X, formerly known as Twitter, is a popular platform where millions of people go to leave opinions and reviews of places, topics, and companies. This includes sharing thoughts about the airline industry, where people have had both positive and negative experiences, both of which get shared on X. The job of sentiment analysis is to predict what sort of experience a user had based on the tweet that they shared about their experience. I am specifically interested in how well the model will perform with regards to the slang and abbreviations that X users use. The slang used on X varies greatly, with some users using very radical slang that could be perceived as negative when in reality it is positive, such as “sick”, or abbreviations like “u” instead of “you”. Considering this, as well as the fact that language evolves rapidly over time, I am wondering how well the Hugging Face sentiment analysis model will be able to accurately predict the sort of experience those users have. Additionally, I think it would be interesting to see which airline has the most negative reviews.

To do this, I wanted to utilize the transformers library to be able to use the Hugging Face sentiment analysis model to predict the sentiment of tweets. I found a dataset from 2015 (before Twitter was X) of tweets from users expressing their feelings about different airlines. These tweets were categorized as “positive”, “negative”, and “neutral”. Since the sentiment analysis model from Hugging Face is trained on binary classes, I would have to filter the tweets to only have the “positive” and “negative” ones. I am planning to run the model 3 times: once on the dataset with all of the tweets as is, another time singling out tweets that I feel have a lot of slang/abbreviations, and another using tweets that I feel have little to no slang/abbreviations. Then I wanted to compare the accuracy of the model and see how or if it changed based on how much slang was used. The primary deliverables for this project would be a Jupyter Notebook that contains all of the data loading/preprocessing, sentiment analysis, and model evaluation. To evaluate how well the model performs I would calculate the model’s accuracy and do a confusion matrix to help visualize the performance of the model, as well.