NLP Approaches to Summarizing Song Lyrics

Sahana Sankar
UC Berkeley
School of Information
sahanasankar@berkeley.edu

Jack Lucas Chang
UC Berkeley
School of Information
jacklucasc@berkeley.edu

Chloe McGlynn
UC Berkeley
School of Information
cmcglynn@berkeley.edu

Abstract

Recent advances in natural language processing have enabled the analysis of creative text forms such as novels, screenplays, and poetry–however, the task of summarizing song lyrics presents unique challenges due to their genre-specific elements, figurative language, and recurrent text structures. This research evaluates the effectiveness of current encoder-decoder transformer models (T5-small, PEGASUS, and BART) for lyric summarization, with T5-small serving as the primary model for all possible data combinations. Our results demonstrated that while enriching the training data with PoemSum annotations improved semantic similarity in the T5-small model, the fine-tuned BART and PEGASUS models ultimately outperformed T5-small in generating high-quality lyric summaries.

1 Introduction

Creative text summarization, including poetry, song lyrics, and other forms of artistic expression, poses unique challenges for natural language processing (NLP) models. Unlike prose, these text types often rely on figurative language, symbolic imagery, and non-linear structures, requiring models to interpret both surface-level and deeper semantic meanings. Song lyrics, in particular, stand out for their hybrid nature, blending literary conventions with musical context. Their repetitive sections, genre-specific stylistic elements, and evolving cultural slang further complicate the task of generating meaningful summaries. Existing summarization models, designed primarily for structured or literal text types such as news articles or academic papers, struggle to capture the rich linguistic and cultural dimensions of lyrics. This work seeks to address this gap by adapting transformer-based models for lyric-specific summarization tasks. Accurate summarization of song lyrics can improve music discovery, as they provide additional metadata that algorithms can use for personalized recommendations.

In this study, we curated a dataset of popular songs from the Billboard Hot 100, enriched with annotations from the Genius API, to train and evaluate transformer-based summarization models. By fine-tuning T5-small, PEGASUS, and BART, we explore how these models handle the nuanced elements of song lyrics. This work evaluates the performance of existing models and provides new insights into adapting NLP techniques to creative text forms.

2 Datasets

2.1 Song Lyrics and Genius Annotations

To train our summarization models, we curated a dataset of 3,187 songs from the Billboard Hot 100 and Billboard Top Artists charts, representing diverse musical genres and styles. Using the Genius API, we retrieved annotations that interpret both surface-level and deeper meanings of lyrics as target outputs for model training. However, these user-generated annotations presented limitations due to inconsistency – some focused on factual information like chart performance, while others offered lyrical interpretations ranging from surface-level observations to deeper analysis. This variation made it difficult for the models to consistently prioritize between factual content and thematic interpretation. Song lyrics were acquired through the Lyrics.ovh API, as Genius did not permit direct lyric extraction.

2.2 PoemSum Dataset

To enhance the models' ability to handle creative and figurative language, we incorporated the PoemSum dataset, which includes 3,011 poems annotated with summaries. This dataset provides examples of metaphorical and condensed language, complementing the linguistic and thematic features

found in song lyrics. The PoemSum dataset is particularly valuable for exposing models to literary elements that are essential for summarizing creative texts.

3 Past Work

In creative text summarization, the PoemSum dataset (Mahbub et al., 2023) demonstrated potential for capturing metaphorical meanings in poetry, but did not address lyric-specific challenges like repetitive structures. Saeed et al (2019). approached lyric repetition using GANs with discriminator feedback, though evaluation remained heuristic. Li et al. (2022) explored semantic understanding of Genius annotations through contextual information integration, while Fell et al. (2019) combined topic-based summarization with audio thumbnailing for lyrics analysis. Transformer architectures like T5, PEGASUS, and BART have set benchmarks in abstract summarization tasks. Our work evaluates these models for lyric-specific summarization, leveraging enriched datasets to capture both linguistic intricacies and cultural context.

While efforts have been made to use music videos to aid summarization effectiveness, most songs do not have accompanying music videos, and their production has been declining due to shifts in music consumption patterns. (Shao, Xa, Kankanhalli, 2004; Xu et al., 2005, Guardian, 2024).

4 Methods

4.1 Overview

Our research approach focused on three transformer architectures–T5-small, BART, and PEGASUS–each selected for their individual strengths in text-to-text generation tasks. We followed a systematic evaluation process, beginning with T5-small as our primary model to test multiple combinations of datasets, then applying the most successful data configuration to BART and PEGASUS for comparative analysis. We selected T5 as our primary model to test based on its demonstrated effectiveness with creative text interpretation in a Poetry Summarization (PoemSum) task study (Mahbub et al., 2023). Their implementation employed T5-base with token lengths of 512 and 256 for text and summary respectively, achieving strong performance (R1: 45.0%, R2: 25.9%, RL: 33.9%, BS: 85.9%), but we adapted their architecture to function within our computational limitations.

4.1.1 Data Processing

All models involved consistent data pre-processing steps and input formatting. Lyrics were formatted with the prompt, "Summarize lyrics and capture meaning: lyrics", while target outputs consisted of corresponding summaries derived from annotations. Data cleaning addressed NaN values and standardized string formatting to maintain consistency across the dataset. An 80/20 train-validation split was implemented for all experiments to balance model training and evaluation.

4.2 Model Architectures

4.2.1 T5-Small Architecture (Primary Model)

The T5-small architecture was selected based on its proven effectiveness with creative text interpretation (Mahbub et al., 2023). Key modifications included reducing the model size from T5-base to T5-small for computational efficiency, extending the maximum input token length to 1024 for longer lyrics, and adjusting training parameters such as a smaller batch size (4), increased gradient accumulation steps (4), a learning rate of 1e-4, early stopping with patience of 3, weight decay of 0.01, and enabling mixed precision training.

4.2.2 BART Architecture

We implemented this model using bart-large-cnn–a summarization model that was pre-trained on English text and fine-tuned on CNN Daily Mail articles–with specific optimizations for abstractive summarization (Lewis et al., 2019). The model retained its original encoder-decoder structure and was implemented using PyTorch Lightning. The input token length was set to 512, aligning with BART's pre-training datasets like CNN/Daily Mail (Liu Lapata, 2019) to ensure computational efficiency and compatibility, while the target token length was capped at 150 tokens to generate concise and abstractive outputs. Training adjustments included increasing the batch size from 4 in T5-small to 16 to leverage BART's larger capacity, stabilize gradient updates, and optimize GPU utilization through mixed precision training. The learning rate was reduced from 1e-4 in T5-small to 2e-5 to fine-tune pre-trained weights effectively, accommodating BART's larger parameter space. Additionally, AdamW was used as the optimizer with weight decay, and gradient accumulation was employed to manage memory constraints. For generation, configuration settings included top-k sampling (k=50),

a temperature of 0.7, no-repeat n-grams (size 3), and a length penalty of 1.0 to enhance output diversity and coherence.

4.2.3 PEGASUS Architecture

We implemented this model using `pegasus-cnn_dailymail`, a pegasus-large model trained on C4 and HugeNews datasets–but with modifications for lyrical content (Zhang et al., 2020). The input sequence length was set to 1024 tokens to allow for longer lyrics, while the maximum summary length was capped at 128 tokens to ensure concise and interpretable outputs. Due to computational constraints, the batch size was limited to 1, with gradient accumulation steps maintained at 4 to effectively simulate a larger batch size during training. The learning rate was reduced from 1e-4 in T5-small to 5e-5 to prevent large updates that could disrupt fine-tuning on PEGASUS's pre-trained weights, ensuring a gradual adaptation to the lyrical content. Weight decay was kept at 0.01 to regularize the model and prevent overfitting. For generation, we used a beam size of 4, maintained a length penalty of 1.0 to balance summary brevity and coherence, enforced a no-repeat n-gram size of 3 to improve diversity, and enabled early stopping to prevent redundant or overly lengthy outputs. These adjustments optimized PEGASUS for abstractive summarization tasks involving creative and nuanced texts like song lyrics.

4.3 Training Infrastructure

Our implementation used PyTorch Lightning to standardize training across all models. The training process incorporated mixed precision training and gradient accumulation to optimize GPU memory usage and manage larger effective batch sizes. We focused initial fine-tuning on the T5-small model, due to its efficiency in handling complex text-to-text transformations. We then extended successful approaches to BART to leverage its strengths in abstractive summarization. We implemented early stopping with configurable patience levels to prevent overfitting while ensuring model convergence. Data loading and batching processes were standardized through custom DataModule implementations to handle our varied data sources consistently. For larger models like BART and PEGASUS, we implemented automatic memory cleanup routines to maintain stable performance during extended training sessions.

4.4 Evaluation Metrics & Framework

For song summarization, metrics like content coverage, semantic similarity, ROUGE, and BERTScore are essential due to the unique challenges posed by lyrical texts. Lyrics often use creative language not used in everyday communication–including metaphors and thematic repetition–making it critical to preserve both the key elements and underlying meaning. Content coverage ensures essential phrases or motifs, such as repeated choruses or iconic lines, are retained, while semantic similarity measures the preservation of intent and emotional nuance, even when the wording changes. ROUGE (R-1, R-2, R-L) evaluates lexical overlap, ensuring foundational elements of the lyrics appear in the summary, particularly for any important lyrics. BERTScore leverages contextual embeddings to assess deeper relationships between words, capturing the nuance and figurative language often present in lyrics. Together, these metrics address the dual challenge of summarizing the literal content and interpreting the thematic and emotional depth of lyrics.

5 Experiments

5.1 Overview

The primary objective of our experiments is to assess the T5-small transformer model's ability to generate meaningful song lyric summaries across different input configurations. We aimed to determine the value of enriching training data with user-generated annotations and poetry text by systematically varying data combinations.

We selected T5-small for its versatility in text-to-text generation and efficiency with semantic transformations. After identifying the best-performing dataset configuration, we extended testing to BART and PEGASUS to validate findings and compare model-specific strengths. This stepwise approach ensures robust evaluation of both data influence and model adaptability for lyric summarization.

5.2 Baseline

For our baseline approach, we trained the T5-small transformer using only raw song lyrics in a self-supervised configuration. This baseline model resulted in very high ROUGE scores (>98.0%) and consistency scores (99.0%), though these metrics revealed a conservative summarization strategy–the model was essentially copying lyrics and directly reproducing them in the output. The moderate

content coverage (41.2%) and semantic similarity (40.4%) suggested a balanced approach–selective in content inclusion while maintaining meaningful vocabulary overlap. However, the extremely high consistency across generations indicated potential overfitting and insufficient generation diversity, revealing the need for model improvements through adjusted generation parameters and enhanced abstractive summarization techniques to increase output diversity. This baseline established the need for additional training data and more advanced model architectures to achieve more interpretive, diverse summaries while maintaining accuracy.

5.3 (T5) Lyrics and Genius

Building on our baseline approach, we enriched the training data by incorporating annotated content from Genius users alongside raw lyrics. We aimed to improve the model's interpretive capabilities through exposure to human-written explanations and thematic analysis.

The T5-small model configuration remained consistent with our baseline hyperparameters, but with input data now consisting of paired lyrics and their corresponding Genius annotations. The model demonstrated significant behavioral changes compared to the lyrics-only baseline.Content coverage decreased (41.2% to 12.1%) and semantic similarity reduced markedly (40.4% to 9.5%), indicating more abstractive summarization rather than text reproduction. While the lower ROUGE scores (21.3%, 4.3%, 13.5%) might suggest degraded performance, the relatively stable BERTScore (81.0% vs 85.8%) indicates preserved semantic understanding. However, the model exhibited several key limitations: generation of generic templated phrases, factual inaccuracies, and inconsistent handling of empty or sparse annotations. While these results appear inferior to the baseline model's performance, they represent progress toward the more challenging goal of interpretation over simple summarization, though improvements in factual accuracy and specific content generation are needed.

5.4 (T5) Lyrics and PoemSum

This experimental configuration explored whether exposure to poetry and corresponding summaries could enhance the model's ability to handle figurative language in lyrics. Through incorporating the PoemSum dataset alongside song lyrics, we aimed to leverage the similarities between poetry and lyrical forms of creative expression.

The training setup was consistent with our previous configurations but required careful data preparation to handle the multi-domain inputs. We combined PoemSum's training data–while preserving the original train and validation splits–with song lyrics, and implemented distinct input prompts for lyrics and poems. Most hyperparameters were held constant, but we reduced worker count from 4 to 2 to optimize processing efficiency. The integration of PoemSum data produced several notable effects, including higher content coverage, increased semantic similarity, and lower ROUGE scores than the Lyrics-Only and Lyrics+Genius approaches. These results suggest that this model approach captures more comprehensive meaning, stronger engagement with figurative and metaphorical content, and shows a shift toward more interpretive generation. The results suggest that while poetic data enhances the model's ability to handle creative language, careful attention must be paid to data preparation and training configuration to fully leverage the cross-domain benefits. This approach demonstrates potential for improving figurative language understanding, though at the cost of lower lexical overlap with reference summaries.

5.5 (T5) Lyrics, PoemSum, Genius

For this experiment, we trained a T5-small transformer using a combined dataset of song lyrics, PoemSum data, and Genius annotations. This approach aimed to improve the model's ability to generate nuanced and interpretive summaries by leveraging a combination of these diverse data sources. Inputs were formatted using task-specific prompts to distinguish lyrics, poems, and annotations, with training conducted at a learning rate of 1e-5, gradient clipping at 1.0, and a batch size of 4 (with 2-step gradient accumulation). Despite our comprehensive training approach, evaluation metrics did not indicate much progress. Content coverage was low (12.9%), semantic similarity was poor (10.6%), and ROUGE scores reflected limited overlap with reference annotations. A BERTScore of 83.3% suggested moderate semantic alignment to the original text. However, the outputs often lacked coherence and relevance, exposing difficulties in balancing interpretive and thematic insights from the diverse data sources. While Model 4 showed slight improvements in BERTScore over Models 2 (Lyrics + Genius) and 3 (Lyrics + PoemSum), it performed worse in content coverage and semantic similarity.

Table 1: Model Performance Comparison Across Different Data Configurations

Data Sources	Model	BERTScore	R1	R2	RL	Content Coverage	Semantic Similarity
Lyrics Only	T5	85.8%	99.1%	98.8%	99.1%	41.2%	40.4%
Lyrics + Genius	T5	81.0%	21.3%	4.3%	13.5%	12.1%	9.5%
Lyrics + PoemSum	T5	78.7%	13.8%	1.8%	8.3%	63.8%	61.4%
Lyrics + PoemSum + Genius	T5	83.3%	21.5%	4.7%	14.1%	12.9%	10.6%
Lyrics + PoemSum + Genius	Pegasus	82.6%	19.2%	4.1%	12.4%	20.0%	17.1%
Lyrics + PoemSum + Genius	BART	83.6%	21.7%	5.2%	14.1%	6.4%	5.1%

Key limitations included handling the integration of three distinct data sources—lyrics, poems, and annotations—with varying structures and purposes. The model often generated summaries that were generic, irrelevant, or inconsistent with the input content. Based on these results, we determined that fine-tuning the model may address these specific weaknesses by tailoring the training process to emphasize better content coverage and semantic understanding.

5.6 (BART) Lyrics, PoemSum, Genius

For this model, we trained a BART-base transformer model on a combined dataset of song lyrics, PoemSum poetic data, and Genius annotations to evaluate its ability to generate thematic, abstractive summaries. The model used a batch size of 16, a maximum sequence length of 512, mixed precision training, and gradient clipping. Despite consistent preprocessing and proper data handling, the model demonstrated low content coverage (6.4%) and semantic similarity (5.1%), indicating it struggled to capture key thematic elements. ROUGE scores (1: 21.7%, 2: 5.2%, L: 14.1%) showed moderate overlap with reference summaries, while a high BERTScore (83.6%) indicated good semantic alignment. However, BART frequently produced template-like outputs focused on metadata, such as release dates or album details, rather than interpreting thematic content, likely influenced by its pre-training on structured document summarization. Compared to T5, BART achieved slightly higher ROUGE scores but failed to capture nuanced interpretive aspects, which is important for creative summarization tasks.

5.7 (PEGASUS) Lyrics, PoemSum, Genius

We used the `google/pegasus.cnn.dailymail` model for PEGASUS because it has been fine-tuned on the CNN/DailyMail dataset, a well-known benchmark dataset for abstractive summarization. This fine-tuning enables the model to generate new summaries by paraphrasing the input text rather than copying sentences verbatim. Due to resource constraints, we used a highly conservative approach like a minimal batch size, and reduced evaluation and logging frequency. The model showed a high content coverage (20%) and semantic similarity (17.1%), which means that the model captured the essential ideas or meaning from the songs and summarized the content appropriately. The lower ROGUE (1: 19.2%, 2: 4.1%, L: 12.4%) and BERTScore (82.6%) means that the summary used different wording compared to the original summary. Compared to the previous models, PEGASUS was more effective at abstracting the input data, resulting in lower ROUGE and BERTScore. However, this is actually preferred in this context, as we aim for the model to generate new content rather than simply repeat the lyrics. Refer to Appendix Section C for an example of a single song lyric generation.

5.8 Key Findings

The PEGASUS model fine-tuned on PoemSum, song lyrics, and Genius annotations achieved the highest semantic similarity (17.1%) compared to models trained on the same dataset and strong content coverage (20%), making it the most effective at capturing nuanced interpretations. The inclusion of PoemSum data was the most impactful, significantly enhancing the models' ability to handle figurative language and improve abstraction.

6 Fine-Tuning Model Approaches

We fine-tuned the models trained on a combination of poetry and Genius annotations to adapt them more effectively to our main goal of song lyric

Table 2: Model Performance Comparison Across Fine Tuned Models

Model	Batch Size	Max Length	Learning Rate	Coverage Metrics		ROUGE Scores			BERT Score
				Content	Semantic	R-1	R-2	R-L	
T5-small	2	512	1.0e-5	10.4%	8.2%	21.5%	5.0%	14.1%	84.1%
T5-small	4	256	3.0e-5	9.2%	7.4%	22.2%	5.5%	14.5%	84.0%
T5-small	2	512	5.0e-5	9.2%	7.2%	22.3%	5.5%	14.6%	83.9%
T5-small	4	512	5.0e-5	8.9%	6.9%	21.6%	5.3%	14.2%	83.6%
BART - Data Aug.	16	512	2.0e-5	8.3%	6.0%	22.3%	4.8%	14.2%	84.1%
BART - Hyper. Tune	8	256	5.0e-5	8.3%	6.2%	22.4%	5.2%	14.6%	84.5%

summarization.

6.1 (T5) Fine Tuning

The fine-tuned T5-small model introduced improvements to better address the challenges of song lyric summarization. While the original model used a batch size of 8 for computational efficiency, the fine-tuned version reduced it to 2 or 4 for more precise weight updates, improving its ability to capture lyrical nuances. Lower learning rates were used in fine-tuning to prevent overfitting and enhance performance on abstract, figurative texts. The results for the fine-tuned T5-small model reveal nuanced trade-offs across configurations. A batch size of 2, a sequence length of 512, and a learning rate of 1e-5 resulted in the highest average semantic similarity (8.2%) and BERTScore (84.1%), reflecting the model's ability to better interpret and capture abstract language in song lyrics. Refer to Appendix Section A for an example of a single song lyric generation for this model. However, configurations with larger batch sizes (e.g., 4) and shorter sequence lengths (256) demonstrated marginal improvements in ROUGE scores (e.g., ROUGE-1: 22.2%) but at the cost of slightly reduced semantic richness.

6.2 (BART) Model Fine Tuning

The fine-tuned BART model introduced important improvements to address summarization performance. Data augmentation, including paraphrasing via Google Translate (back-translation) and synonym replacement was applied to diversify the training data, introducing linguistic variations to improve generalization while preserving the original meaning of the lyrics. This approach showed limited improvement over the baseline, suggesting that augmented data alone was insufficient for capturing the nuances of song lyrics. In contrast, hyperparameter tuning achieved the best overall performance by optimizing the learning rate (5e-5), batch size (8), and sequence length (256 tokens). Refer to Appendix Section B for an example of a single song lyric generation for this model.

6.3 Fine Tuning Analysis

We fine-tuned the T5 models and BART models to see if there were any improvements in ROUGE scores. Unfortunately, we were not able to fine-tune Pegasus due to computing limitations. However, based on the comparison between the fine-tuned BART and T5-small models and PEGASUS, PEGASUS remains the most ideal model for summarizing song lyrics. Although BART with hyperparameter tuning achieves slightly higher ROUGE (1: 22.4%, L: 14.6%) and BERTScore (84.5%) compared to PEGASUS (ROUGE: 1: 19.2%, L: 12.4%, BERTScore: 82.6%), it falls significantly behind in semantic similarity (PEGASUS: 17.1% vs. BART: 6.2%) and content coverage (PEGASUS: 20% vs. BART: 8.3%).

7 Conclusion

The experiments demonstrate that incorporating diverse datasets like PoemSum and Genius annotations improves lyric summarization. Among tested models, BART excelled in semantic alignment while PEGASUS better captured nuanced themes and figurative language. While efficient, T5-small fell short of achieving the same interpretive richness. Future work will focus on implementing topic modeling to uncover key lyrical themes and sentiment analysis to capture emotional tone. Further parameter tuning may improve coherence and consistency in generated summaries. These refinements aim to produce summaries that capture both literal content and creative essence of lyrics, advancing NLP applications for creative texts. These efforts will advance NLP applications for creative texts, bridging the gap between technical precision and artistic understanding.

References

Michael Fell, Elena Cabrio, Fabien Gandon, and Alain Giboin. 2019. Song lyrics summarization inspired by audio thumbnailing. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 328–337, Varna, Bulgaria. INCOMA Ltd.

Xian-Sheng HUA, Lie LU, and Hong-Jiang ZHANG. 2004. Automatic music video generation based on temporal pattern analysis. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, MULTIMEDIA '04, page 472–475, New York, NY, USA. Association for Computing Machinery.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Jen-Chun Lin, Wen-Li Wei, James Yang, Hsin-Min Wang, and Hong-Yuan Mark Liao. 2017. Automatic music video generation based on simultaneous soundtrack recommendation and video editing. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 519–527, New York, NY, USA. Association for Computing Machinery.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders.

Ridwan Mahbub, Ifrad Khan, Samiha Anuva, Md Shihab Shahriar, Md Tahmid Rahman Laskar, and Sabbir Ahmed. 2023. Unveiling the essence of poetry: Introducing a comprehensive dataset and benchmark for poem summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14878–14886, Singapore. Association for Computational Linguistics.

Asir Saeed, Suzana Ilić, and Eva Zangerle. 2019. Creative gans for generating poems, lyrics, and metaphors.

The Guardian. 2024. The decline of music videos in modern music consumption. *The Guardian*. Retrieved January 2024.

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

Appendix: Model Comparisons

A. T5-Fine Tuned Example

The generated annotation for "God You Are Going Far" touches on broad themes but fails to meaningfully engage with the song's lyrics and their deeper meanings. While it mentions the themes of a personal journey and emotions, these are generic and surface-level. It provides no discussion of figurative language, cultural references, or layered meanings, leaving the emotional and narrative richness of the song untouched.

Table 3: Comparison of T5-Fine Tuned Model's Annotation with Genius Annotation.

Genius Annotation	**Fine Tuned T5 Generated Annotation**
"The Way You Are" was the first Tears for Fears song that songwriter Roland Orzabal wrote after the release of the band's hugely successful debut album. As the band was at that time very hot (i.e., marketable) in the wake of their success, their record company was eager to release any new material they had, and so the band set about polishing and releasing "The Way You Are" as a single. The song only reached #24 in the charts, however, a ranking Orzabal "reflected how good it was" (namely, not first-tier). In hindsight, Tears for Fears and their collaborators consider "The Way You Are" a second-rate song, and regret the amount of work they put into it and its release as a single. The track was later included on the 2014 Super Deluxe Edition Box Set, although it was written and released before that second album of the band's.	"God You Are Going Far" is a song about a person's journey and how they feel about themselves. The song was released as a tribute to a friend of the late singer-songwriter. It was also featured on the album.

B. BART Model Example

The generated annotation inaccurately attributes the song to Lady Gaga and includes placeholders, indicating poor data processing. It misinterprets the lyrics by focusing on rejecting materialism, while the song emphasizes personal longing and disappointment. However, unlike the T5 model, it is able to correctly identify themes of emotional fulfillment and the importance of relationships, aligning with broader holiday sentiments. Despite this, it fails to capture the specific tone and narrative of the lyrics.

Table 4: **Comparison of Fine-tuned BART Model's Annotation with Genius Annotation**

Genius Annotation	Fine Tuned BART Generated Annotation
The Voice coaches Ariana Grande and Kelly Clarkson sing of not wanting any of the Christmas festivities because they didn't get what they wanted, and they beg for Santa to give them what they wanted because otherwise they don't see a point in Christmas.	"What I Want for Christmas" is a holiday song written and produced by Lady Gaga. The song was released as the fourth single from her sixth studio album. The song is about giving up material things in order to spend the holiday season with your loved ones. In this song, Gaga talks about what she wants for Christmas and how she sent a letter to her lover asking for what she wanted, but he wasn't able to hear it.

C. Pegasus Model Example

Although the generated annotation oversimplifies the song themes, it recognizes the setting of California and the beach as central elements of the song's imagery. It fails to reference Katy Perry's intention behind the song as a homage to California culture. The generated annotation also has grammatical errors and repetition; however, it is overall able to extract key themes in the song.

Table 5: **Comparison of Pegasus Model's Annotation with Genius Annotation.**

Genius Annotation	Pegasus Generated Annotation
"California Gurls (Demo)" is the demo version of Katy Perry's song. The unmastered demo version leaked online days prior to the official version's release, which was released May 7, 2011. The beats in the demo are a bit louder, and a few new adlibs are found in the outro.	"California Girls" is the second single off of Katy Perry's third studio album. The song is about a group of women who are obsessed with California and the beach. Katy describes the song as a love song to the state. The lyrics are about a girl who is obsessed with the beach and the people there.