ANLP Project - Song Parts(SP) Classification[1]

Shaked Amar, Ariel Barzilay, Tomer Yaacoby

Introduction and Motivation

Modern music services need to know where the chorus and where the verse starts.
Good segment labels help with lyric summarization, playlist skipping, thumbnailing, and even AI music generation.

Old rule-based tricks (for example, "the chorus is the line that repeats the most") break down when the writer plays with form, and they fail completely on pure-text data. We therefore test deep language models that learn the patterns directly from the words.

Related Work

Early work on text-only structure relied on surface repetition or rhyme rules.
More recent research trains neural models:
Fell et al. used a CNN to split lyrics into structural blocks and showed large gains over hand-tuned heuristics.
Wang et al. proposed a chorus-section detector for lyrics text (English + Japanese) and got great results, although they also explored the lyrics of the songs, their paper has some significant differences than the models we used. First of all, they tried to create a model that should work both for English and Japanese, but also, they divided the songs into one-line chunks, in contrast to our verse/chorus chunking. In addition, they kept the song full, whereas we kept only the first chorus to make the task "harder"
The DeepChorus uses the song's audio as input and combines multi-scale convolutions with self-attention for chorus detection.

To the best of our knowledge, **no prior study tests long-context transformers on *pure lyrics* for the specific "verse vs chorus" question**, nor compares "local-only" and "full-song" inputs side-by-side.

Problem Setting

Task 1 - Given the words of a single song part (SP), predict **"verse" or "chorus"**.

Task 2 - Does adding the rest of the song (shuffled) and/or using a model with a longer context window improve accuracy.

[1] Github repo for code: https://github.com/tomer4185/ANLP-Project.git

Solution - Our Approach

We began by building a simple classifier that decides whether a given song part(the “SP”) is a verse or a chorus. For training data we took the **mrYou/Lyrics_eng_dataset** from HuggingFace, which holds about 188k English-language songs together with the year they were released and their genre.

To see how much extra information helps, we trained our models in **two different settings**. In the first, the model sees only the words inside the target SP. In the second, the model also receives the rest of the song—the other verses and choruses—mixed together in random order so it cannot cheat by position alone.

We fine-tuned two transformer models under both settings. The first model is a standard BERT-style transformer that can read up to 512 tokens at once. The second model is a Longformer, which can process much longer inputs, up to 2 048 tokens, and therefore can capture a bigger slice of the song when context is included.

Each model was trained and evaluated **five times**, using a fresh set of hyper-parameters on every run. Afterward we kept the best-performing checkpoint for each of the four combinations (BERT/Longformer × with/without context) and compared their accuracy.

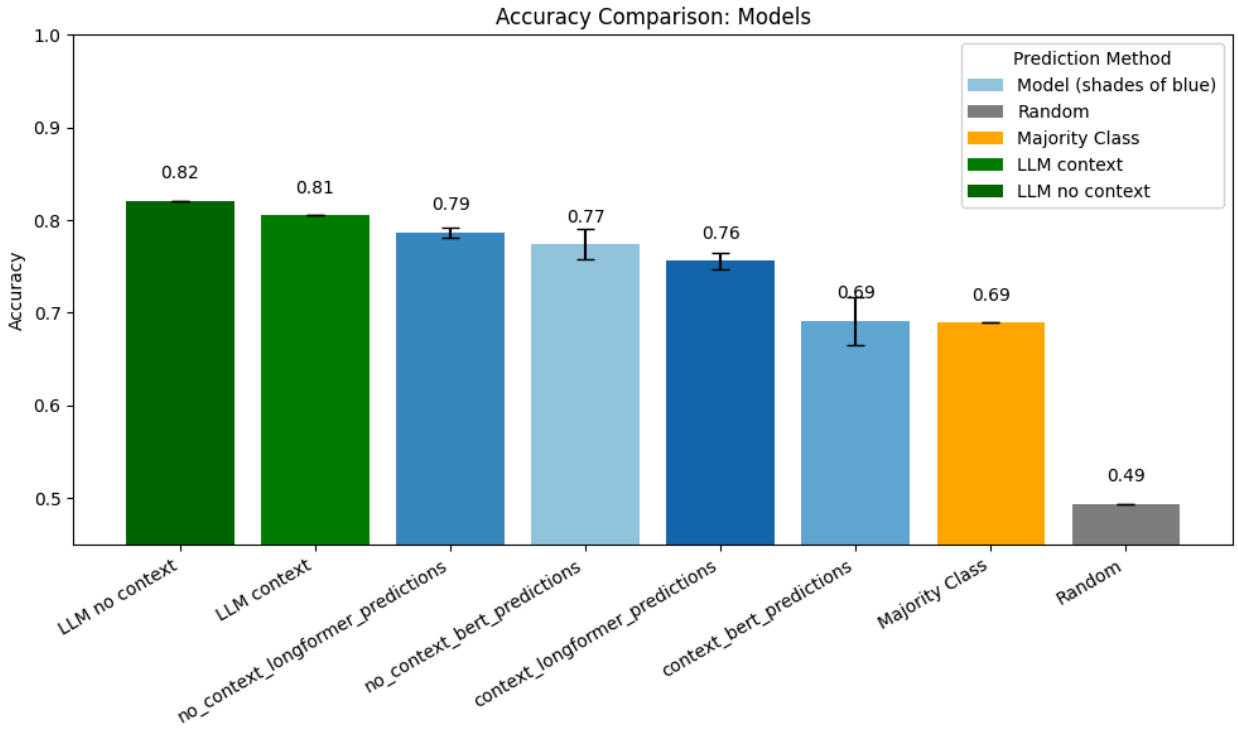
Finally, to get a rough “human-style” baseline, we asked Google’s generative AI model, **Gemini**, to label the same data—once with context and once without—and measured how well it matched the ground-truth answers.

Experiments

Both checkpoints come from HuggingFace Transformers. We add a single linear layer for **binary classification**. **Training details:**

- Optimizer: AdamW, LR 2×10-5
- Epochs: 100 (Longformer) / 200 (BERT).
- Five random seeds. We report the mean accuracy between 5 models for each setting.

Results



Our experiments show a clear performance hierarchy. Gemini, the large-language model we queried directly, achieves the top accuracy (0.82) when it sees only the target part, and practically the same score (0.81) when we also hand it the rest of the song. Among the fine-tuned transformers, Longformer without context follows closely at 0.79, while BERT without context trails at 0.77. Surprisingly, adding shuffled full-song context hurts both models: Longformer drops to 0.76 and BERT falls to 0.69, no better than the majority-class baseline. The random baseline sits near chance at 0.49. These results suggest that a longer input window helps, but supplying unstructured global context can confuse the model rather than help it, at least in our current setup.

Analysis

To understand where our system succeeds and where it fails, we carried out a focused error analysis. First, we asked whether certain “trigger words” push the model toward a chorus or a verse label. For the Longformer with context, we compared the words that appear in its false-positives and false-negatives (Fig. 4). The distributions look almost the same, so no single word - or small set of words - seems to dominate its decisions. Next, we checked if either class benefits from a strong advantage. Across all runs, the models score roughly the same on verses and on choruses, so there is no clear class-level bias.

We also inspected the confidence scores. Both transformers usually predict with very high certainty (95%-100%). Figure 5 shows that BERT's output probabilities remain steadier when we change the random seed, whereas Longformer's confidence spreads a bit wider. Finally, we plotted accuracy against the song's release year and its genre (Figs. 2 and 3). The curves are flat, which tells us the models are not favoring any particular era or musical style.

Summary

We built verse-vs-chorus classifiers on 2000 English songs using BERT (512 tokens) and Longformer (2048). Longformer without extra context scored 79% accuracy, close to Gemini's 82%, whereas naïve shuffling of full-song context lowered performance (Longformer 76%, BERT 69%). Errors show no trigger-word bias, high confidence, and stability across years and genres.
Future work - key directions include probing attention and ablations to uncover why shuffled context lowers accuracy, auditing whether BERT, Longformer, or Gemini have prior exposure to the lyrics to measure memorisation effects, extending the task to full segmentation (intro, bridge, outro), and scaling training to a larger, genre-balanced corpus spanning more release years.

Ethics and Limitations

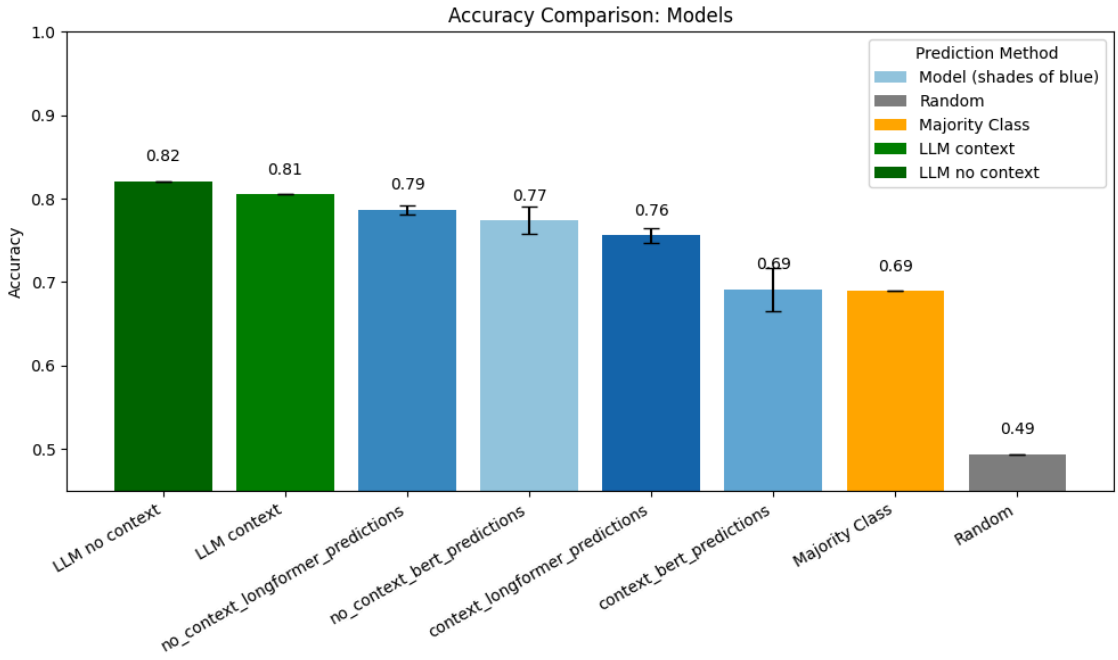
We use publicly available or properly licensed lyrics for research purposes, ensuring compliance with copyright law. Our model may reflect genre or cultural bias due to dataset composition and may struggle with unconventional song structures or long inputs. Chorus-versus-verse labeling can be subjective, introducing annotation noise. Additionally, transformer-based models have context-length limits and limited interpretability.

Figures

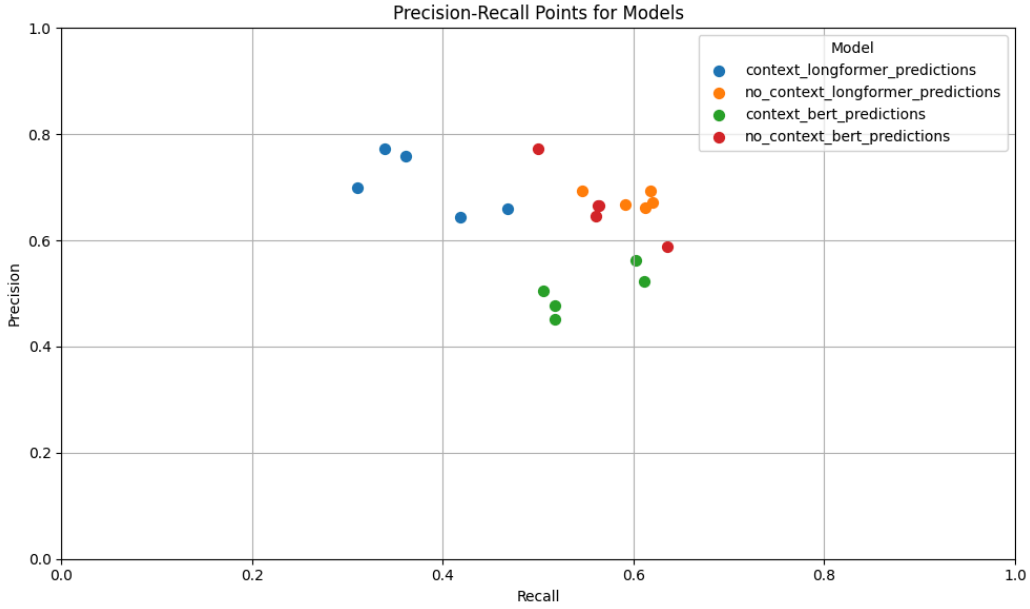
In the following figures, 1 / Positive is chorus, 0 / Negative is verse.
Also, where we present the ‘Bert finetunes with no-context’ model, the results for the other models are similar.

Fig.1 - Accuracy of the different models.
Fig.2 - Precision-Recal on each running configuration.
Fig.3 - Confusion matrix on Bert finetunes with no-context.
Fig.4 - Year’s analysis for the Bert finetunes with no-context model.
Fig.5 - Genre’s analysis for the Bert finetunes with no-context model.
Fig.6 - Word cloud for the FP/FN for the Bert finetunes with no-context model.
Fig.7 - Confidence histogram. The count is for the number of SPs the model confidence was lower than 0.9.

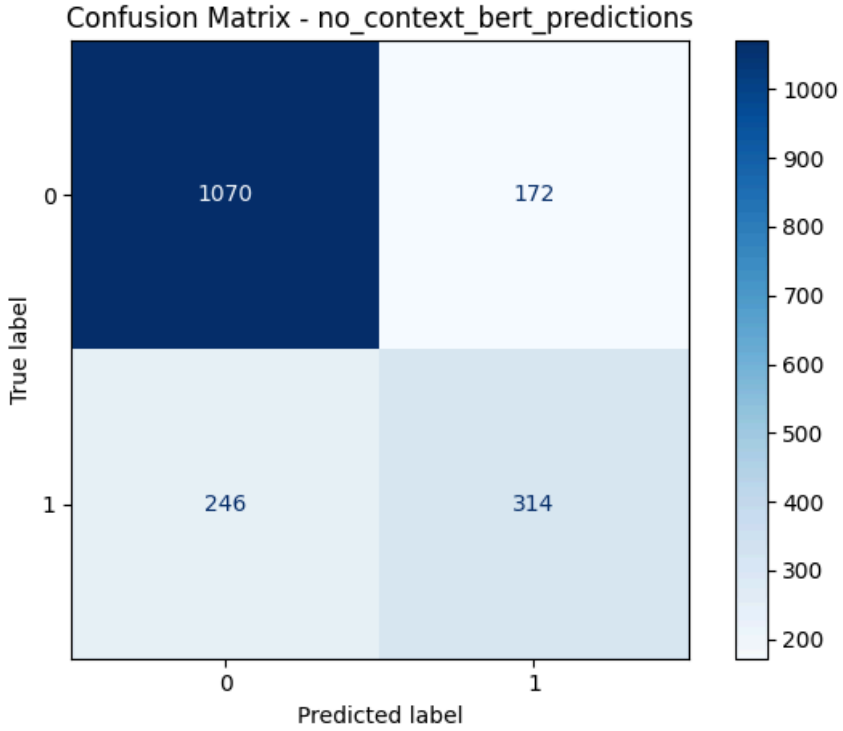
1.



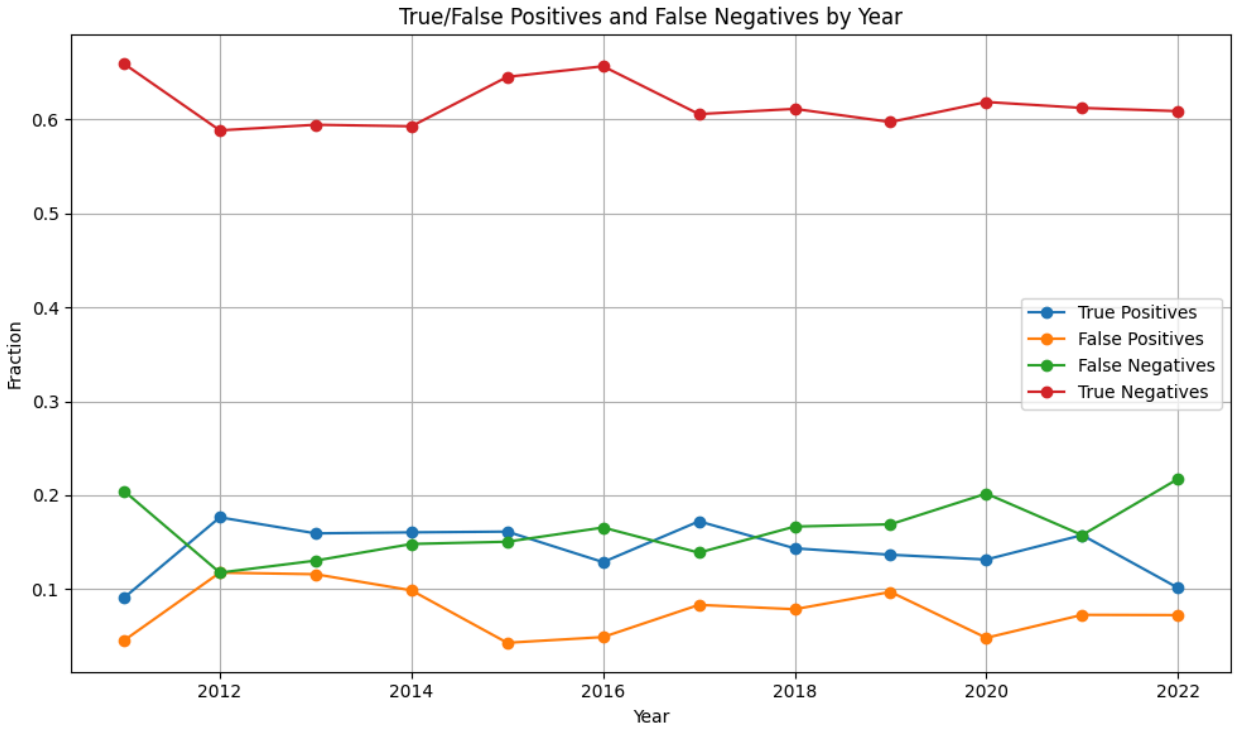
2.


Precision-Recall Points for Models
Model
context_longformer_predictions
no_context_longformer_predictions
context_bert_predictions
no_context_bert_predictions
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0

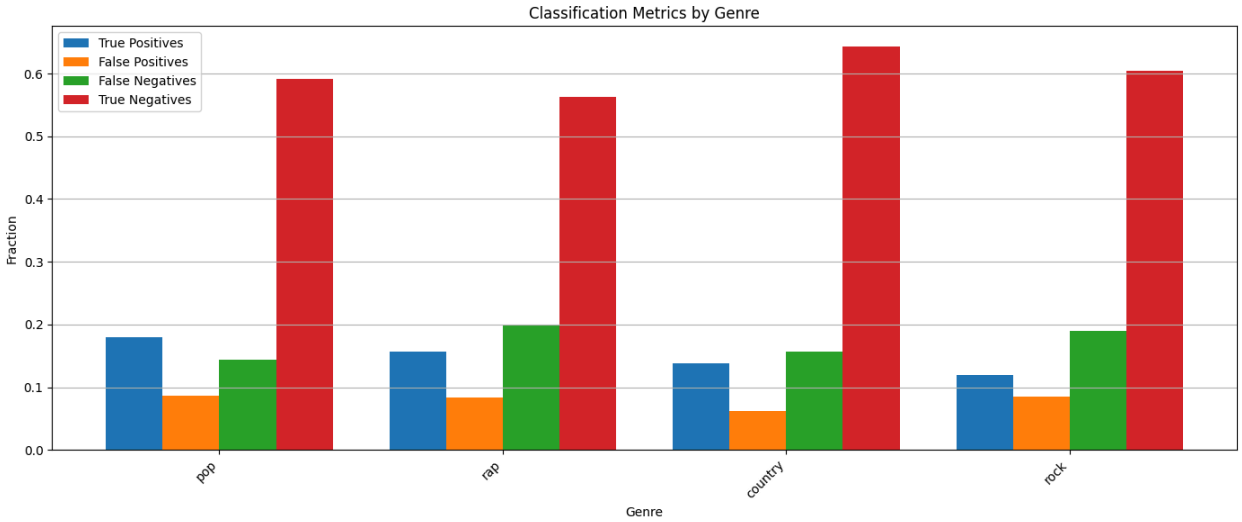
3.


Confusion Matrix - no_context_bert_predictions
1070
172
246
314
True label
Predicted label
0
1
1000
900
800
700
600
500
400
300
200

4.

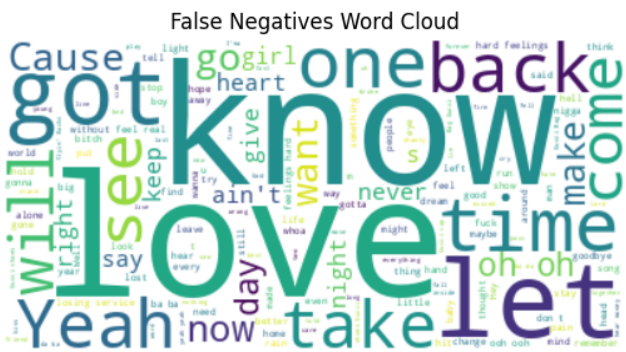

True/False Positives and False Negatives by Year
Fraction
Year
0.1
0.2
0.3
0.4
0.5
0.6
2012
2014
2016
2018
2020
2022
True Positives
False Positives
False Negatives
True Negatives

5.


Classification Metrics by Genre
True Positives
False Positives
False Negatives
True Negatives
Fraction
Genre
0.0
0.1
0.2
0.3
0.4
0.5
0.6
pop
rap
country
rock

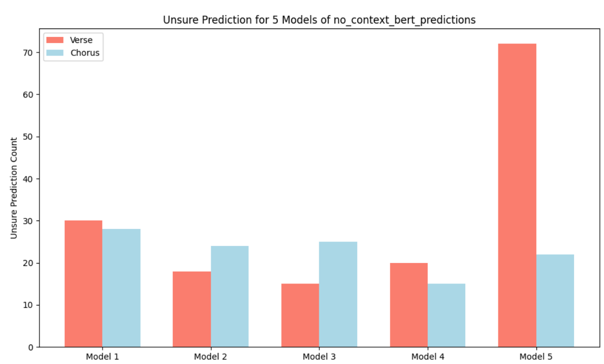
6.

False Positives Word Cloud
know
want
go
yeah
feel
now
Cause
come
take
see
time
love
Give
right
let
got
baby
never
will

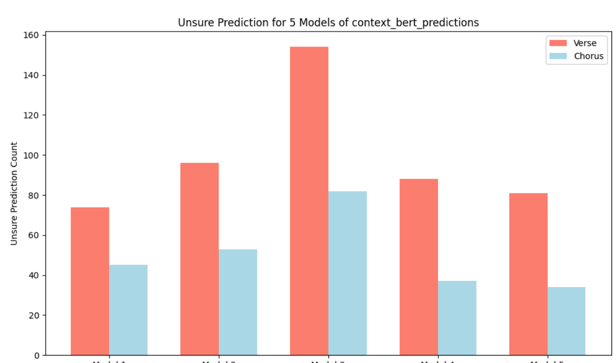


False Negatives Word Cloud
know
love
let
Yeah
take
time
got
back
one
Cause
come
see
will
make
go
girl
heart

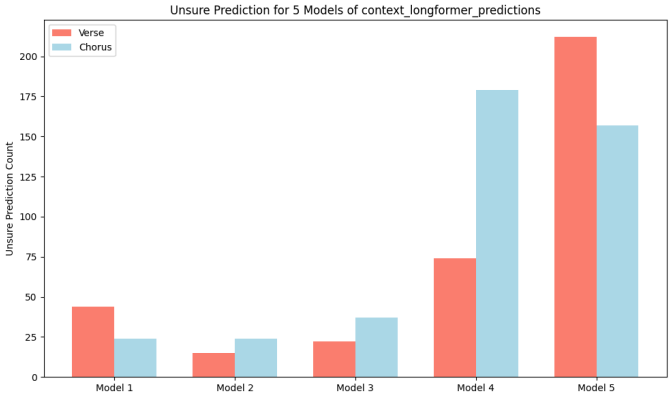
7.

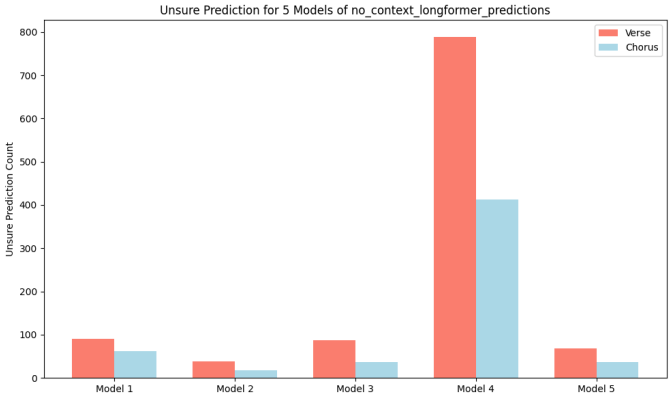


Unsure Prediction for 5 Models of no_context_bert_predictions
Verse
Chorus
Unsure Prediction Count
0
10
20
30
40
50
60
70
Model 1
Model 2
Model 3
Model 4
Model 5



Unsure Prediction for 5 Models of context_bert_predictions
Verse
Chorus
Unsure Prediction Count
0
20
40
60
80
100
120
140
160
Model 1
Model 2
Model 3
Model 4
Model 5





References

Fell M. et al., Lyrics Segmentation: Textual Macro-structure Detection using Convolutions (COLING 2018).

Wang Z. et al., A Chorus-Section Detection Method for Lyrics Text (ISMIR 2020).

He Q. et al., DeepChorus: Hybrid Convolution + Self-Attention for Chorus Detection (ICASSP 2022).