ANALYSIS AND DETECTION OF SECTIONS OF ENGLISH POP SONG LYRICS USING TRANSFER LEARNING FROM THE LONGFORMER MODEL

Willy Reiji Nurhuda Ekaputra
Universitas Pakuan

ABSTRACT

Understanding the structural composition of song lyrics is essential for various applications, including music recommendation, summarization, and computational creativity. In this study, we explore automated section classification of song lyrics—specifically identifying parts such as verse, chorus, bridge, and others—using a transformer-based model. We fine-tuned a Longformer model on a dataset of 10,000 English pop lyrics with annotated section labels. The model was trained as a token classification task without the use of global attention, relying solely on local context to capture structural cues. Despite working with a significantly reduced dataset and limited training resources, the model achieved strong performance on the dominant structural classes, reaching F1-scores of 0.78 for verse and 0.77 for chorus. Secondary and infrequent sections such as bridge and prechorus showed moderate performance, while more ambiguous categories like postchorus and other were less accurately predicted. Analysis of the confusion matrix revealed that most misclassifications occurred between semantically overlapping sections, particularly among chorus-adjacent types. The results demonstrate that transformer models can effectively learn lyric structure from text alone, even with constrained data and without musical input. Our findings suggest that such models can serve as a strong foundation for future lyric analysis systems and that performance can be further improved through dataset expansion, label refinement, and multimodal integration.

Keywords: *Lyrics Segmentation, Section Classification, Transformer, Longformer, Natural Language Processing, Music Structure Analysis, Chorus Detection, Lyric Modeling.*

1. INTRODUCTION

Song lyrics follow a structured format that contains specific sections such as intro, verse, pre-chorus, chorus, bridge, and outro. Each of these sections plays a distinct role in conveying the overall meaning of the song. Identifying these sections is crucial for various applications, such as lyric analysis, AI-assisted songwriting, song information retrieval, and AI-generated music. However, unlike general document text, song lyrics are often repetitive and non-linear, making detection more difficult.

Most previous research on music section detection has focused solely on the chorus section of music, such as in [1] and [2], treating it as a binary classification problem (chorus and non-chorus). These studies only achieved F1-scores ranging from 59.24% to 67.4%. These models generally use repetition in lyrics as the primary feature for detecting choruses. However, this method typically fails to capture the overall song structure because most other song sections or complex song structures do not follow repetitive pattern rules, unlike choruses.

Research such as [3] and [4] only uses audio to detect choruses. The first study uses a CNN-based model and is also one of the first studies to use this method to detect choruses from audio; this study consistently achieved an F1-score above 60%. In the second study, an end-to-end deep learning model using self-attention and multi-scale convolutional networks was employed to learn features from mel- spectrograms. This model was named DeepChorus and achieved a score of 67.5%, with the lowest score being 50%. Although

both studies sound promising, they both require audio to function properly. Additionally, the small amount of data has a significant impact on model performance.

In addition to using plain text, there are also several studies that combine both audio and lyrics to segment song lyrics, such as the studies by [3] and [5]. Both models use quite different approaches to song segmentation. In the [3] study, a BERT model was used, combined with a Graph Attention Network (GAT) to process lyrics, and MFCC using chord embeddings to process audio. This study yielded significantly better evaluation results compared to previous studies, with an accuracy of 85.94% and an F1-score of 85.67%. Meanwhile, the research by Fell et al., 2022, used CNN-based segmentation that combined SSM with audio Mel-spectograms using MFCC. This model only achieved an F1-score of 75.3% when text and audio were combined to predict the chorus. Both models in these studies had difficulty detecting rap and hip-hop songs, where the lyrics are less repetitive to highlight meaningful patterns.

These studies share a common limitation: they can only detect one part of the song, the chorus. This is because the chorus is the easiest part of the song to predict due to its repetitive nature. Additionally, chorus detection research using text alone cannot achieve satisfactory accuracy for reliable use. Using audio and lyrics can improve accuracy, but not all song search services can easily obtain audio files, and audio analysis also takes excessive time, limiting its use. To address this issue, this study proposes a deep learning model to detect multi-class sections in song lyrics through transfer learning from existing models such as Longformer to accelerate training time and improve model accuracy. Unlike traditional methods that rely solely on lyric repetition to identify choruses, this approach aims to detect and classify all important sections within a song.

2. METHODS

2.1. Song Structure and Lyrics Sections

Song structure refers to how a song is composed, using several different sections, each of which has a specific role in the mood and tempo of a piece of music. Music generally includes verses, choruses and bridges in the following order: intro — verse — chorus — bridge — chorus — outro [6]. The following are the sections of a song that are usually adopted in lyrics:

1. Introduction (Intro): This is the opening section of the song, introducing the main theme or context to the listener.
2. Verse: This section is often used to explain the narrative of the song, if applicable.
3. Pre-Chorus: This section is located before the chorus and is typically optional, not required before every chorus.
4. Chorus: The most important part of a song. This section usually has repetitive and memorable lyrics from the entire song, emphasising the main theme and meaning of the song.
5. Post-chorus: This section is usually located after the chorus and is often used to reinforce the melody and conclude the chorus, creating a smooth transition to the next section of the song.
6. Bridge: This section is usually located at the end of the song before the final chorus,

placing the verse before the chorus. The bridge is often used to replace the third verse, which has a different and unique emotion from the previous verses.

7. Outro: Not always present, this section is located at the end of the lyrics and is usually used as the final conclusion of the song's main theme.

The lyrics in the table Table 1 are examples of song lyrics taken from the song 'Hello' by Adele. This song has a structure commonly used in pop music, with clear and easily recognisable sections. However, not all songs follow this structure strictly. Some songs may have more complex or different sections, depending on the style and purpose of the song.

Table 1. Lyrics from Adele's song 'Hello'

Verse 1	Hello, it's me I was wondering if, after all these years, you'd like to meet To go over everything They say that time's supposed to heal ya But I ain't done much healin'
Pre-Chorus	There's such a difference between us And a million miles
Chorus	Hello from the other side I must've called a thousand times To tell you I'm sorry for everything that I've done But when I call, you never seem to be home
Verse 2	Hello, how are you? It's so typical of me to talk about myself, I'm sorry I hope that you're well Did you ever make it out of that town Where nothing ever happened?
Pre-Chorus	It's no secret that the both of us Are running out of time
Chorus 2	So hello from the other side (Other side) I must've called a thousand times (Thousand times) To tell you I'm sorry for everything that I've done But when I call, you never seem to be home

2.2. Research Model

This research applies a model called Knowledge Discovery in Databases (KDD). KDD is the auto- matic and exploratory analysis and modelling of large data repositories in an organised process to identify valid, unique, useful, and understandable patterns from complex datasets. Data Mining is the core process of KDD, encompassing a set of algorithms that explore data, develop models, and discover previously unknown patterns [7]. Figure 1 illustrates the common stages of the KDD process.

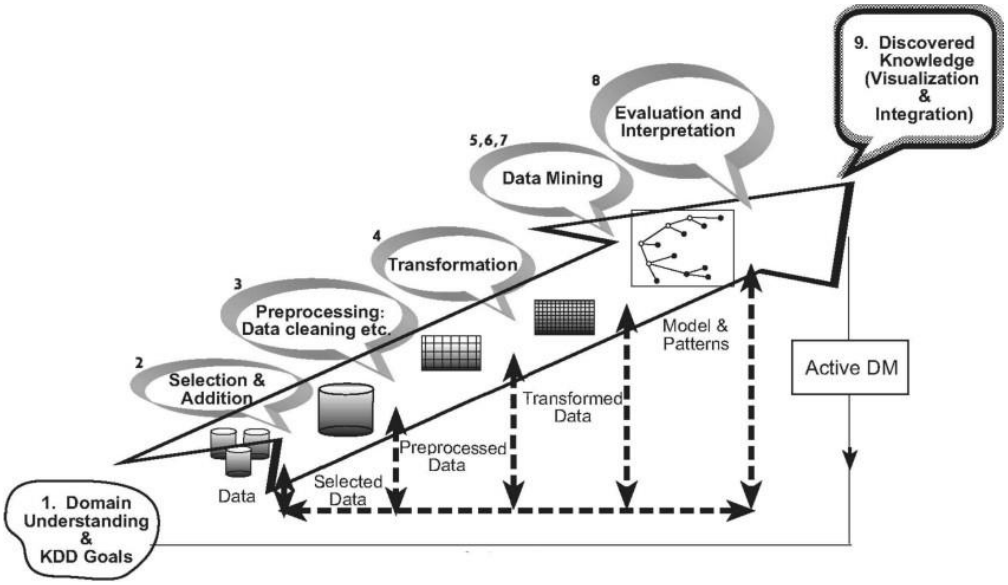

Figure 1. Proses Knowledge Discovery in Database [7]

2.3. Data Selection

The research dataset was obtained from a collection of song lyrics obtained through web scraping on Genius.com, a community platform that provides song lyrics and metadata.

The scraping results from Genius.com were obtained through the Kaggle website, which was created in 2022 with a dataset of approximately 3 million songs. The dataset features can be viewed at Table 2.

To ensure the reliability of the data in this research, a selection process will be carried out following certain criteria. These criteria include language, genre, and song labels. The language used is English because most of the song lyrics on Genius are in English, and this language was also chosen because text processing models typically find it easier to process. The genre used is pop music, supported by previous research [8], where country and pop songs have the highest chorus detection results for capturing song segments. Pop music was chosen over country music due to its larger volume.

Table 2. Dataset Features

Data Features	Description
Title	Song title, mostly songs, but there are also books, poems, and other forms of work
Tag	Song genre according to page classification

Artist	Song creator
Year	Year of release of the song
Views	Number of visitors to the website page
Features	Additional information about the artist who contributed to the song
Lyrics	The main text of the song
Id	Unique identifier assigned by Genius
Language_cld3	Lyric language according to CLD3, unclear results marked as NaN
Language_ft	Lyric language according to FastText, results with low confidence levels (< 0.5) marked as NaN
Language	combination of language_cld3 and language_ft, only has a value if both engines agree on one language, otherwise it will be marked as NaN

Section labels (verse, chorus, bridge, etc.) in lyrics are essential for accelerating the labelling process for transfer learning. Therefore, lyrics without section labels will be separated but can still be used for test data. Using these criteria, the dataset size can be significantly reduced, yet it remains reliable as a corpus for model training and evaluation.

2.4. Data Preprocessing

Preprocessing will be performed to ensure consistency in text processing. There are several methods commonly used in text mining in general, but the following are the decisions chosen for data preprocess- ing:

1. Lowercasing: All text in the lyrics is converted to lowercase to reduce variations caused by uppercase letters. Depending on the type of tokenizer used, capitalization can affect how the model interprets the meaning of a word (e.g., “apple” for the fruit and “Apple” for the electronics company). Capitalization is important for NER, but since this only detects song lyrics, capital- ization is not required.
2. Punctuation and special characters: Transformer models like BERT and Longformer can recog- nize context from punctuation. Since Longformer’s token limit allows us to load many tokens at once, it is better to keep punctuation and special characters to ensure that the model can under- stand the entire context of the song.
3. Standardization of Section Labels: The selected lyrics already have their own section labels, but variations may occur due to inconsistent formatting. To ensure all labels have the same names, a specific naming scheme has been chosen to replace all these labels. This step helps the model recognize the boundaries of song lyrics sections.

In addition to the above, traditional NLP preprocessing methods, such as stopword removal and lemmatization, are not performed. Both of these can contribute to the formation of song structure and may reduce the effectiveness of the model by altering the structural role of each word in the lyrics. This is compounded by the fact that lyrics often contain non-standard words, making stopword detection and lemmatization unpredictable.

2.5. Longformer

Longformer is an extension of RoBERTa that uses sparse attention techniques to reduce the computa- tional and memory complexity required to process long inputs. By using sparse attention, Longformer can process inputs up to 4096 tokens or more in length, thereby reducing complexity.

Longformer uses sliding window attention where each token only sees its neighboring tokens at a predetermined distance rather than the entire sequence. This can expand the input tokens that the model can handle, but with the addition of many layers to capture long-range dependencies. Then each attention head will be performed for each local window and each output from the attention head will be concate- nated, as in the standard multi-head attention performed previously.

In the lyric section detection task of this research, global attention is used for every token from the first line of the input. However, a version for global attention use will also be used during the training phase to evaluate whether local dependencies are sufficient for detecting lyric section transitions.

2.6. Transformer Model Architecture

Transformer models generally have two distinct main components, the encoder and the decoder. The encoder is responsible for processing the input sequence, while the decoder is more focused on generating the output sequence based on the input.

The decoder has an additional layer in its processing. One of these is Masked Multi-Head Attention. This attention mechanism works similarly to Multi-Head Attention, but the sentences trained on this layer prevent the model from seeing subsequent words, focusing only on previous words to emphasize cause-and-effect attention. After passing through this layer, the model performs cross-attention to view the results of the encoder output, adding important information before proceeding to the Feed-Forward layer, creating new words as output based on the results of the encoder output and training the decoder. Not all Transformer models use a decoder component; one exception is the BERT and Transformer models. Figure 2 Shows how this structure is formed in Transformer models.

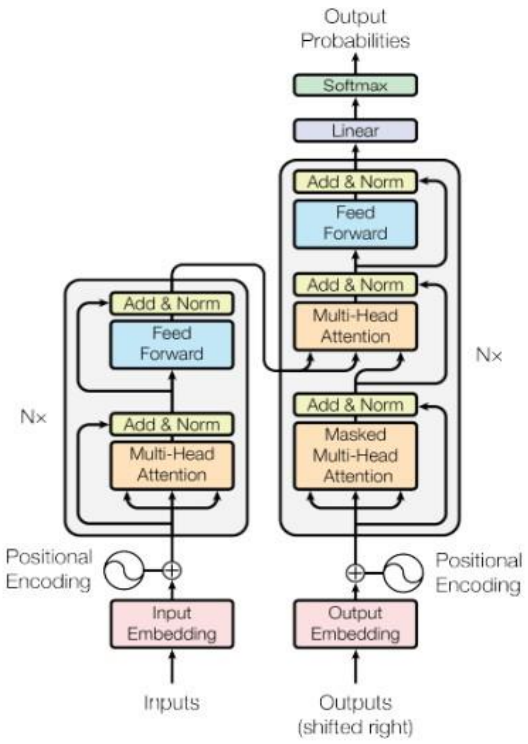

Figure 2. Encoder-Decoder Architecture in Transformer [9]

Just like other transformation models, text input must first be converted into a numerical representation that the model can understand. This can be done through tokenization and input embedding. BERT and Longformer use a technique called WordPiece Tokenization as their tokenization format. This method divides words into sub-word units, allowing the model to handle words outside the vocabulary, thereby improving efficiency. After tokenization, the text must be converted into numerical vectors before being processed by BERT. This process is similar to input embedding in a standard transformer model, but with a few additional steps.

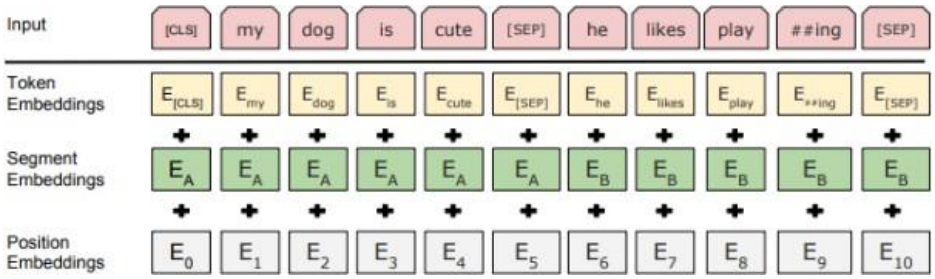

Figure 3. Layer Embedding [10]

In the BERT model, and by extension, Longformer, there are three embedding stages: token embedding, segment embedding, and position embedding, as shown by Figure 3. Together, these three embeddings can help the model understand the given text sequence.

2.7. Tokenization

Tokenization is the first step in processing text data for transformer-based neural network models. Since neural networks operate using numerical data, raw text must first be converted into numerical form. This can be achieved by using tokenization, which divides text into smaller units called tokens before feeding it to the [11] model.

The input model takes text that has undergone preprocessing and converts it into tokens that can be used by Longformer for processing. This tokenization format includes tokenization form, section markers, and padding.

Each song section label is converted into a special token that serves as an explicit indicator and delimiter, helping the model distinguish between sections without relying on pattern recognition. Each token following the special label token is then assigned a predefined numerical value corresponding to the label order in image x (each special token is assigned the numerical label 0). This labeling strategy ensures that every word or token in the lyrics is closely related to a section, making transitions between sections easier for the model to interpret. Each token is then converted into a token ID for the model to interpret. A token ID is a numeric index defined by the tokenizer.

The task of detecting song lyric sections is formulated as a sequence classification problem, where each input sequence (song lyrics) is assigned a lyric section label (verse, chorus, bridge, pre-chorus, post- chorus, intro, outro). Due to the structure of song lyrics, where sections follow a sequential pattern, the model must understand both the content and transitions between lyric sections.

2.8. Model Training Strategy

Evaluation is performed to assess the results of the modeling. The methods used are Precision, Recall, and F1-Score to calculate the prediction performance for each label, and accuracy is used to calculate the overall model accuracy.

To ensure a smooth model process, a strategy is implemented to balance performance with existing limitations, such as time and computational resources. Some of the strategic decisions made are listed in Table 3.

Table 3. Model Trainer Strategy

Parameter	Decision	Description
Dataset Size	10,000 English pop lyrics	reduced to limit training time so that the set time can be realized.
Evaluation and saving strategy	steps	To ensure the model can be saved periodically without waiting for the epoch to complete. This is done to mitigate unexpected errors and enable regular evaluation.
number of steps	50	The model is saved and evaluated every 50 steps to track the model's progress more frequently
Batch size	2	Reducing the batch size to mitigate device limitations at the expense of processing speed.
learning rate ()	5×10^{-5}	Constant variable
weight_decay	0.01	Constant variable
Global Attention	Not Used	Since the lyrics used are not too long, Local attention is usually sufficient to maintain context

3. RESULT AND DISCUSSION

3.1. Training and Validation Loss

Training and validation loss are two metrics used to evaluate models during the training process. Figure 4 shows the trend of model training for every 50 steps, starting from the evaluation and training itself. The blue line represents the training loss and the red line represents the validation loss. The lower the value, the better. If the validation loss is significantly higher than the training loss, it indicates that the model is overfitting.

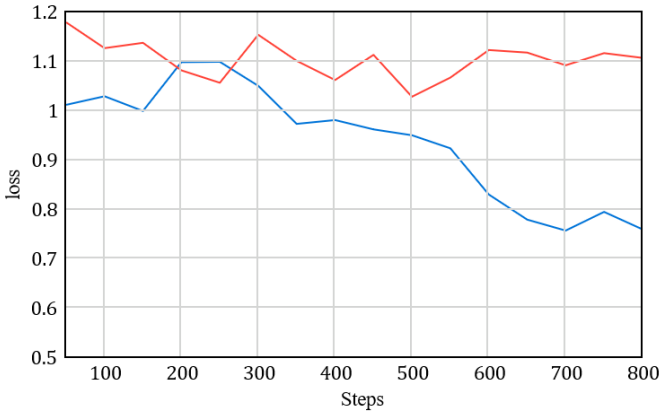

Figure 4. Training and Validation Loss Graph

3.2. Evaluation Metrics

To perform additional evaluation to ensure model performance, a classification report containing information about precision, recall, and F1-Score is used, and support is the number of actual labels in the training dataset. This evaluation is performed on the best model after training and is conducted on 1000 datasets from the sample division of the training and test datasets. Table 4 contains the results of this metric evaluation. The overall model accuracy is 67%.

Table 4. Classification Evaluation

label	Precision	Recall	F1-Score	Support	Support
verse	0.76	0.81	0.78	116246	33.86%
chorus	0.74	0.81	0.77	119509	34.81%
bridge	0.46	0.64	0.53	19532	5.69%
outro	0.45	0.56	0.50	13288	3.87%
intro	0.64	0.44	0.52	7076	2.06%
prechorus	0.52	0.55	0.53	19562	5.7%
postchorus	0.21	0.10	0.14	5590	1.63%
other	0.53	0.14	0.22	42536	12.39%
Total				343339	100%

3.3. Confusion Matrix

The Confusion Matrix is also implemented in the evaluation to see which labels were incorrectly predicted by the model. This is useful for drawing conclusions about the behavior

and limitations of the model in predicting one or more models. Figure 5 provides a visualization of the model's prediction results and which labels were correctly and incorrectly predicted.

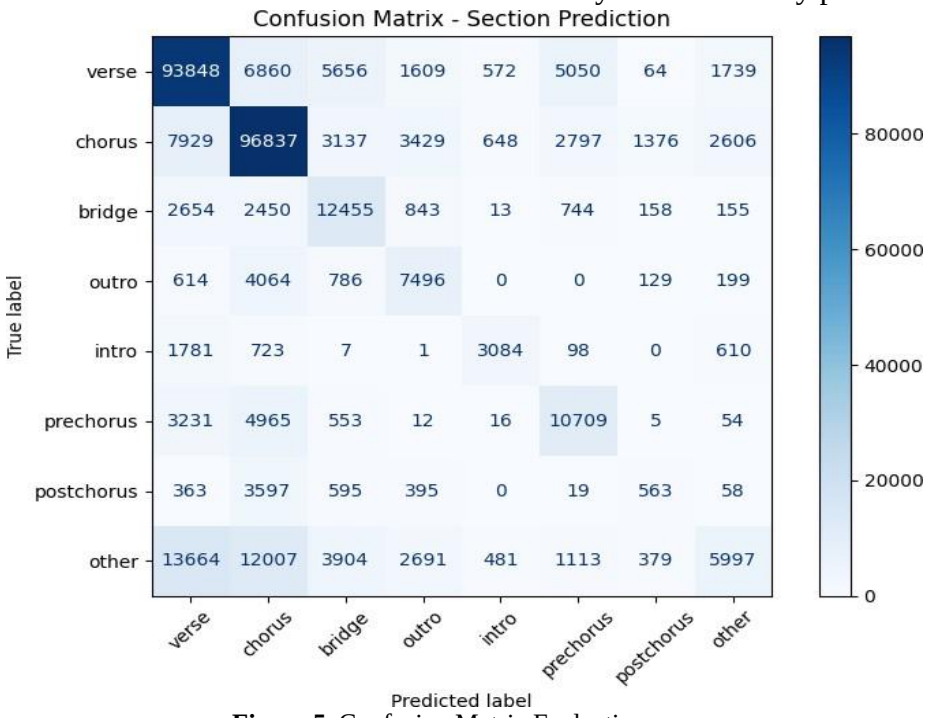

Figure 5. Confusion Matrix Evaluation

3.4. Qualitative Evaluation

To evaluate the model qualitatively, it was given three different song lyrics, each with its own unique characteristics. The first lyrics were "Your Reality" by [12], "The Words I Never Said in DnB" by [13], and "Clarity" by [14]. The song lyrics were also obtained from Genius and include a label section to aid in qualitative evaluation.

3.5. Discussion

3.5.1. Training and Validation Loss

As seen in Figure 4, where training and validation loss are monitored at intervals (every 50 steps), it can be observed that training loss shows a fairly consistent downward trend, indicating that the model successfully learned something from the dataset used. The data starts at ∼ 1.15, increases until step 300, and then begins a significant decline until step 700.

However, the validation loss shows a different pattern. Initially, this loss decreases alongside the train- ing loss, even intersecting with the training loss at step 200 − 300, but this trend changes after that point, fluctuating consistently around 1.0 − 1.2. This may indicate that the model is beginning to memorize its training data but may face difficulties in generalizing to external data.

The behavior of these two loss metrics is important for determining future training decisions, espe- cially with signs of overfitting at this early stage. Continuing training without making any changes could have adverse effects, but since this occurs at an early stage, training may still be continued. However, due to time and computational constraints, training was

prematurely terminated at the third epoch with the hope that the model would perform well despite these limitations.

3.5.2. Classification Metric Evaluation

Table 4 shows the performance of the model on each label. The following is a summary of the overall model performance:

1. Verse and Chorus are the labels with the best performance across the entire system, with F1- Scores of 0.78 and 0.77. This is even more impressive considering the model was trained using only 10,000 samples (out of a total of approximately 250,000 lyrics) to optimize training time and computational resources. The model has demonstrated the ability to predict these two structures.
2. The Bridge, Outro, Intro, and Prechorus labels have moderate F1-scores, between 0.50 and 0.53, meaning the model can capture the patterns of these labels, albeit with low confidence.
3. The Postchorus and Other labels have low scores, only 0.14 and 0.22. These sections are typically noisier and less consistent than other labels, making them difficult to predict.

Overall, this model can rival traditional models used in the referenced paper (such as by [2]) when used solely for detecting Chorus. However, this model does not perform well for other song sections besides Verse and Chorus.

3.5.3. Confusion Matrix

The Confusion Matrix in Figure 5 shows some common classification errors, such as:

1. Verse and Chorus labels are often mixed up. This can happen if the Chorus of a song differs from other Choruses, which disrupts its repetitive nature, making it difficult to predict without the audio or musical context.
2. Bridge and Prechorus are often confused with Chorus and Verse. This is understandable given their placement between Chorus and Verse.
3. Postchorus is the worst label to predict, with only 563 correct predictions out of approximately 9,000 tokens. This label is more often classified incorrectly than correctly. This may occur because the model is unable to distinguish it from other sections.
4. The Other label, like Postchorus, is a poor label. Although the total number of correct predictions is higher, many other labels are incorrectly classified using this label. This makes it an ambiguous label for the model, lacking any distinct characteristics.
5. Intro and Outro are also quite weak, but compared to the Other and Postchorus labels, these labels can still be recognized by the model. This limitation may stem from their short nature and often having boundaries that are too similar to the next section, making them easy to confuse.

Overall, this Confusion Matrix highlights the model's strengths and challenges in predicting song sections. The model has a significant tendency to favor strong dominants (such as chorus and verse) when faced with uncertainty. This can be addressed through further labeling adjustments, a larger dataset, or alternative segmentation strategies in the future.

3.5.4. Qualitative Evaluation

The model's predictions were performed on three different lyrics. Each of the three lyrics has its own characteristics, which are clearly evident in the prediction results.

1. The lyrics of "Your Reality" have a fairly inconsistent structure. The short and non-repetitive chorus (textually) and ambiguous verse boundaries make it difficult for the model to provide accurate labels. This lyric is one type where the division is determined by the musical context, not the text.
2. The lyrics of "The Words I Never Said," on the other hand, have a more consistent and repetitive structure. The model can accurately predict the appropriate labels with minimal errors, such as labeling the bridge and delayed section boundaries.
3. The lyrics of the song "Clarity" have a mix of consistent structure and unique labels, such as the pre-chorus. The model can accurately predict some clear labels but struggles with meaningful boundaries. The model also fails to predict the pre-chorus, defining it as the chorus.

Overall, this evaluation demonstrates that the model can produce satisfactory results if the provided lyrics have a consistent and transitional structure. The model will encounter difficulties if the lyrics lack a traditional structure, such as lyrics with heavy narrative segments.

4. CONCLUSION

This study investigates whether classifying song lyrics using a transformer model, namely the Long- former architecture, can accurately identify the parts of a song's lyrics.

Although this study limited the use of the dataset to 10,000 due to time and resource constraints, the model was able to produce promising results. For example, the model achieved an F1-Score of 0.78 for the verse section and 0.77 for the chorus section, which is quite close to previous studies, such as [2], which achieved an accuracy of 85.4% in detecting song choruses with 100,000 lyrics using a rule-based method with semantic embedding.

This model performs well in predicting dominant labels, such as verse and chorus, in lyrics with consistent structures. However, its performance drops significantly if the song lacks a consistent chorus structure, which can lead to ambiguity without audio context. Additionally, the model still has poor performance for transitional parts, such as pre-chorus and bridge, preferring them as part of the chorus or verse, and the model is even worse at handling classes with low distribution or ambiguity, such as bridge, pre-chorus, and outro. Both of these issues are reinforced by the results of a qualitative evaluation of three different lyrics with their own characteristics.

This research is primarily limited by computational and time constraints, which affect the volume of the training dataset and the number of training epochs. Additionally, some sections particularly the other and post-chorus sections are inherently ambiguous, both in definition and in annotation quality, which introduces noise during training and evaluation.

Further development could explore several potential avenues to improve model performance, such as expanding the dataset to a full dataset of $\sim$ 250, 000 song lyrics to enhance machine learning capabilities, incorporating musical structure or audio features to complement

this text-based model, implementing a better labeling strategy, such as hierarchical classification to address ambiguous labels, and adding or expanding the model's ability to predict genres and languages beyond pop and English.

REFERENCES

K. Watanabe and M. Goto, "A method to detect chorus sections in lyrics text," *IEICE Transactions on Information and Systems*, no. 9, pp. 1600–1609, 2023.

K. Watanabe *et al.*, "Modeling discourse segments in lyrics using repeated patterns," *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, vol. 26, no. 9784879747020, pp. 1959–1969.

J. Wang, Z. Li, B. Gu, T. Zhang, Q. Liu, and Z. Chen, "Multi-modal Chorus Recognition for Improving Song Search," in *Lecture Notes in Computer Science*, Springer International Publishing, 2021, pp. 427–438.

Q. He, X. Sun, Y. Yu, and W. Li, "Deepchorus: A hybrid model of multi-scale convolution and self-attention for chorus detection," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 411–415.

M. Fell, Y. Nechaev, G. Meseguer-Brocal, E. Cabrio, F. Gandon, and G. Peeters, "Lyrics segmen- tation via bimodal text–audio representation," *Natural Language Engineering*, vol. 28, no. 3, pp. 317–336, 2021.

"Songwriting 101: Learn Common Song Structures - 2025."

O. Maimon and L. Rokach, "Introduction to Knowledge Discovery in Databases," in *Data Mining and Knowledge Discovery Handbook*, Springer-Verlag, pp. 1–17. [Online]. Available: https://doi. org/10.1007/0-387-25465-x_1

M. Fell, Y. Nechaev, E. Cabrio, and F. Gandon, "Lyrics Segmentation: Textual Macrostructure Detection using Convolutions," *COLING 2018 - 27th International Conference on Computational Linguistics, Proceedings*, vol. 0, pp. 2044–2054.

A. Vaswani *et al.*, "Attention Is All You Need." [Online]. Available: https://arxiv.org/abs/1706. 03762

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." [Online]. Available: https://arxiv.org/abs/1810.04805

V. S and J. R, "Text Mining: open Source Tokenization Tools – An Analysis," *Advanced Compu- tational Intelligence: An International Journal (ACII)*, vol. 3, no. 1, pp. 37–47, Jan. 2016, doi: 10.5121/acii.2016.3104.

D. Salvato, "Your Reality." [Online]. Available: https://genius.com/Dan-salvato-your-reality-lyrics

Mage, "The Words I Never Said." [Online]. Available: https://genius.com/Mage-the-words-i-never-said-in-d-b-lyrics

Zedd and Foxes, *Clarity*. Zedd, 2012. [Online]. Available: https://genius.com/Zedd-clarity-lyrics

D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: state of the art, current trends and challenges," *Multimedia Tools and Applications*, vol. 82, no. 3, pp. 3713–3744, Jul. 2022, doi: 10.1007/s11042-022-13428-4.

K. W. Church, "Word2Vec," *Natural Language Engineering*, vol. 23, no. 1, pp. 155–162, Dec. 2016, doi: 10.1017/s1351324916000334.

T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science, Business Media, 2001.

I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization." [Online]. Available: https:// arxiv.org/abs/1711.05101

I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The Long-Document Transformer." [Online]. Available: https://arxiv.org/abs/2004.05150

B. Muller, "BERT 101." [Online]. Available: https://huggingface.co/blog/bert-101

M. M. Lopez and J. Kalita, "Deep Learning applied to NLP." [Online]. Available: https://arxiv.org/ abs/1703.03091

J.-C. Wang, J. B. Smith, J. Chen, X. Song, and Y. Wang, "Supervised chorus detection for popular music using convolutional neural network and multi-task learning," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun. 2021, pp. 566–570.