity through the rhyme complexity framework, as well as a clear correlation between rhyme complexity and general quality for the survey participants, a system that scores each of the generated phrases and picks the phrases with higher rhyme complexity scores may yield a greater

[1] www.genius.com
[2] www.musixmatch.com

chance of selecting the generated phrases that will be perceived well.

7.3.4 Rhyme Complexity of Generated Phrases

The generated phrases are generally fairly short, and as has previously been discussed in Section 7.2, the rhyme complexity framework disproportionately and consistently reward longer phrases, thus comparing the calculated rhyme complexity of generated phrases with the rhyme complexity of entire songs offer a clear disadvantage. This discrepancy in rhyme complexity for existing lyrics and generated phrases becomes somewhat mitigated when comparing the phrases to just the existing phrases used in the survey. Although ultimately, the average rhyme complexity score of the generated phrases was lower than any of the other categories of phrases, it did in some instances outperform the rhyme metrics of the existing phrases. In some instances, the values of rhyme metrics in generated phrases even performed higher than the averages of other categories of entire song lyrics. This indicates that there is a clear potential for achieving higher rhyme scores for generated phrases.

7.4 Survey and Findings from Human Evaluation

This section is dedicated to an in-depth discussion about the participation in the valuation survey, evaluation of the survey itself and the results, as well as the implications and validity of the findings. Firstly there will be a presentation of the survey, followed by some general tendencies found in the responses from the survey, and then a presentation and discussion of possible challenges and reasons for doubt as to the validity of the results and findings.

7.4.1 Setup and Contents of Survey

The survey was designed to evaluate generated phrases of lyrics alongside lyrics from existing hip hop songs on five metrics; *general quality*, *rhyme complexity*, *cohesion / meaningfulness*, *grammar* and perception of the lyrics as *AI / human generated* on a scale from 1 to 5. This characterization was deemed adequate to determine whether there are any aspects that contribute more to the overall quality of the phrases. This did, however, not turn out to yield any significant insight, as there are strong correlation between general quality and all the other metrics. This can be seen very clearly in Figure **??** with a very strong correlation between general quality and the average of all metrics evaluated.

The existence of this correlation suggests that the general quality metric is somehow influencing the participants' evaluation of the other metrics for each phrase, or that the general quality metric already encapsulates the essence of all of the other metrics. Although the general quality metric was intended as a first impression evaluation, and accordingly described as such in the introduction for the survey, this might have yielded adverse consequences to the outcome of the responses.

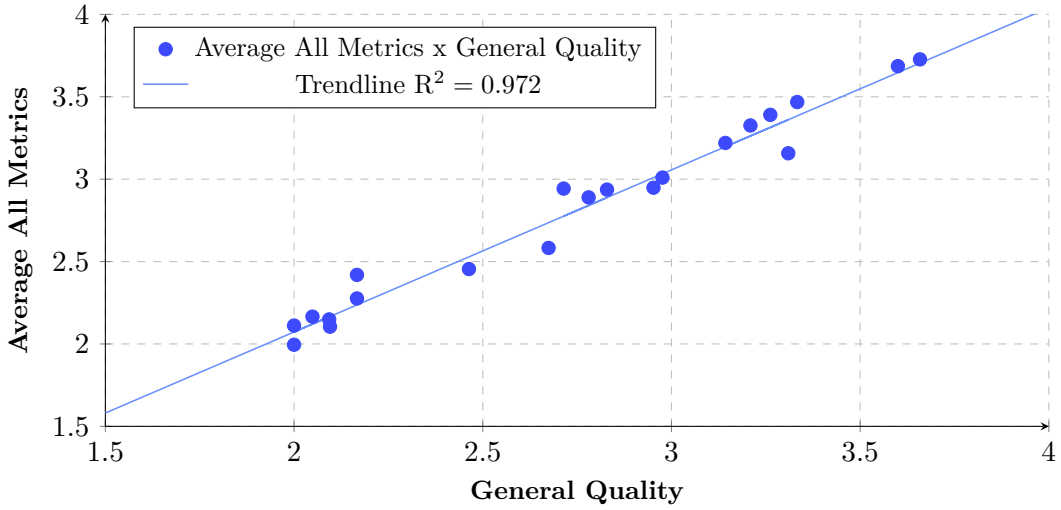


Figure 7.1: Correlation between general quality and and average score for all metrics evaluated.

As for the selection of phrases, the inherent weakness in providing a small quantitative selection of phrases will undoubtedly affect the score in some manor. While in this instance, the critically despised songs yielded significantly higher scores on every survey metric, a slightly altered selection of phrases may yield a different result.

As all the generated phrases were generated using the same start seed of *"ain't seed"*, they do in some sense share subject matter. Introducing some variation to the start seed might yield a more diverse set of phrases, which might in turn make it even harder to distinguish which phrases are generated by AI. However, the nine generated phrases consist of 279 words in total, where 177 (∼63%) are unique words that are only used once, which suggest a fairly large variation in the subject matter of the generated phrases. The start seed used was so done to utilize some of the most popular words in the same lyrics catalog as the existing phrases were chosen from, so it is reasonable to believe that the generated phrases share a certain degree of subject matter with the existing phrases used as well.

There were three phrases representing each of the four categories; critically acclaimed, critically despised, popular and unpopular songs, while there were nine phrases generated by the system. While this is intended to form a better basis for people's perception of the generated lyrics, this may offer an unfair advantage to existing lyrics. However, this may work both ways.

7.4.2 General Tendencies of Responses to Survey

The overall scores for the generated phrases paint a clear picture. None of the phrases scored higher than any of the existing phrases in any of the categories, with the exception of perception of AI / human generation, which will be further discussed in Section

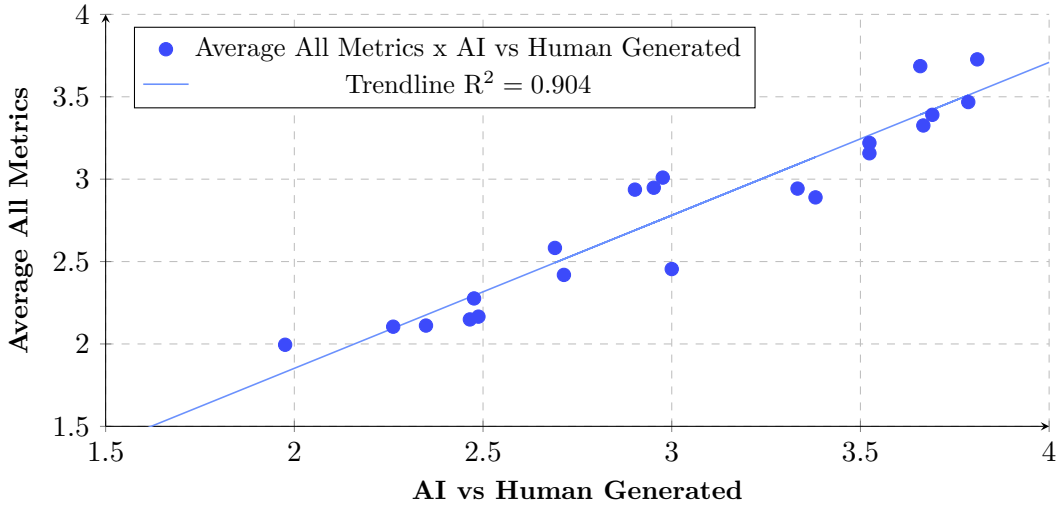


Figure 7.2: Comparing the survey responses of average of all metrics with perception of lyrics as AI generated.

7.4.4. The reasons for separating the evaluation of phrases into five different metrics was among other to determine whether for instance perception of poor grammar would be a significant signifier that lyrics was machine generated, and how this would in turn relate to overall quality and average score. As it turns out, while that might still be the case, the most significant correlation between perception of lyrics being computer generated was cohesion / meaningfulness. This is illustrated in Appendix H, in Figure 1 which displays a strong correlation between general quality and cohesion, with a low degree of dispersion between the observations.

7.4.3 Participation in Survey

Given that the survey was distributed through a network of fellow students, family and acquaintances, there may not be as large of a dispersion as would be preferable. However, there does appear to be a fair dispersion in age group and relationship to hip hop, assuring some variation in background and relationship to the subject matter. Having a quantitative research method with the collection of responses mostly connected to the academic network, many of the responses are from individuals within the computer science and similar relationship with rap and artificial intelligence. The distribution of age for the responses are however fairly well dispersed. There were however, only three participant with a self reported 5 out of 5 relationship with hip hop, and only six people with a reported 5 out of 5 knowledge base for AI, so the upper extremities of both these segments of participants.

7.4.4 Ramifications of Perceived AI Generated Lyrics

The fact that there is a strong correlation between the perception of lyrics as human made to the overall average score for the lyrics. As is clearly visualized in Figure 7.2, the more convincingly human generated a phrase is, the more likely it is to receive a higher average score for all metrics evaluated in the survey. This is in line with the findings of Simon Colton's findings regarding the perception of creativity and computers [Colton, 2008, 2012, Colton and Wiggins, 2012].

This also emphasizes the significance of the instance of "I'm smokin' genius", the computer generated phrase was more convincingly human made than some of the existing human written phrases. While this phrase was more convincingly human made than three of the existing phrases, it was still evaluated below all of them in every metric, and even lower than one other of the generated phrases. This simply highlights the implication that although the general tendencies show a strong correlation in one direction, single instances might still not abide by the norm.

7.4.5 Dispersion in Responses

Antecedently eluded to in Section 6.3.4 and further visualized in Appendix F, there is a large dispersion in responses for every phrase in the survey. Only one of the existing phrases did not get scored on every single value for every metric in not at least one of the responses. In other words, for every phrase, at least one of the participants deemed each metric as both bottom of the barrel and top of the heap.

Regarding the generated phrases, there is a large dispersion as well, albeit overall more concentrated on the 1 to 4 scale. The evaluation of any metric as top of the heap (5 out of 5), is more rare among the generated phrases and in more than a handful of instances, the phrases do not have a single 5 response in one of the metrics. However, all generated phrases display at least one 5 evaluation in the AI / Human Generated metric, which is a good indication that system generation of lyrics is on the cusp of creating convincingly human generated lyrics. This essentially means that even though the average score indicates that generated phrases are computer generated, all of the phrases are convincingly human generated to at least one of the participants in the survey.

Being a subjective survey, this dispersion is not all that surprising, it does however mean that even though the average values for all responses show a decisive pattern, this may not be the case given a different set of participants. For the instance, as presented in Figure 6.5, participants with lower self-perceived relationship to hip hop and knowledge about AI generally evaluate the generated phrases higher in every metric.

7.4.6 Survey Score for Different Categories of Phrases

In Appendix G, there can be found an overview of the average score for the different categories of phrases; *critically acclaimed*, *critically despised*, *popular*, *unpopular* and *generated* phrases. One exiting implications of the score of these different categories

Category:	GQ:	RC:	G:	M:	AI/H:	AAM:
Critically Despised Phrases	3,51	3,48	3,65	3,65	3,72	3,60
Popular Phrases	3,17	2,95	3,26	3,39	3,47	3,25
Critically Acclaimed Phrases	3,14	2,68	3,30	3,18	3,34	3,13
Unpopular Phrases	2,77	2,84	3,05	2,74	3,21	2,92
Average All Phrases	**2,74**	**2,62**	**2,89**	**2,76**	**3,03**	**2,81**
Generated Phrases	2,19	2,12	2,33	2,13	2,49	2,25

Table 7.3: Average score of each metric for all phrases representing each of the categories, along with the average value for all phrases in the survey.

is that the phrases hailed from critically despised songs are constantly scored higher than all the other categories on all the metrics. This also includes the lyrics critically acclaimed songs, indicating that the perception of the participants in the survey are in direct contrast when juxtaposed to the perception of critics. The same goes for the rhyme complexity for critically despised songs in contrast critically acclaimed songs. They exhibit the human perceived rhyme complexity of 3,48 and 2,68 respectively, implying that the lyrics of critically despised songs are perceived as much more complex in rhyme structure than that of critically acclaimed songs. This is in direct contrast with the findings in Experiment 1, and will accordingly be further discussed in Section 7.5.

Further, it can be observed that when calculating the average score of the metrics for all the phrases, all of the categories display a value greater than the average, with the notable exception of generated phrase. This further solidifies the generated phrases as being less favorably evaluated than any of the other categories of phrases. It warrants to be pointed out, as previously touched upon in Section 7.3, that even though the phrases are hailed from songs from critically despised albums, this might not mean that the lyrics is representative of the critics evaluation of the album. Additionally, there were only selected three phrases from each of the four categories of existing lyrics and 43 responses in total, which means that any alterations or additions to the selection of existing phrases may significantly alter the outcome of the evaluation.

7.4.7 Validity of Results

The correlation seen between general quality of the phrases and each of the other metrics may indicate that each of the metrics are fairly intertwined with the general perception of the phrases. However, this could also simply implicate that judging phrases on the merits of first impression / general quality is sufficient to deem the lyrics as being a certain degree of good, or more precisely stated that it scores highly on the metrics designed to signify overall quality of lyrics for this thesis.

As seen in Figure 7.3, the fact that age group is closely related to general relationship and knowledge about rap or AI might yield some interesting input to the findings of the survey. As this might indicate that the people representing the less familiar with

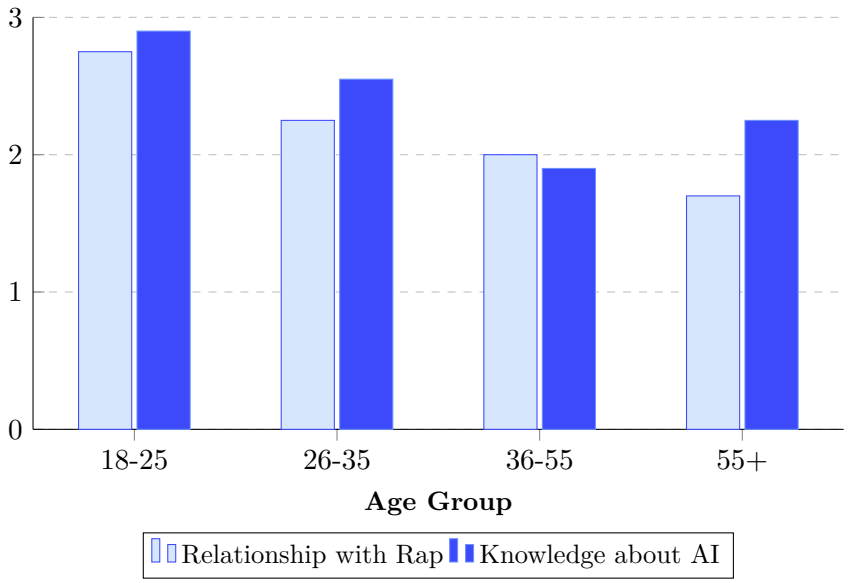


Figure 7.3: Relationship between age group of participants and their self perceived knowledge about rap music and artificial intelligence.

rap and AI might also be of an older age group and less familiar with English language. The omission of any acknowledgement of knowledge about the English language for the participants is regrettable, and may adversely affect the results of the survey, particularly given the fact that the survey was distributed primarily to Norwegian native speakers and that all lyrics was presented in English.

Over the presentation of the results, there were presented many different interpretations and visualisations of the responses for the survey. Given the points discussed above there are many aspects to take in to consideration when finally drawing a conclusion, however, all representations of the results from the experiment points in a particular direction. Although the average scores of the survey yielded a decisive trend, in the light of the points presented over this discussion there might still warrant the same conclusion. This will be further elaborated upon in Chapter 8.

7.5 Inference of Results for Both Experiments

There are some areas of interest when comparing the results of Experiment 1 and Experiment 2. Particularly on the topic of rhyme complexity there are some implications to examine in the cross-section of results from system calculation in lyrics analysis to human evaluation of existing lyrics. Lastly, we will discuss the opportunity and potential of utilizing results from lyrics analysis to generate better lyrics using AI.

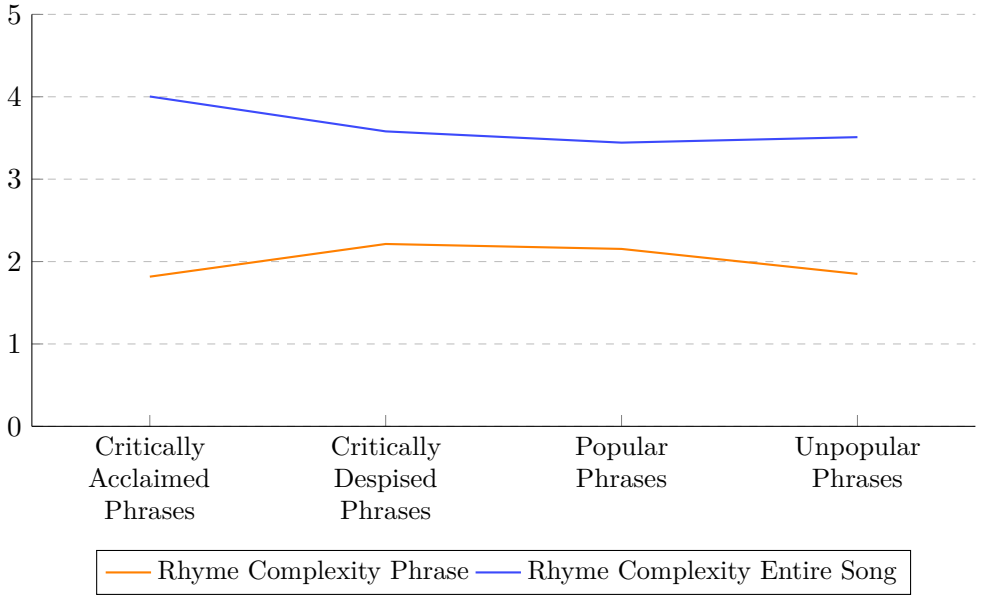


Figure 7.4: System calculated rhyme complexity for phrases used in the survey in each respective category, in relation to the rhyme complexity of the entire song.

7.5.1 Contradiction in Rhyme Complexity

In Experiment 1 the results showed a very decisive correlation between Metacritic score of songs and the system calculated rhyme complexity, implying that more highly regarded songs are regarded more favorably. The same correlation can be seen in the human evaluation in Experiment 2, indicating that both these distinct group of evaluators regard rhyme complexity as at least a part of the quality of reap lyrics. On the other hand, there appears to be a contradiction in the perception of rhyme complexity altogether. It is essential to point out here that when analyzing rhyme complexity in Experiment 1, the system calculates the rhyme complexity for entire songs, while in Experiment 2 these same songs are represented by short phrases. This distinction turns out to be of importance for this contradiction.

Examining the correlation between rhyme complexity for the existing phrases used in the survey and the rhyme complexity of the entire songs they are extracted from, does in fact paint a very different picture of the perception of rhyme complexity in relation to critical acclaim. In Figure 7.4 the calculated rhyme complexity for all the phrases in each category are seen in relation to the calculated rhyme complexity for the entire songs they are hailed from. This shows that the rhyme complexity for the individual phrases chosen for the survey increases as the complexity for the entire song decreases, when the average is calculated by category. This indicates that even though the entire lyrics for critically acclaimed songs are higher in rhyme complexity than that of critically despised songs, as shown in Experiment 1, the opposite is the case for the short phrases used in

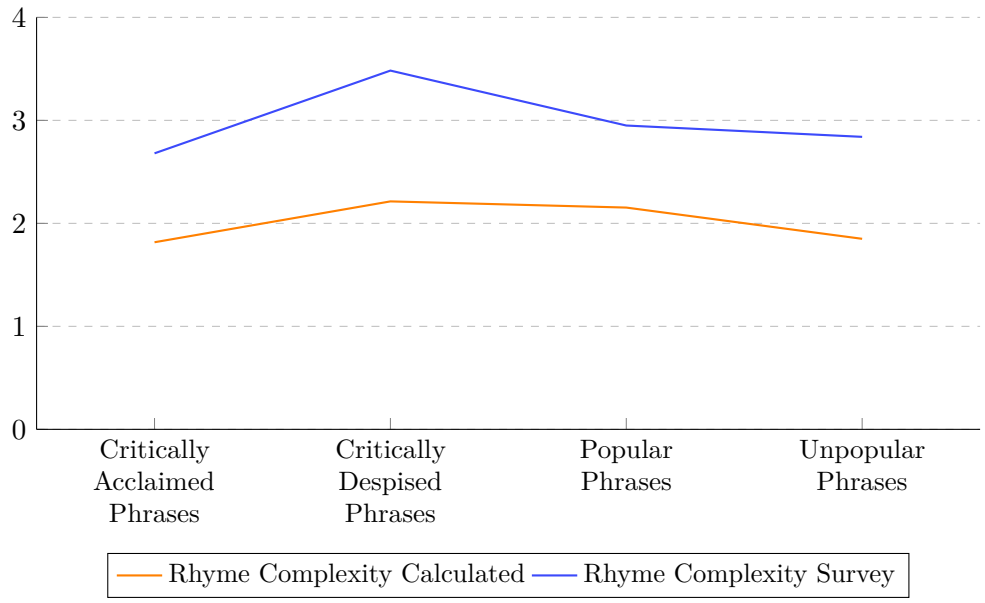


Figure 7.5: System calculated rhyme complexity for phrases used in the survey in each respective category, in relation to perceived rhyme complexity from responses to the survey.

Experiment 2. This is in line with the responses to the survey, as seen in Figure 7.5, where the calculated rhyme complexity for the phrases in each category increases and decreases along with the perceived rhyme complexity for the same categories from the survey responses.

This latest finding regarding the correlation between calculated, and perceived rhyme complexity offer a satisfying reinforcement to the validity of the rhyme complexity framework as a measure of complexity for rhymes in rap lyrics. Furthermore, the correlation between rhyme complexity measure and the general quality of phrases implies that the framework could also be used as a measure of objective quality. This will be stated in more definite terms in Chapter 8.

7.5.2 Lyrics Analysis to Generate Better Lyrics

As the lyrics analysis in Experiment 1 was conducted before the lyrics generation, the opportunity to have this influence lyrics generation in Experiment 2 did not present it self over the course of this thesis. There has however been some areas of affirmations of the stated assumptions about rhyme complexity as a aspect of quality for lyrics, based on the combination of these two experiment. Particularly based on the latest discussion regarding coinciding views of rhyme complexity for the framework created for this thesis and the perceived rhyme complexity through human evaluation implies that implementing measures to bolster rhyme complexity during generation may increase overall quality of

the phrases.

8 Conclusion and Future Work

This final chapter constitutes a conclusion to the thesis in light of the the research conducted and results presented in Chapters 5 and 6, particularly as it relates to the discussion and evaluation in Chapter 7. To conclude the thesis properly, the results and key takeaways from the discussion will be directly tied the *Research Questions* (**RQ**)s posed in the introduction, to see how well they have been answered, and whether their corresponding goals have been achieved.

Over the course of this thesis research has been conducted into the field of lyrics analysis, culminating in a framework for the assessment of overall rhyme complexity for song lyrics. While the experiments conducted with the rhyme complexity framework for the thesis expressly evaluated lyrics from rap songs, the framework itself is not designed with this in mind, meaning that it is just as qualified for analyzing lyrics from any other genre of music. This framework revealed a decisive correlation between complexity of rhymes in rap lyrics and critic score as well as user score, and an inverse correlation between rhyme complexity and popularity. The potential for this framework is evident, and is regarded as the main contribution to the field of lyrics analysis and natural language processing, attempting to breach the gap of understanding the intersection between linguistics and computer technology.

Furthermore an experiment was conducted using *Artificial Intelligence* (AI) to generate lyrics that were evaluated in relation to existing lyrics, both by quantitative human evaluation and by the rhyme complexity framework designed for, and used in the lyrics analysis. The generated lyrics ended up yielding nine phrases, where one of them was deemed to be indistinguishable from human generated lyrics, displaying the clear potential of emulating human creativity using computers. Despite this, all phrases were evaluated to be less favorable than existing lyrics in every metric measured, albeit displaying the occasional glimmer of potential for production of great phrases, and plentiful room for improvement. The merits of this conclusion will be further explored over the following sections.

8.1 Conclusion to Lyrics Analysis and Rhyme Complexity

The objective of the lyrics analysis was stated clearly in *Goal 1* in Chapter 1, which says to "analyze rap lyrics to discern what separates lyrics of popular and critically acclaimed songs from unpopular and critically despised songs". In this section we will discern how well this has been achieved by answering **RQ 1** and its sub-questions. These **RQs** will be reiterated here, for ease of accessibility:

RQ 1 *Is it possible to determine what separates lyrics of popular and critically acclaimed rap songs from lyrics of unpopular and critically despised songs?*

RQ 1.1 *Is it possible to utilize statistics to identify patterns that are used in the lyrics of rap songs with different degrees of popularity?*

RQ 1.2 *Is it possible to utilize statistics to identify patterns that are used in the lyrics of rap songs with different degrees of critical acclaim?*

8.1.1 RQ 1.1 - Hip Hop Lyrics and Popularity

The overarching question of whether it is possible to identify patterns that are used in rap lyrics of songs with different degree of popularity has been successfully answered. That is, at least to the extent of rhyme complexity. There was detected a weak negative correlation within given parameters for a specific domain, namely the relationship between rhyme complexity and the popularity for the specific catalog of lyrics that was used, indicating that more popular music tend to be less complex in rhyme structure. It nevertheless warrants to be pointed out, in light of the points discussed in Chapter 7, that any alteration in the dataset used during analysis may yield different results and display a different correlation.

8.1.2 RQ 1.2 - Hip Hop Lyrics and Critical Acclaim

Much in the same vein as with popularity, the initial answer has been answered successfully. It is possible to detect patterns in lyrics with a varying degree of critical acclaim, with an even higher degree of certainty. However the same limitations in domain specificity and definition of metrics used, make it reasonable to believe that the results do not display the full picture.

8.1.3 Goal Achievement for Lyrics Analysis

Given that the two sub-questions regarding lyrics analysis was answered affirmatively, the overarching goal of the lyrics analysis for this thesis could be deemed successfully achieved. The object of analyzing lyrics to see what separates lyrics from different ends of the critical and popular spectrum yielded some interesting insight into the preferences of critics and consumers alike, although with a firm caveat that this is solely based on textual interpretation of rap music. Furthermore, the clear correlation between system calculated rhyme complexity and human perceived rhyme complexity lend credence to the framework as a valid mode of assessing rhyme complexity.

8.2 Conclusion to Lyrics Generation

The overarching goal or lyrics generation was to be able to generate rap phrases that are better than lyrics from existing popular and critically acclaimed songs. To determine whether or not this was achieved, each of the underlying sub-questions **RQ 2.1**, **RQ**

2.2 and **RQ 2.3** of **RQ 2** will be addressed categorically. This will in turn inform the overall goal achievement.

RQ 2 *Is it possible to generate rap phrases using AI that is better than lyrics from existing popular and critically acclaimed rap songs?*

RQ 2.1 *Will the generated lyrics score highly on metrics defined for evaluating rap lyrics, including findings during rap lyrics analysis (**RQ 1**)?*

RQ 2.2 *Will the generated lyrics be perceived as better than lyrics from existing popular and critically acclaimed rap songs in human evaluation?*

RQ 2.3 *Will the generated lyrics be indistinguishable from human generated rap lyrics?*

8.2.1 RQ 2.1 - Generated Lyrics and Rhyme Complexity

As presented in Section 6.3.1 on rhyme complexity of the generated lyrics, the generated phrases consistently score below the averages of all other categories of existing lyrics, most notably for the rhyme complexity metric. This indicates that generally the rhyme complexity of generated lyrics are lower than that of existing lyrics, however, in all but one rhyme metric, the generated phrases score comparable to or higher than one or more of the other categories of existing phrases.

In the case of the *highest word rhymes* metric, the generated phrases score higher than critically acclaimed phrases, and the highest value of the rhyme metrics for any of the generated phrases, even perform above the averages of all songs. This illustrates a clear potential for generating phrases that perform better than critically acclaimed songs with regards to rhyme complexity. However, the fact that the average rhyme complexity of generated phrases is lower than any other of the categories of lyrics, makes the affirmative conclusion to this **RQ** dubious at best.

8.2.2 RQ 2.2 - Perception of Generated Lyrics and Existing Lyrics

The second sub-question, concerns human perception of the quality of generated phrases alongside existing phrases. When evaluated by the four metrics; general quality, rhyme complexity, grammar and cohesion/meaningfulness, and aggregated scores, non of the generated phrases scored higher than any of the existing phrases in any metric. This may be partly due to small errors in the lyrics that makes it somewhat apparent which phrases are generated and which are written by humans. The fact that there is a clear correlation between whether a phrase is perceived as AI generated and the general quality of the phrase, bolsters the notion previously stated in Section 2.5, that the perception of art being computer generated affects the consumers enjoyment and assessment of the art itself.

8.2.3 RQ 2.3 - AI Generated Lyrics vs. Human Generated Lyrics

This **RQ**, on whether or not the generated phrases would be indistinguishable from existing lyrics, yielded the most significant result. One of the generated phrases was deemed more convincingly created by a human than three of the existing phrases. Of these phrases, one was from a critically acclaimed song, one from a popular song, and one from an unpopular one. Although this was the case for only one of the generated phrases, it displays clearly the potential of generating lyrics that is indistinguishable from human generated lyrics. Rap lyrics does not necessarily follow conventional rules for lyrical writing as stated in Chapter 2.1, and as such might be harder to distinguish AI generated rap phrases from human made rap phrases than other lyrical genres that are generally more beholden to set conventions.

8.2.4 Goal Achievement in Lyrics Generating

Of the three sub-questions of **RQ 2**, one of them was answered affirmatively, one was answered negatively, and the third one dubiously positive. Given this, it would be hard to argue that the overarching goal of generating rap lyrics that is better than existing popular or critically acclaimed songs have been achieved. There is however reason for optimism regarding the achievement of the overarching goal, as there are instances where the generated lyrics does perform well, both with regards to calculated rhyme complexity and by human evaluated metrics.

8.2.5 Conclusion to Lyrics Generation

The state of the lyrics generation for this thesis yields phrases that, while generally assessed in lower regard than existing lyrics, occasionally display individual aspects of generated phrases that outperform the ones existing phrases. This truly highlights the prowess of generating great rap lyrics by the use of AI. On the intersection between lyrics analysis and lyrics generation, the findings on rhyme complexity and correlation between the human perceived rhyme complexity indicates that the rhyme complexity framework is a valid approach to assessing rhyme complexity in lyrics. This in addition to the fact that there is a clear correlation between perceived rhyme complexity and general quality from human evaluation further indicates that generating lyrics with a greater fixation on rhyme complexity might yield phrases that are evaluated in higher regard.

8.3 Future Work

Finally, this section will highlight some areas that were deemed to be of interest to explore further, as well as some clear areas of improvement for the systems presented in Experiment 1 and Experiment 2. This concerns both the lyrics analysis and the lyrics generation, and would ultimately bolster the possibility of generating the perfect rap lyrics phrase.

8.3.1 Expanding on the Rhyme Complexity Framework

There is no clear path to expanding upon the rhyme complexity framework, however, there are lots of ways to go. As stated in the discussion, in Section 7.1, the framework is currently relative to the specific domain of the dataset used. Making a universal framework that is domain independent would provide the opportunity of comparing rhyme complexity between datasets, for instance different musical genres. In addition, there are multiple metrics still not accounted for the would make for a more complete evaluation of the rhyme complexity. The other clear enhancement for the analysis itself would be to perfect the task of word-to-phoneme conversion. As it stands, the limitation of this conversion clearly inhibits complete detection of rhyme structure and may highly impact the quality and validity of the results.

8.3.2 Optimize Lyrics Generation

While the output used for the evaluation of generated text shows that in some instances, people perceived lyrics to be more convincingly human made than some of the existing phrases, there is still plenty left to be desired when generating phrases with better grasp on grammar, cohesion and rhyme structure. The generated phrases that were evaluated were a manually selected set of phrases from thousands of phrases, where a majority of them were deemed unsatisfactory with regard to grammar, cohesion or rhyme structure. Spending more time and resources fine tuning the parameters used for the generation of phrases may bring us closer to achieving the goal of generating the perfect rap phrase. A natural step forward could be to utilize a larger set of training data, or simply make small alterations to the data used, to bolster chance of satisfactory output.

8.3.3 Implementation of Automatic Phrase Selection

For the scope of this thesis, the eventual selection of generated rap phrases was done manually. Ideally, an automated system would be implemented that selected phrases from the vast catalog of generated phrases. This automated selection module could for instance highlight specific features like the ones discovered through lyrics analysis, and favorably chose phrases that score highly on rhyme complexity. By running all generated phrases through the rhyme complexity framework presented, or even implement different filters that reward cohesion and proper grammar could yield instant results of satisfactory phrases.

Bibliography

ACC. The definition of computational creativity, 2020. URL `https://computationalcreativity.net/home/about/computational-creativity`. [Online; accessed 20. Mar. 2021].

Kyle Adams. On the Metrical Techniques of Flow in Rap Music. *Music Theory Online*, 15(5), Oct 2009. doi: 10.30535/MTO.15.5.1.

Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. ACM press New York, 1999. ISBN 13:9780201398298.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1171–1179, Cambridge, MA, USA, 2015. MIT Press. doi: 10.5555/2969239.2969370.

Margaret A. Boden. Creativity and Artificial Intelligence. *Artificial Intelligence*, 103(1): 347 – 356, 1998. ISSN 0004-3702. doi: 10.1016/S0004-3702(98)00055-1.

Margaret A. Boden. *The Creative Mind: Myths and Mechanisms*. Routledge, 2004. ISBN 9780415314527.

Adam Bradley. *Book of Rhymes: The Poetics of Hip Hop*. Basic Civitas Books, 2009. ISBN 0-465-00347-8.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems 33 pre-proceedings (NeurIPS 2020)*, 2020. arXiv: 2005.14165 [cs.CL].

John A. Bullinaria. Neural network learning from ambiguous training data. *Connection Science*, 7(2):99–122, 1995. doi: 10.1080/09540099550039309.

Simon Colton. Creativity Versus the Perception of Creativity in Computational Systems. In *AAAI Spring Symposium: Creative Intelligent Systems*, pages 14–20, 2008.

Simon Colton. *The Painting Fool: Stories from Building an Automated Painter*, pages 3–38. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-31727-9. doi: 10.1007/978-3-642-31727-9_1.

Simon Colton and Geraint A. Wiggins. Computational creativity: The final frontier? In *Proceedings of the 20th European Conference on Artificial Intelligence*, ECAI'12, page 21–26, NLD, 2012. IOS Press. ISBN 9781614990970.

Robert Dale, H. L. Somers, and Hermann Moisl. *Handbook of Natural Language Processing*. Marcel Dekker, Inc., USA, 2000. ISBN 0824790006.

Kees van Deemter, Mariët Theune, and Emiel Krahmer. Real versus Template-Based Natural Language Generation: A False Opposition? *Computational Linguistics*, 31(1): 15–24, 03 2005. ISSN 0891-2017. doi: 10.1162/0891201053630291.

Emily Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In *Advances in Neural Information Processing Systems*, volume 28, pages 1486–1494. Curran Associates, Inc., 2015. URL `https://proceedings.neurips.cc/paper/2015/file/aa169b49b583a2b5af89203c2b78c67c-Paper.pdf`.

Minal Dhankar and Nipun Walia. An introduction to artificial intelligence. In *Emerging Trends in Big Data, IoT and Cyber Security*, pages 105–108, 2020. ISBN 978-93-86238-93-1. URL `https://msi-ggsip.org/wp-content/uploads/conference2020.pdf#page=118`.

Paul Edwards. *How to Rap: The Art & Science of the Hip-Hop MC*. Chicago Review Press Incorporated, 2009. ISBN 9781569763773.

Paul Edwards. *How to Rap 2: Advanced Flow and Delivery Techniques*. Chicago Review Press Incorporated, 2013. ISBN 9781613744024.

Fernand Gobet and Giovanni Sala. How artificial intelligence can help us understand human creativity. *Frontiers in Psychology*, 10:1401, 2019. ISSN 1664-1078. doi: 10.3389/fpsyg.2019.01401.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680. Curran Associates, Inc., 2014.

Alex Graves. Generating sequences with recurrent neural networks. arXiv: 1308.0850 [cs.NE], 2014.

Hussein Hirjee and Daniel G. Brown. Automatic detection of internal and imperfect rhymes in rap lyrics. In *In Proc. 10th International Society for Music Information Retrieval Conference (ISMIR) 2009*, pages 711–716, 2009. doi: 10.1.1.205.8962.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735.

Astrid Inge Holtman. *A Generative Theory of Rhyme: an Optimaily Approach*. PhD thesis, Utrecht: Research Institute of Linguistics, 1996. ISBN: 9789054340614.

Adam Krims. *Rap Music and the Poetics of Identity*. Cambridge University Press, 2000. ISBN 0521634474.

Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, and Yong Yu. Neural Text Generation: Past, Present and Beyond. 2018. arXiv: 1803.07133 [cs.CL].

Jose P. G. Mahedero, Álvaro MartÍnez, Pedro Cano, Markus Koppenberger, and Fabien Gouyon. Natural language processing of lyrics. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, MULTIMEDIA '05, page 475–478, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595930442. doi: 10.1145/1101149.1101255.

Eric Malmi, Pyry Takala, Hannu Toivonen, Tapani Raiko, and Aristides Gionis. DopeLearning: A Computational Approach to Rap Lyrics Generation. 2016. doi: 10.1145/2939672.2939679.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. Encode, tag, realize: High-precision text editing, 2019. arXiv: 1909.01187 [cs.CL].

Amy McKenna. Griot | African troubadour-historian, Nov 2020. URL `https://www.britannica.com/art/griot`. [Online; accessed 23. Nov. 2020].

Nielsen Music. 2017 U.S. Music Year-End Report, 2017. URL `https://www.nielsen.com/us/en/insights/report/2018/2017-music-us-year-end-report`. [Online; accessed 19. Nov. 2020].

A. Oktradiksa, C. P. Bhakti, S. J. Kurniawan, F. A. Rahman, and Ani. Utilization artificial intelligence to improve creativity skills in society 5.0. *Journal of Physics: Conference Series*, 1760:012032, jan 2021. doi: 10.1088/1742-6596/1760/1/012032.

Christopher Olah. colah's blog, 2015. URL `http://colah.github.io/posts/2015-09-NN-Types-FP/`. [Online; accessed 09. Jun. 2021].

F. Paupier. Github - fpaupier/raplyrics-scraper: Data sourcing and pre-processing for raplyrics.eu - a rap music lyrics generation project. `https://github.com/fpaupier/RapLyrics-Scraper`, 2021. [Online; accessed 09. Jun. 2021].

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 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, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1221.

Peter Potash, Alexey Romanov, and Anna Rumshisky. Evaluating Creative Language Generation: The Case of Rap Lyric Ghostwriting. In *Proceedings of the Second Workshop on Stylistic Variation*, pages 29–38, New Orleans, jun 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-1604.

Sameerchand Pudaruth, Sandiana Amourdon, and Joey Anseline. Automated Generation of Song Lyrics Using CFGs. In *2014 Seventh International Conference on Contemporary Computing (IC3)*, pages 613–616, 2014. doi: 10.1109/IC3.2014.6897243.

Shahzad Qaiser and Ramsha Ali. Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. *International Journal of Computer Applications*, 181:25–29, 2018. doi: 10.5120/ijca2018917395.

Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018. URL `https://www.bibsonomy.org/bibtex/273ced32c0d4588eb95b6986dc2c8147c/jonaskaiser`.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019. URL `https://openai.com/blog/better-language-models/`. [Online; accessed 08. Dec. 2020].

Luis Rivas. Hip-hop is Resistance Against the Inequalities in Society. *Sundial*, Dec 2020. URL `https://sundial.csun.edu/79161/opinions/hip-hop-is-resistance-against-the-inequalities-in-society`. [Online; accessed 4. Dec. 2020].

Ronald Rosenfeld. Two decades of statistical language modeling: and where do we go from here? In *IEEE 88(8)*, pages 1270–1278, 2000. doi: 10.1109/5.880083.

Asir Saeed, Suzana Ilić, and Eva Zangerle. Creative GANs for Generating Poems, Lyrics and Metaphors. *NeurIPS 2019*, 2019. arXiv: 1909.09534 [cs.CL].

Richard Savery, Lisa Zahray, and Gil Weinberg. Shimon the Rapper: A Real-Time System for Human-Robot Interactive Rap Battles. *International Conference for Computational Creativity 2020, ICCC 20*, 2020. arXiv: 2009.09234 [cs.AI].

Magnus Själander, Magnus Jahre, Gunnar Tufte, and Nico Reissmann. EPIC: An energy-efficient, high-performance GPGPU computing research infrastructure. arXiv: 1912.05848 [cs.DC], 2019.

Alan F. Smeaton. Progress in the Application of Natural Language Processing to Information Retrieval Tasks. *The Computer Journal*, 35(3):268–278, 06 1992. ISSN 0010-4620. doi: 10.1093/comjnl/35.3.268.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. arXiv: 1706.03762 [cs.CL].

Tony Veale. Creative Language Retrieval: A Robust Hybrid of Information Retrieval and Linguistic Creativity. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 278–287, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P11-1029`.

Elijah Wald. Hip Hop and Blues, Jul 2004. URL `https://elijahwald.com/hipblues.html`. [Online; accessed 24. Nov. 2020].

Elijah Wald. Talking 'Bout Your Mama: The Dozens, Snaps, and the Deep Roots of Rap, by Elijah Wald, Jan 2014. URL `https://www.elijahwald.com/dozens.html`. [Online; accessed 24. Nov. 2020].

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 2852–2858. AAAI Press, 2017.

Mennon van Zaanen and Pieter Kanters. Automatic Mood Classification Using TF*IDF Based on Lyrics. *11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, pages 75–80, 2010.

Appendix A - Term Frequency Lyrics Catalog

A list of the 111 most commonly used words in the lyrics generation dataset. This list was devised by running the dataset through a term frequency algorithm with lemmatization, stemming and removal of stop words.

#	Word	Amount
1	i'm	15063
2	like	13196
3	nigga	11753
4	get	8661
5	thi	8254
6	got	7896
7	know	7265
8	fuck	6977
9	it	6091
10	shit	5688
11	ain't	5655
12	wa	5379
13	bitch	5155
14	go	4589
15	yeah	4410
16	see	4181
17	back	4011
18	love	3779
19	one	3608
20	make	3603
21	never	3586
22	that	3527
23	time	3311
24	come	3150
25	man	3115
26	say	3012
27	caus	2933
28	want	2868
29	feel	2811
30	can't	2799
31	let	2706
32	life	2639
33	wanna	2606
34	take	2552
35	tell	2515
36	money	2382
37	look	2381

#	Word	Amount
38	need	2228
39	way	2225
40	right	2218
41	'em	2186
42	us	2165
43	keep	2152
44	still	2115
45	ya	2064
46	hi	2063
47	think	1994
48	day	1987
49	give	1980
50	put	1932
51	live	1868
52	call	1804
53	real	1790
54	tri	1763
55	ass	1730
56	whi	1727
57	oh	1702
58	god	1688
59	you'r	1677
60	could	1667
61	even	1655
62	gotta	1653
63	onli'	1638
64	die	1632
65	said	1631
66	girl	1618
67	hit	1612
68	y'all	1585
69	world	1530
70	i'll	1522
71	caus	1522
72	good	1520
73	black	1506
74	thing	1491

#	Word	Amount
75	yo	1490
76	i'm	1460
77	better	1427
78	everi'	1411
79	talk	1337
80	mind	1334
81	new	1322
82	made	1321
83	game	1295
84	around	1284
85	babi	1280
86	motherfuck	1265
87	boy	1253
88	play	1251
89	big	1242
90	gon'	1234
91	uh	1227
92	use	1222
93	littl	1208
94	run	1175
95	kill	1167
96	hard	1146
97	high	1143
98	stop	1134
99	start	1113
100	head	1111
101	people	1106
102	rap	1085
103	show	1078
104	name	1076
105	watch	1075
106	turn	1073
107	would	1065
108	hold	1061
109	realli	1057
110	i'v	1040
111	night	1039

Appendix B - Survey for Evaluation of Rap Phrases

A presentation of the survey used during evaluation of existing and generated hip hop phrases.

The Perfect Rap Lyrics [Master's Thesis]

This survey is conducted as part of a Master Thesis in Computer Science NTNU (Norwegian University of Science and Technology) in the field of Computational Creativity. The goal of the thesis is to analyze existing rap phrases (critically acclaimed/despised, popular/unpopular) and generate new rap phrases, based on the results of the analysis, utilizing machine learning. This survey contains a total of 21 rap phrases from both existing rap lyrics and phrases generated by AI through the work on this thesis. Your task will be to subjectively evaluate these phrases based on a set of defined metrics. To lessen the difference between the existing and generated phrases, all phrases have been standardized with capital letters on new lines and commas have been added for incresed readability.

The Perfect Rap Lyrics [Master's Thesis]

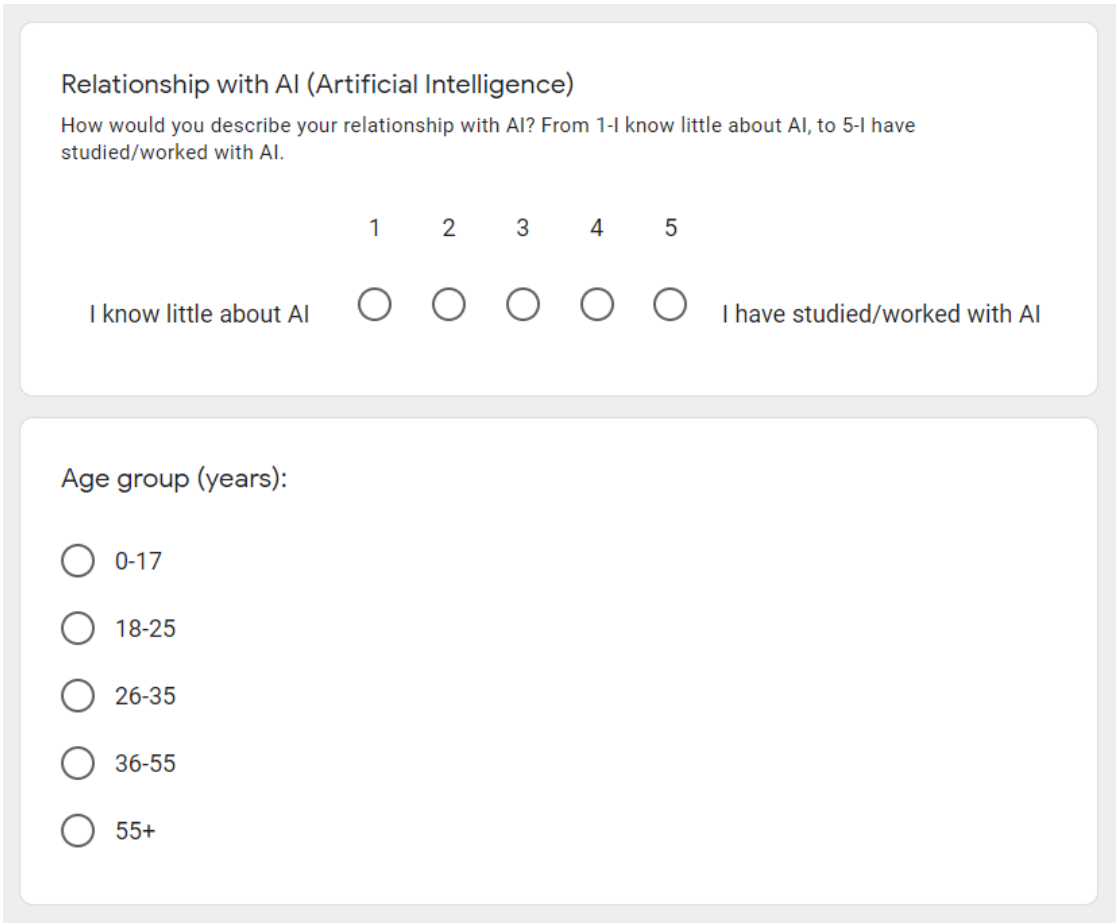
About participant

In the following section there are questions about you as a participant, this information will only be used in an aggregated function.

Relationship with rap / hip-hop:

How would you describe your relationship with rap music and rap lyrics? From 1-I almost never listen to rap lyrics, to 5-I am exceptionally interested in rap lyrics.

	1	2	3	4	5	
Almost never listen to rap	○	○	○	○	○	Exceptionally interested in rap



Relationship with AI (Artificial Intelligence)

How would you describe your relationship with AI? From 1-I know little about AI, to 5-I have studied/worked with AI.

	1	2	3	4	5	
I know little about AI	○	○	○	○	○	I have studied/worked with AI

Age group (years):

- ○ 0-17
- ○ 18-25
- ○ 26-35
- ○ 36-55
- ○ 55+

Rap phrases

In this section you will be presented with 21 rap phrases, some from critically acclaimed rap songs, some from critically despised rap songs, some from popular rap songs, some from unpopular rap songs and some generated by our system.

Every phrase will be evaluated based on 5 metrics:

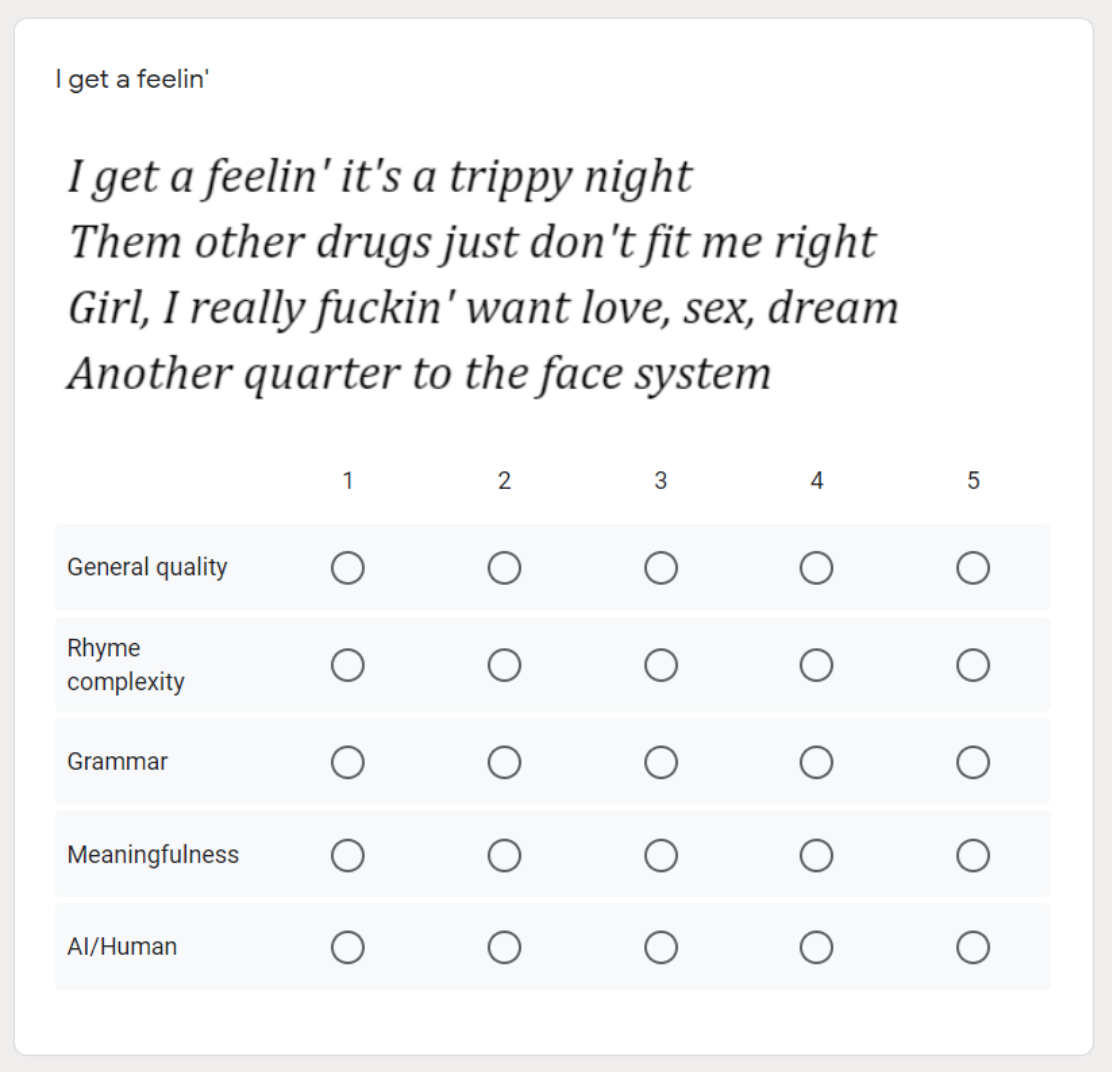
- General quality (What is your general impression of the quality of the phrase? From 1-Poor, to 5-Good.)

- Rhyme complexity (What is your impression of the rhyme complexity of the phrase? From 1-Poor, to 5-Good.)

- Grammar (What is your impression of the grammar in the phrase? From 1-Poor, to 5-Good.)

- Meaningfulness (How successful is the phrase in conveying a meaning? From 1-Not at all, to 5-Very.)

- AI generated/Man made (How certain are you that the phrase was made by a human? From 1-Confident it is generated by an AI, to 5-Confident it was made by a human.)



I get a feelin'

I get a feelin' it's a trippy night
Them other drugs just don't fit me right
Girl, I really fuckin' want love, sex, dream
Another quarter to the face system

	1	2	3	4	5
General quality	○	○	○	○	○
Rhyme complexity	○	○	○	○	○
Grammar	○	○	○	○	○
Meaningfulness	○	○	○	○	○
AI/Human	○	○	○	○	○

Appendix C - Rap Phrases Used in Survey

A complete overview of the phrases used for the evaluation of rap lyrics, placed under the category under which capacity they were used in the survey.

Critically Acclaimed Phrases

Kendrik Lemar - How Much a Dollar Cost
Guilt trippin' and feelin' resentment
I never met a transient that demanded attention
They got me frustrated, indecisive and power trippin'
Sour emotions got me lookin' at the universe different

Chance the Rapper - Same Drugs
Don't forget the happy thoughts
All you need is happy thoughts
The past tense, past bed time
Way back then when everything we read was real

Kanye - Gorgeous
And kiss the ring while they at it, do my thing while I got it
Play strings for the dramatic ending of that wack shit
Act like I ain't had a belt in two classes
I ain't got it I'm coming after whoever who has it

Critically Despised Phrases

Lil Wayne - She Will
And you could take it to the bank and deposit that
Put your two cents in and get a dollar back
Some people hang you out to dry like a towel rack
I'm all about I, give the rest of the vowels back

Mac Miller - Smile Back
I could change the world with just a single track
Underage table in the back, smokin', drinkin' yak
Pittsburgh, let me show the world just where my city at
On the map, illest city out, I guess that's just a fact

Eminem - Believe
These thoughts can be harsh and cold as ice
To me they're just ink blots
I just fling 'em like slingshots and so precise
So you might wanna think it over twice

Popular Phrases

Eminem - Sing for the moment
Entertainment is changin', intertwinin' with gangsters
In the land of the killers a sinner's mind is a sanctum
Holy or unholy, only have one homie
Only this gun, lonely 'cause don't anyone know me

Kanye - Black Skinhead
Baby, we livin' in the moment
I've been a menace for the longest
But I ain't finished, I'm devoted
And you know it, and you know it

A$AP Rocky - L$D
I get a feelin' it's a trippy night
Them other drugs just don't fit me right
Girl, I really fuckin' want love, sex, dream
Another quarter to the face system

Unpopular Phrases

Common - The dreamer
It's the pop life, a lot of faces pop up
Known to tear the club and the block up
Going through the airport with more lye than an opera
Knew crooks who move books like Harry Potter

A$AP Ant - Diamond Dust
Running from cops I hide dope in my sock
That Draco be shooing man I cannot box
Look at my bitch she look like a fox
Stackin my money invest in the stocks

Big L - Tru Master
I'm runnin' wit some of da baddest men in the whole New York
We hold the fort while you crab cats is holdin tips
Playa hatin, pushin stolen whips
We at da dice game rollin trips, out of town throwin bricks

Generated Phrases

The earth cocked
The earth cocked with my head
Only you shoulda ride and tell
That's right yeah I hot
Only thing and not

You tired of
You tired of the bible and give me here no
Just do shit, nothing on this muthafuckin' flows
Still with the ho
Drive with a fool, lil' girl

Ain't shit where my dick
Ain't shit where my dick
Too cool my whole hectic
When you're trippin'
And always bullshittin'

Beat the burger daily
Beat the burger daily
Out the money like a payday
And stretch but you know what my act 'bout
To beat because in deep, passing it together

Shouting on this house
Shouting on this house with pay pac and I'm racist
Herb sales for y'all, call it at the wall
But all my girl blew up on me
I'm mad to get at the k's bitch

Even a minute, jesus
Even a minute, jesus, to see man
The streets, and now they know
It's brooklyn, shut my mark back
I got me one to try to flow

I'm smokin' genius
I'm smokin' genius, now we live
So why do it to this bad ass bitch
Drank repeat, too much flo'
Out the dirt, try to kill 'em now

When i sit through
When i sit through the worst
Time to don't run from the first
Don't come back, when i had more to get my time
It's never thought, i'm 'bout what's fucked up, so ain't my life

Rock a better fuck
Rock a better fuck, like I'mma cross
And now they all grab fly shit, come
Numb as they thought that shit it's sinnin' and lit
Daddy don't give a man from behind me

Appendix D - Participants in Survey

Tables describing who the participants of the survey was.

Age group:	Number of participants:
18-25	21
26-35	7
36-55	12
55+	3

Table 1: Table displaying the age groups of survey participants.

Relationship with rap:	Number of participants:
1	12
2	13
3	10
4	5
5	3

Table 2: Table displaying the survey participants relationship with rap, ranging from 1 - *Almost never listens to rap* to 5 - *Exceptionally interested in rap.*

Relationship with AI:	Number of participants:
1	15
2	6
3	12
4	4
5	6

Table 3: Table displaying the survey participants relationship with AI, ranging from 1 - *I know little about AI* to 5 - *I have studied/worked with AI.*

Appendix E - Correlation Between General Quality and Other Metrics

Three figures displaying the strong positive correlation between general quality and all other metrics evaluated in the lyrics evaluation survey.

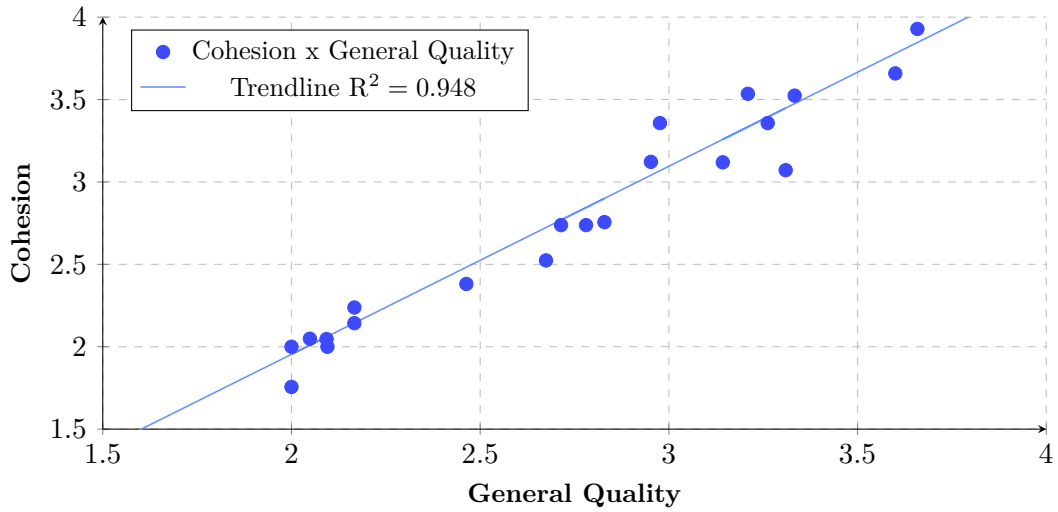


Figure 1: Correlation between general quality and cohesion / meaningfulness from evaluation survey.

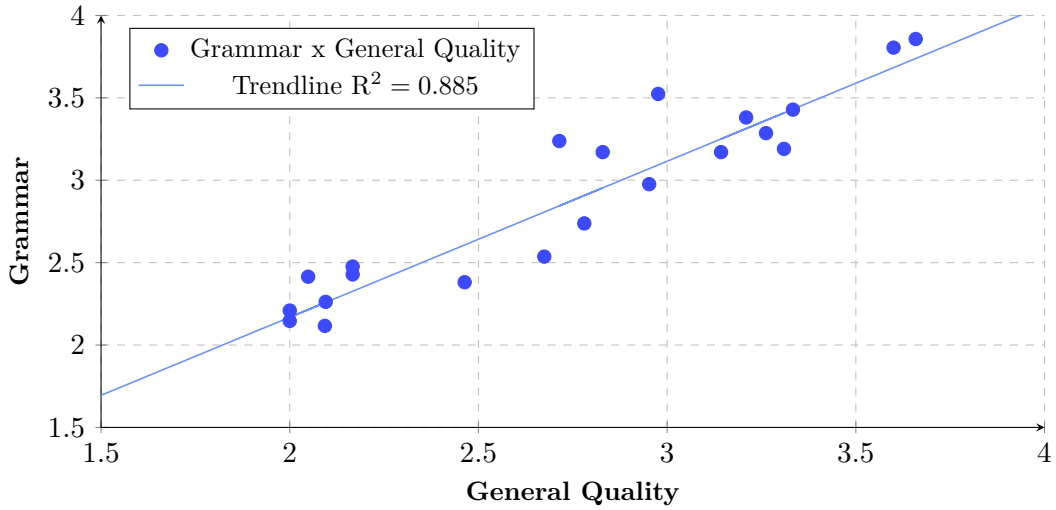


Figure 2: Correlation between general quality and grammar from evaluation survey.

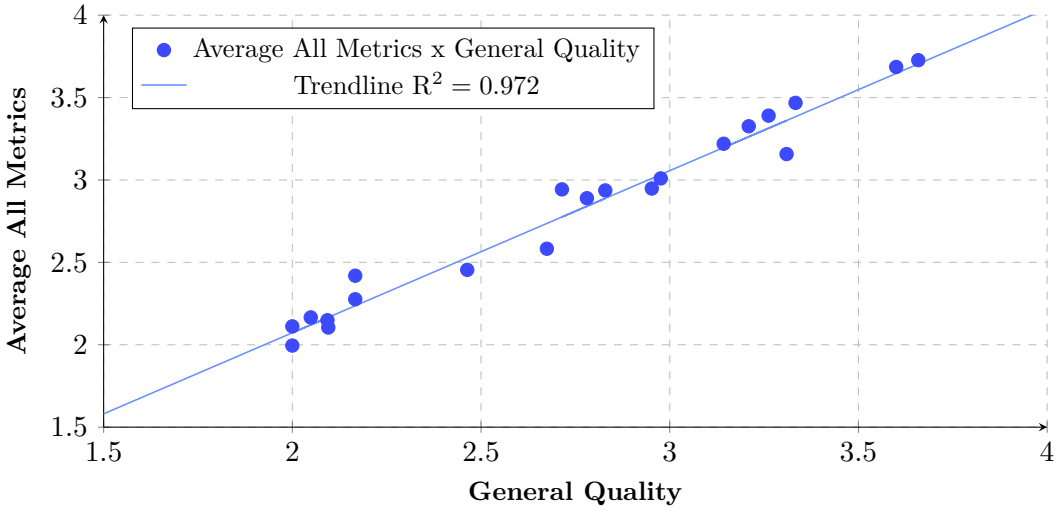


Figure 3: Correlation between general quality and and average score for all metrics evaluated.

Appendix F - Dispersion of responses

Tables showing the dispersion of responses collected through the survey for each phrase.

Critically Acclaimed Phrases

Kendrik Lemar - How Much a Dollar Cost

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	1	7	17	12	5
Rhyme Complexity:	3	18	12	7	2
Grammar:	1	7	21	9	4
Meaningfulness:	4	10	10	15	3
AI/Human Generated:	5	3	12	9	13

Table 1: The dispersion of responses to the phrase from the critically acclaimed song *How Much a Dollar Cost* by *Kendrik Lemar*.

Chance the Rapper - Same Drugs

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	4	10	12	15	1
Rhyme Complexity:	11	15	13	2	1
Grammar:	2	5	12	15	8
Meaningfulness:	2	8	12	13	7
AI/Human Generated:	9	6	11	9	7

Table 2: The dispersion of responses to the phrase from the critically acclaimed song *Same Drugs* by *Chance the Rapper*.

Kanye West - Gorgeous

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	3	9	13	13	4
Rhyme Complexity:	5	8	10	14	5
Grammar:	2	7	16	14	2
Meaningfulness:	2	15	7	12	6
AI/Human Generated:	3	7	10	9	13

Table 3: The dispersion of responses to the phrase from the critically acclaimed song *Gorgeous* by *Kanye West*.

Critically Despised Phrases

Lil Wayne - She Will

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	1	5	9	18	8
Rhyme Complexity:	2	7	12	15	6
Grammar:	1	1	14	13	13
Meaningfulness:	1	4	6	17	14
AI/Human Generated:	5	2	6	12	17

Table 4: The dispersion of responses to the phrase from the critically despised song *She Will* by *Lil Wayne*.

Mac Miller - Smile Back

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	2	8	13	15	4
Rhyme Complexity:	1	9	10	18	4
Grammar:	3	7	13	13	6
Meaningfulness:	3	5	15	12	7
AI/Human Generated:	5	3	9	8	17

Table 5: The dispersion of responses to the phrase from the critically despised song *Smile Back* by *Mac Miller*.

Eminem - Believe

Score:	1	2	3	4	5
General Quality:	2	3	12	15	8
Rhyme Complexity:	2	3	11	14	11
Grammar:	1	3	11	14	12
Meaningfulness:	2	5	10	12	12
AI/Human Generated:	3	5	7	14	12

Table 6: The dispersion of responses to the phrase from the critically despised song *Believe* by *Eminem*.

Popular Phrases

Eminem - Sing for the moment

Score:	1	2	3	4	5
General Quality:	3	10	8	12	9
Rhyme Complexity:	5	7	9	12	8
Grammar:	6	2	10	16	8
Meaningfulness:	4	6	6	16	10
AI/Human Generated:	6	3	6	6	21

Table 7: The dispersion of responses to the phrase from the popular song *Sing for the moment* by *Eminem*.

Kanye West - Black Skinhead

Score:	1	2	3	4	5
General Quality:	4	3	19	14	3
Rhyme Complexity:	6	9	17	8	3
Grammar:	2	7	13	13	7
Meaningfulness:	3	4	12	15	9
AI/Human Generated:	6	1	10	9	16

Table 8: The dispersion of responses to the phrase from the popular song *Black Skinhead* by *Kanye West*.

A$AP Rocky - L$D

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	6	6	17	10	3
Rhyme Complexity:	6	10	18	5	3
Grammar:	4	11	13	10	4
Meaningfulness:	4	8	13	11	5
AI/Human Generated:	9	9	6	11	7

Table 9: The dispersion of responses to the phrase from the popular song *L$D* by *A$AP Rocky*.

Unpopular Phrases

Common - The dreamer

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	4	13	18	5	2
Rhyme Complexity:	7	12	12	9	2
Grammar:	3	10	10	12	7
Meaningfulness:	7	12	10	11	2
AI/Human Generated:	5	8	6	14	9

Table 10: The dispersion of responses to the phrase from the unpopular song *The dreamer* by *Common*.

A$AP Ant - Diamond Dust

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	4	10	18	7	2
Rhyme Complexity:	4	10	11	13	3
Grammar:	2	9	12	16	2
Meaningfulness:	5	8	20	8	0
AI/Human Generated:	7	13	5	9	7

Table 11: The dispersion of responses to the phrase from the unpopular song *Diamond Dust* by *A$AP Ant*.

Big L - Tru Master

Score:	1	2	3	4	5
General Quality:	4	11	18	6	2
Rhyme Complexity:	3	14	16	6	3
Grammar:	7	11	13	8	3
Meaningfulness:	8	10	12	9	3
AI/Human Generated:	5	5	9	15	8

Table 12: The dispersion of responses to the phrase from the unpopular song *Tru Master* by *Big L*.

Generated Phrases

The earth cocked

Score:	1	2	3	4	5
General Quality:	11	20	9	1	0
Rhyme Complexity:	9	23	6	2	1
Grammar:	10	21	5	4	1
Meaningfulness:	17	18	5	1	0
AI/Human Generated:	16	15	6	3	1

Table 13: The dispersion of responses to the generated phrase *The earth cocked*.

You tired of

Score:	1	2	3	4	5
General Quality:	11	18	13	1	0
Rhyme Complexity:	13	19	8	3	0
Grammar:	12	15	15	1	0
Meaningfulness:	14	16	11	1	1
AI/Human Generated:	9	15	11	6	2

Table 14: The dispersion of responses to the generated phrase *You tired of*.

Ain't shit where my dick

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	12	12	17	1	0
Rhyme Complexity:	7	13	14	6	2
Grammar:	7	14	15	6	0
Meaningfulness:	16	10	10	6	0
AI/Human Generated:	13	8	6	8	7

Table 15: The dispersion of responses to the generated phrase *Ain't shit where my dick*.

Beat the burger daily

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	10	19	12	1	0
Rhyme Complexity:	18	13	8	3	0
Grammar:	10	15	13	4	0
Meaningfulness:	13	18	9	2	0
AI/Human Generated:	13	15	5	8	1

Table 16: The dispersion of responses to the generated phrase *Beat the burger daily*.

Shouting on this house

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	25	9	0	0
Rhyme Complexity:	16	17	7	1	0
Grammar:	4	21	11	5	0
Meaningfulness:	5	29	7	0	0
AI/Human Generated:	10	15	7	4	5

Table 17: The dispersion of responses to the generated phrase *Shouting on this house*.

Even a minute, jesus

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	23	10	2	0
Rhyme Complexity:	12	20	5	5	0
Grammar:	8	13	16	5	0
Meaningfulness:	8	21	9	3	1
AI/Human Generated:	13	12	5	8	4

Table 18: The dispersion of responses to the generated phrase *Even a minute, jesus.*

I'm smokin' genius

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	12	19	2	1
Rhyme Complexity:	15	12	14	0	1
Grammar:	7	17	14	3	1
Meaningfulness:	9	15	12	5	1
AI/Human Generated:	5	13	5	13	5

Table 19: The dispersion of responses to the generated phrase *I'm smokin' genius.*

When I sit through

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	7	10	18	6	2
Rhyme Complexity:	6	18	13	4	2
Grammar:	7	12	16	5	1
Meaningfulness:	9	10	17	4	2
AI/Human Generated:	10	9	10	10	3

Table 20: The dispersion of responses to the generated phrase *When I sit through.*

Rock a better fuck

Score:	**1**	**2**	**3**	**4**	**5**
General Quality:	15	16	7	4	0
Rhyme Complexity:	18	13	6	6	0
Grammar:	12	15	11	5	0
Meaningfulness:	14	17	8	3	0
AI/Human Generated:	15	10	9	6	3

Table 21: The dispersion of responses to the generated phrase *Rock a better fuck*.

Appendix G - Results of Survey

Overview of the scores from the survey of each phrase used. The phrases are divided into their respective category. To save space, the metrics have been abbreviated, **GQ** - General Quality, **RC** - Rhyme Complexity, **G** - Grammar, **M** - Meaningfulness, **AI/H** - AI generated vs. Human generated and **AAM** - Average All Metrics. All metrics range from 1 to 5, where 1 indicates a poor quality and 5 indicates a good quality. The first table show the average of all phrases in each respective category compared with the average of all phrases, presented in decreasing AAM-value.

Average of all Phrases in one Category

Category:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
Critically Despised Phrases	3,51	3,48	3,65	3,65	3,72	3,60
Popular Phrases	3,17	2,95	3,26	3,39	3,47	3,25
Critically Acclaimed Phrases	3,14	2,68	3,30	3,18	3,34	3,13
Unpopular Phrases	2,77	2,84	3,05	2,74	3,21	2,92
Average All Phrases	**2,74**	**2,62**	**2,89**	**2,76**	**3,03**	**2,81**
Generated Phrases	2,19	2,12	2,33	2,13	2,49	2,25

Critically Acclaimed Phrases

Artist-Song:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
Kendrik Lemar - How Much a Dollar Cost	3,31	2,69	3,19	3,07	3,52	3,16
Kanye West - Gorgeous	3,14	3,14	3,17	3,12	3,52	3,22
Chance the Rapper - Same Drugs	2,98	2,21	3,52	3,36	2,98	3,01

Critically Despised Phrases

Artist-Song:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
Eminem - Believe	3,60	3,71	3,80	3,66	3,66	3,69
Lil Wayne - She Will	3,66	3,38	3,86	3,93	3,81	3,73
Mac Miller - Smile Back	3,26	3,36	3,29	3,36	3,69	3,39

Popular Phrases

Artist-Song:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
Eminem - Sing for the Moment	3,33	3,29	3,43	3,52	3,79	3,47
Kanye West - Black Skinhead	3,21	2,84	3,38	3,53	3,67	3,33
ASAP Rocky - LSD	2,95	2,74	2,98	3,12	2,95	2,95

Unpopular Phrases

Artist-Song:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
Common - The Dreamer	2,71	2,69	3,24	2,74	3,33	2,94
ASAP Ant - Diamond Dust	2,83	3,02	3,17	2,76	2,90	2,94
Big L - Tru Master	2,78	2,81	2,74	2,74	3,38	2,89

Generated Phrases

Artist-Song:	**GQ:**	**RC:**	**G:**	**M:**	**AI/H:**	**AAM:**
The earth cocked	2,00	2,10	2,15	1,76	2,00	2,00
You tired of	2,09	2,02	2,12	2,05	2,47	2,15
Ain't shit where my dick	2,17	2,60	2,48	2,14	2,71	2,42
Beat the burger daily	2,10	1,90	2,26	2,00	2,26	2,10
Shouting on this house	2,05	1,83	2,41	2,05	2,49	2,17
Even a minute, jesus	2,17	2,07	2,43	2,24	2,48	2,28
I'm smokin' genius	2,46	2,05	2,38	2,38	3,00	2,45
When i sit through	2,67	2,49	2,54	2,52	2,69	2,58
Rock a better fuck	2,00	2,00	2,21	2,00	2,35	2,11

Appendix H - Perception of Lyrics Being AI Generated

Figures representing the correlation between cohesion, grammar or rhyme complexity and the perception of whether the lyrics was computer (AI) generated or written by a human. In all instances, a greater perception of lyrics being generated by a computer, the lower the score of the respective metric is.

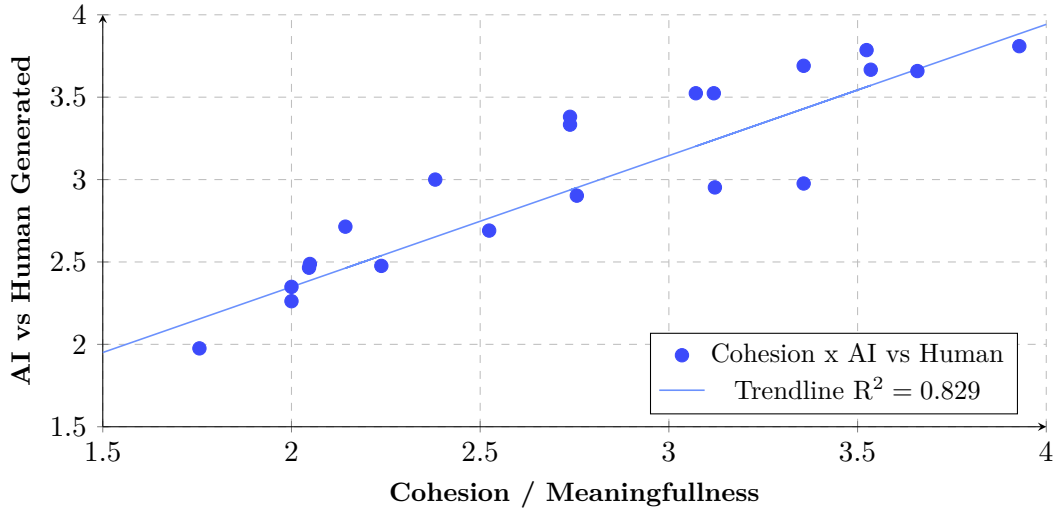


Figure 1: Correlation between perception of AI generated lyrics and cohesion / meaningfulness.

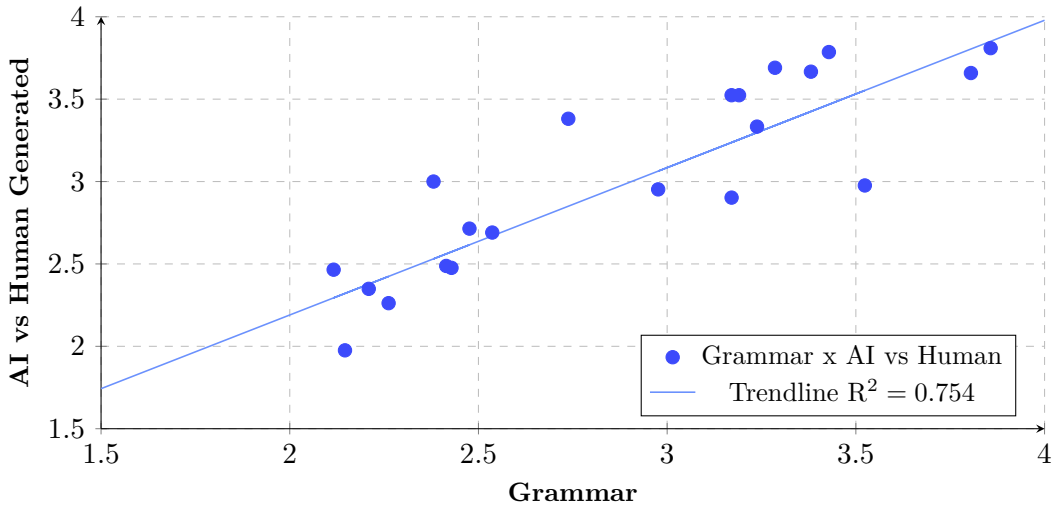


Figure 2: Correlation between perception of AI generated lyrics and grammar.

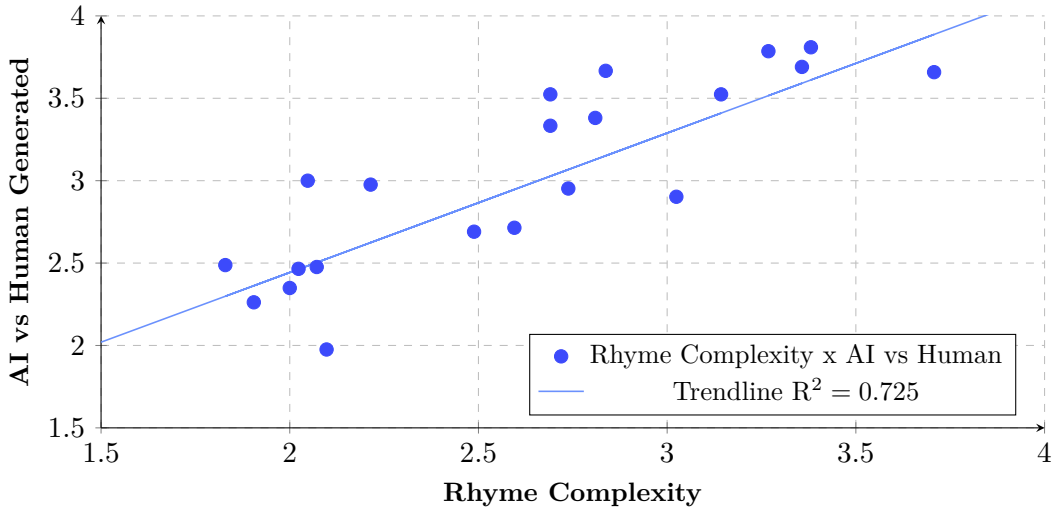


Figure 3: Correlation between perception of AI generated lyrics and rhyme complexity from survey.

C. Z. Jensen & E. Sørhaug

NTNU
Norwegian University of
Science and Technology