John Guo - jg843
Shahrukh Showkath - ss4266

ORIE 5741 Final Project Report

This project explores the development of a customer churn prediction transformer model for music streaming services. A publicly available music streaming dataset was used to analyze behavior and identify factors influencing customer retention. A transformer model was used as it may offer unique and valuable insights for these streaming services, potentially improving their ability to retain customers. The report will detail the methodology employed for data exploration, preprocessing, and model development.

Customer churn is the act of customers canceling their subscriptions or downgrading to a cheaper or free tier plan. Customer retention is of utmost importance for a business, especially under the viewpoint that recurring customers are far more profitable than new customers. The acquisition costs of attracting new customers involve marketing campaigns, promotions, and sales efforts, which generally require significantly more financial resources. Contrastingly, retention costs of existing customers tend to be less expensive, which would involve loyalty programs, exclusive offers, or improved customer service. Furthermore, existing customers are more likely to make repeat purchases, upgrade their subscriptions, act as a stable and predictable revenue stream, recommend the service to others, and serve as data points that allow the music service to offer personalized recommendations and improve the overall customer experience. Prioritizing customer retention translates to a more stable and profitable business in the long term. Thus, it is imperative to avoid the event of subscriber churn that disrupts the otherwise recurring revenue stream to the business and potentially negatively impacts the business's future profits. The project aims to predict whether a customer is at risk of churning or not. By analyzing a user's streaming session and the relevant information of the session information, it is possible to predict if a user will soon conduct actions conducive to them churning, such as clicking cancel subscription, downgrade, and others. Analyzing what type of content or features a user engages with during a session and how long they spend consuming can reveal preferences and overall engagement. If this is correctly predicted, interventive measures can be taken. These targeted inventions could include personalized recommendations, promotional offers, or proactive customer support. This combination of session analysis, churn prediction, and targeted inventions would allow a streaming service to proactively address user concerns and retain customers.

Under this notion, the idea of using a transformer-based learning model arose. Traditional machine learning models often struggle with complex sequential data like user listening history[2]. Thus, transformers provide a unique benefit over other learning models for three main reasons [1]. Firstly, they should model complex user behavior better: user listening information holds key insights into each user's behavior in that they may exhibit patterns over long periods. Transformer architecture allows them to analyze both recent and past actions, building a comprehensive snapshot of the user. Transformers can capture these long-term dependencies and model this activity. Additionally, their multi-head attention mechanism lets them focus on different aspects of a sequence simultaneously. This could be useful in attending to all the factors of a user's listening session, such as song choice, session length, location, and more, all within the same sequence. By

attending to these diverse factors concurrently, transformers create a richer understanding of user behavior within a session. Finally, with increasingly larger datasets, a transformer-based model is inherently more flexible in its architecture to incorporate more features in the future. Streaming services constantly collect more user data, including new features and interaction points. This scalability can result in the model capturing more intricate relationships when a business can collect more data over time. Thus, opting for a transformer appears to be a worthwhile solution to the problem.

With this motivation, we conducted exploratory data analysis (EDA) on the dataset. We used the Sparkify provided by Udacity, a music streaming dataset where users can use a free or paid subscription of the service. Each user's online interactions with the service are recorded along with their timestamp. We used the Sparkify Mini Dataset, which contains music choices from 225 customers over a 60 day period, which has 286,500 rows. The features of the dataset consist of user information (user_id, name, gender, location, etc.,), service information (level, registration), user interactions information (method, page, status, etc.,), and song/artist information (song, artist, length). We began EDA by inspecting each feature, the format they were stored in, and what they actually represent. We then generated some visualizations to uncover any data imbalances potentially present. We noticed that there was a negligible difference between male and female users (Fig. 1)

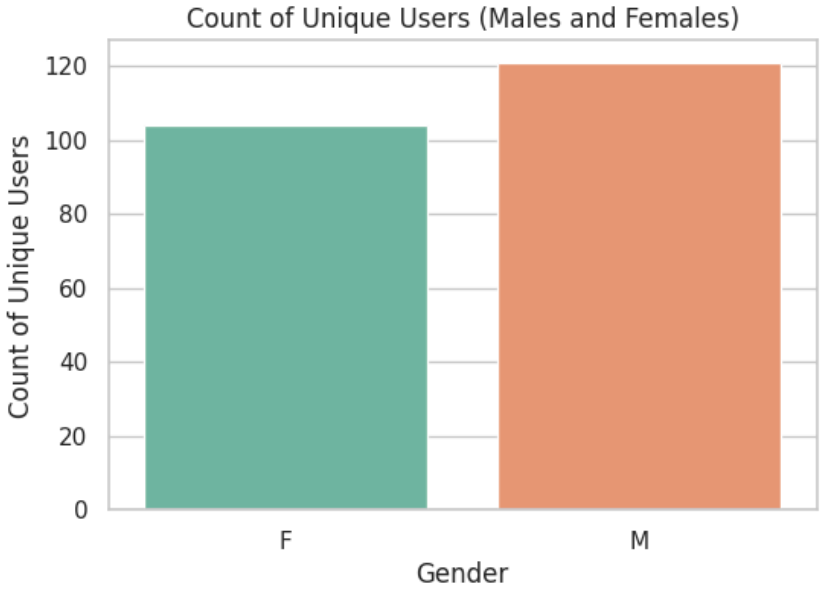

Figure 1.

Additionally, there was a negligible difference in subscription rates between the genders, with 82.6% of males being subscribed versus 74% of females (Fig. 2).

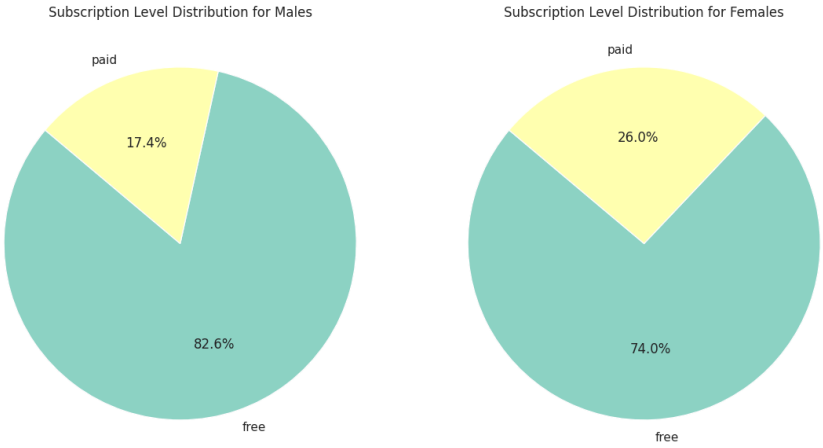

Figure 2.

Inspecting the distribution of pages resulted in a troubling realization that the vast majority of the pages were *NextSong* (Fig. 3). This could potentially result in the model just predicting *NextSong* every time, and was thus acknowledged as a feature to regulate. Further inspections were done on location, distribution of user engagement by subscription level, and song lengths by subscription level, with the only key insight being that subscribed users tend to listen to more songs in one session as opposed to free users (Fig. 4). This could mean that subscribed users, by listening to more songs, may be more reluctant to leave or downgrade the service.

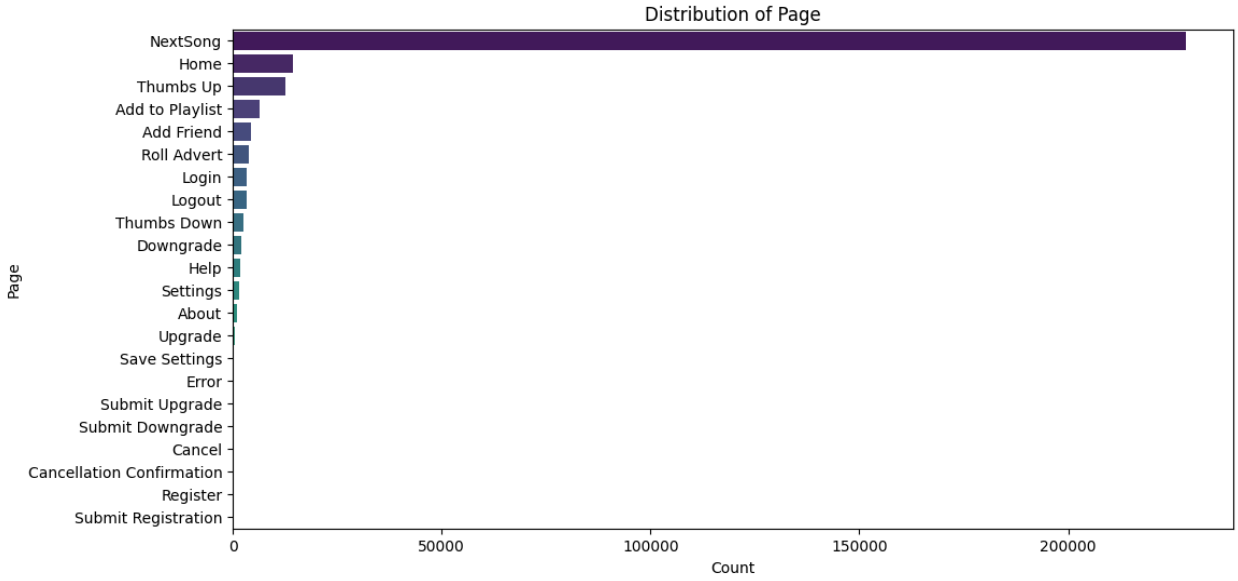

Figure 3.

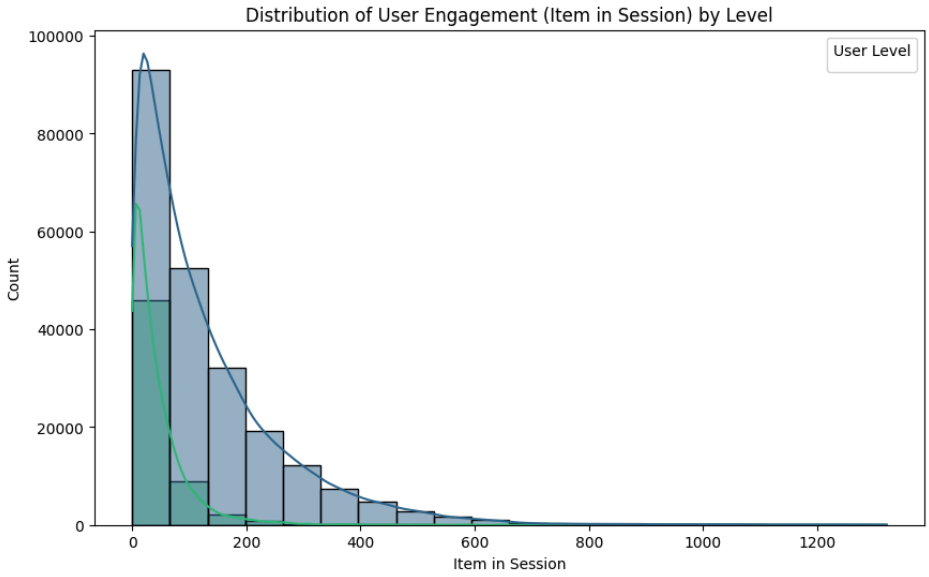

Figure 4.

With this data exploration, we began preprocessing the data for the transformer. We conducted feature transformations on the dataset by tokenizing non-numeric features and labeling churn targets. This was done manually by labeling pages that possibly indicate churn, including *free, paid, cancel, downgrade,* etc. After identifying churn-related pages, each user session was assigned a 1 if they visited a page indicating positive churn action (e.g., *cancel*), a -1 if they visited a page indicating a negative churn action (e.g. *paid*), or 0 if they visited a neutral page with no churn implication. We noticed that some of the data was null or had missing values. This was handled by removing any row containing missing values, and eliminating them ensures the model trains on complete and clean information. With this cleaned data table, we began sequence embedding. Each user session represented as a sequence of tokens, was transformed into a dense vector to allow the transformer to efficiently process the sequential data. Positional encoding was also injected into these embeddings. Since transformers inherently lack the understanding of order within a sequence, these embeddings tell the model the relative position of where the webpage visit token was. This information is imperative for the model to learn how to predict which web page a user visits. Additionally, certain indices were masked to prevent the model from seeing future information. This prevents the model from "cheating" by viewing future actions and forcing the model to predict churn solely based on the information available up to a certain point in the session. We then employed a train/test split on this preprocessed data to ensure an unbiased, objective performance assessment of the model training. By following these data preprocessing steps, the transformer model is equipped to analyze the user session sequences.

With the data completely preprocessed, we moved on to model training. The core function of architecture is the transformer decoder, specifically designed for sequence-to-sequence tasks. The decoder interprets the encoded representations and generates an output. The decoder is built upon a series of stacked transformer layers that analyze the user sequences, and each layer refines the previous layer's knowledge. Each transformer layer has a multi-head self-attention mechanism to let

it attend to different aspects of the user's activity at each point in the sequence. While self-attention captures the relationships within a sequence, it does not capture explicitly the non-linear relationships. This is all done through a position-wise feed-forward network that injects nonlinearity into the model, allowing it to capture more complex relationships between user actions. To further account for the imbalanced data, we opted for weighted cross-entropy loss. This technique overcomes the imbalanced churned data by assigning higher weights to the minority class during loss calculation. This regularization effect penalizes the misclassifications of churned users more heavily to discourage the model from overfitting to the majority class. The transformer model, with its stacked layers, multi-head self-attention mechanisms, and position-wise feed-forward network, combined with the weighted cross-entropy loss, is ready to analyze the user session sequences.

While the transformer architecture held promise in churn prediction, its computational demands quickly became apparent during training. Training the model revealed significant bottlenecks in processing power and data availability. Training took significant computational resources as well as time, thus making it infeasible to train the model to convergence and heavily limiting our ability to wait for its performance to stabilize. However, we opted to train the model for a feasible amount of time to observe the learning trend. This approach still provided insights into the model's potential, even if it did not represent its full capacity. The model proved to show a general increase in accuracy over time, for both the training and testing datasets (Fig. 5 & Fig. 6)

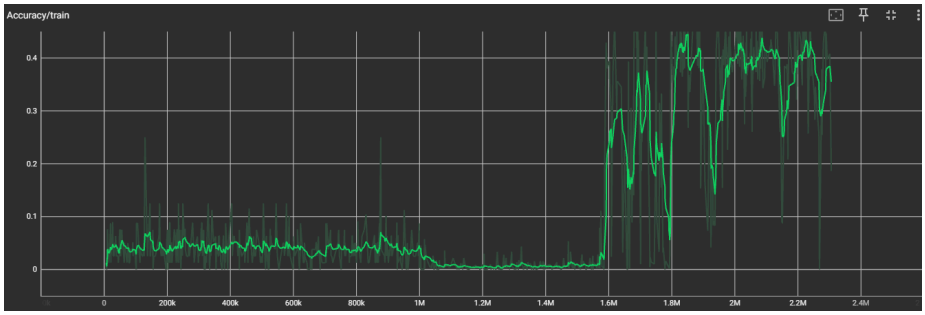

Figure 5.

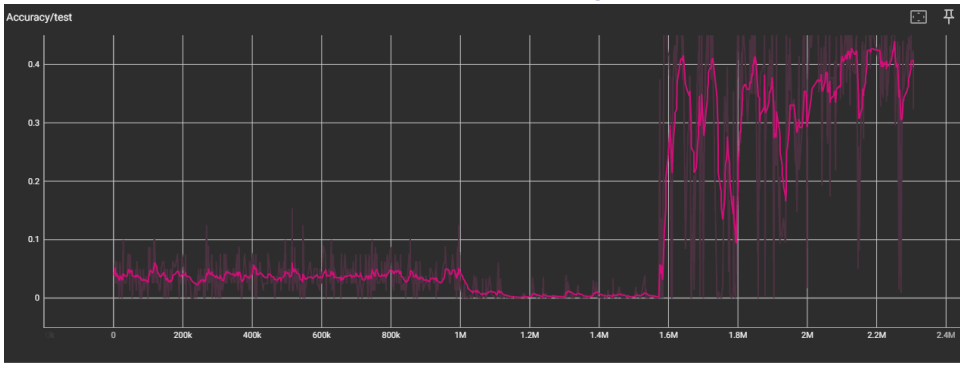

Figure 6.

When observing the loss functions, we can notice a similar trend. The loss functions decrease generally for both the training and testing sets (Fig. 7 & 8).

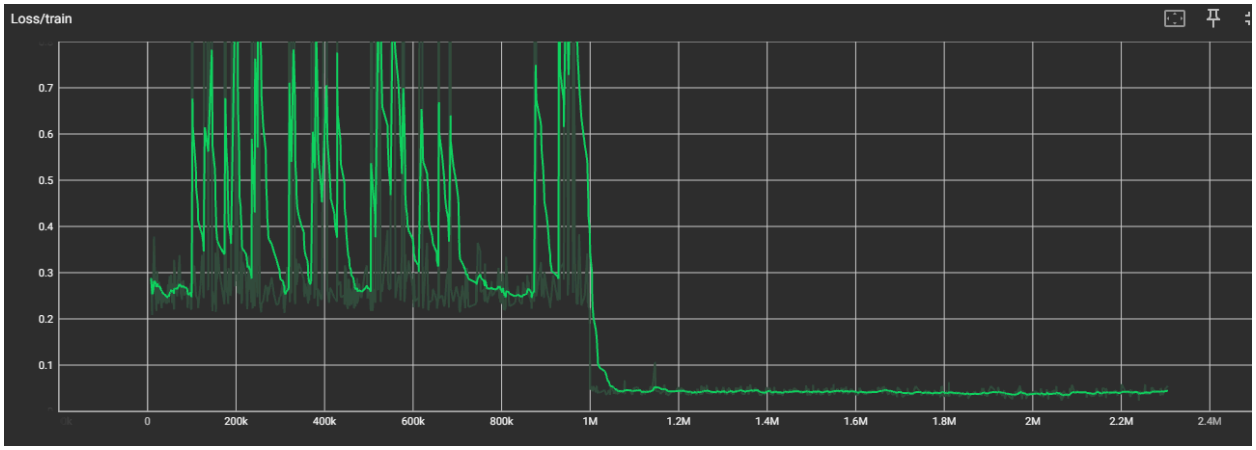

Figure 7.

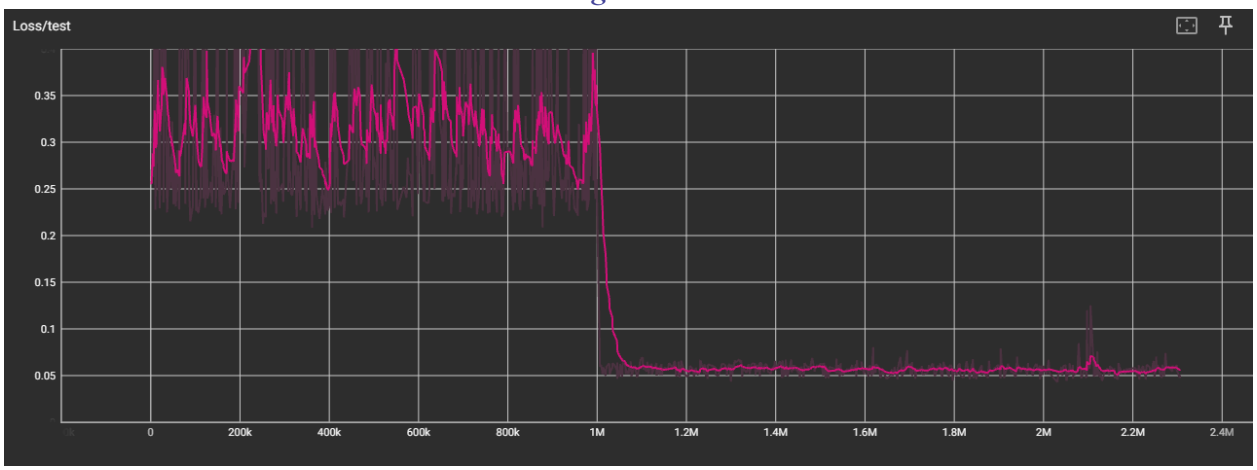

Figure 8.

The analysis of both the training and testing data reveals promising trends about the model's ability to learn and predict a user's churn behavior. An upward trend in training accuracy signifies the model's ability to correctly classify the user session on the training data. The decreasing trend of loss also highlights the model's ability to align its prediction with the true labels. One of the reasons for the sudden change visible in the graphs can be attributed to a change in batch size when we realized we were able to increase the batch size without significantly hindering training speed or exceeding the amount of memory at our disposal. The sparsity of churn signals needed to be addressed as well. The data holds user sequences represented by webpage visits and service interactions. However, the churn-indicating pages appear infrequently within most user sequences, meaning that the model needs to learn from relatively rare churn pages compared to the vast amount of inconsequential data present in the sequences. This sparsity of churn signals creates a weak learning signal for the transformer, and it struggles to effectively learn the patterns. This could have potentially been avoided with data augmentation and creating variations within the existing sessions to improve the learning signal. Overall, the observed trends are promising signs that the transformer model is learning and has the potential to predict user churn behavior. However, a comprehensive evaluation with a full dataset and convergence is needed to confirm the

model's generalizability and effectiveness, and we are not confident in the results at this stage. While transformers do excel at sequence modeling, they may not always be the most efficient choice due to their computational and data demands. This may make it suboptimal for production, and we would not be willing to deploy this assuming there is a similar lack of computational resources, where an alternative or classical model may be more efficient, even if it is less accurate overall.

Due to their intricate architecture, transformers have proven to be computationally expensive in this regard. Future work can include comparing transformers to simpler or classical models in terms of performance and efficiency. These models, such as Recurrent Neural Networks or Long Short-Term Memory networks, are proven to be successful in sequence prediction tasks. They are generally less computationally demanding than transformers, but may struggle to capture the intricacies that transformers can. This could lead to a decrease in prediction accuracy, but may be worthwhile with their computational efficiency. Further work can be done in trying to optimize transformer training for efficiency. Techniques that could be explored include model pruning, where redundant or trivial connections within the transformer are removed. This reduces overall model complexity without sacrificing too much accuracy. Knowledge distillation, involving training a smaller model to mimic the predictions of a larger transformer, could also achieve comparable accuracy [3]. Carefully comparing the performances of transformers with simpler models and implementation optimization techniques like pruning and knowledge destination could be the work needed for the real-world deployment of transformers in music streaming services.

Ethical considerations like fairness and potential for misuse are equally important for real-world application to technical aspects. In a churn prediction model, fairness is a crucial criterion, as streaming service data may contain biases. For example, users who listen to a specific artist may be targeted for retention efforts based on the model's predictions, potentially leading to unfair treatment of certain user groups. The model itself could introduce bias in training. If the training data disproportionately represents a certain demographic of users, the model may prioritize keeping those users while overlooking others, even when accounting for regularization techniques. Data cleaning and balancing is crucial in identifying and addressing biases, including techniques like oversampling of underrepresented groups or applying de-biasing algorithms to reduce the effect of biases. Explainable models and transparency can also rectify potential biases in the model. It is also important to acknowledge the potential for misuse. The model relies on user data, potentially raising privacy concerns. User behavior could be personal, and ensuring secure storage and responsible use is important. Obtaining user consent and anonymizing data can help mitigate these issues. The model could also psychologically manipulate user behavior. If used unethically, the service may provide users predicted to churn with aggressive promotions, creating an unfair advantage for certain users. Practicing transparency and user control can help minimize all of these potentials for weaponization. Transparency regarding how churn prediction is used and what data is being collected is imperative, and users should have the control of whether their data gets collected or not. They should additionally be able to opt out of any model that leverages their information. Prioritizing airness, responsible use, and addressing potential biases can result in

churn prediction models being invaluable tools for music streaming services to keep users in a responsible manner.

In essence, this project explored the use of a transformer-based model for customer churn prediction for a music streaming service. Using the Sparkify dataset, we preprocessed the data to account for imbalances and modified the dataset for the transformer to read. While the model portrayed promising signs of learning, the limitations of computational resources, data availability, and time prevented a full evaluation of its effectiveness. Further investigation with larger datasets and more powerful computing resources is necessary to assess their full effectiveness. Careful consideration should also be given to potential biases and the ethics of deploying such models to ensure they are used fairly and responsibly. Addressing these concerns could show the valuable effect of transformers for music streaming services to improve user retention and achieve long-term success.

References

1. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems, 2017 https://arxiv.org/pdf/1706.03762

2. Cho et al., "On the Properties of Neural Machine Translation: Encoder-Decoder Architectures with Regard to Language Typology," arXiv preprint arXiv:1409.1114, 2014 https://arxiv.org/abs/1409.1259

3. Hinton et al., "Distilling a teacher model into a smaller student," arXiv preprint arXiv:1503.00711, 2015 https://arxiv.org/pdf/1503.02531