

## Term Project Proposal

### PROBLEM

Videos from streaming platforms are downloaded in chunks and fluctuations in network performance can result in rebuffering, essentially lowering the quality of user experience. To minimize the interruptions caused by loading delays, our project aims to predict the time needed to download the next video chunk. By accurately predicting these download times, streaming platforms such as YouTube and Netflix can adjust video quality or pre-buffer data to reduce loading interruptions, while improving user experience.

### MOTIVATION

*What is the motivation for the problem? What is the exact specification of the problem?*

The motivation for this problem stems from the increasing demand for high-quality video streaming without interruptions. Popular streaming platforms rely on dynamic adaptation of video quality to match network conditions and avoid buffering. The problem involves the use of machine learning to predict the download time of video chunks based on the data provided from previous video chunks.

### APPROACHING THE PROBLEM

*What are some existing approaches to this problem? What are some existing datasets that you can work on? What is the novelty of your project? New problem? New approach? New dataset?*

The *Puffer* project from Stanford University, is an existing approach to this problem, which leveraged machine learning to improve video-streaming algorithms. We plan to reference the *Puffer* project and work with their dataset, which provides network traffic data, including packet-level metadata, transfer rates, and video quality metrics. This dataset is well-suited for training machine learning models to predict video chunk download times. Although we plan on using the *Puffer* project as our guide for our research, we aim to focus specifically on predicting video chunk download times rather than optimizing overall streaming performance. Our project will also explore how varying network conditions impact the accuracy of these predictions, using different

machine learning models to refine and improve prediction accuracy beyond what has been demonstrated in the *Puffer* project.

## **IMPLEMENTATION**

*How are you going to implement your approach, what are your metrics of success, and what's your evaluation plan?*

We will use the *Puffer* project's network packet data and metadata as input to train a machine learning model to predict video chunk download times. Specifically, we will use packet-level data and metadata from past video chunks, such as packet size, timestamps, and transfer rates, to train a machine learning model to predict the download time for future chunks. The model will be implemented using Python with libraries like TensorFlow or Scikit-learn.

### Metrics of Success:

- To measure our accuracy of the model's download time predictions, we will use Mean Absolute Error (MAE).
- We will also consider evaluating the effectiveness of the predictions and how it reduces the video rebuffering. This will be done by observing fewer interruptions in the video playback.

### Evaluation Plan:

- We will divide the dataset into training and testing sets. The model's performance will be evaluated on the test set, comparing predicted vs. actual chunk download times.
- Then, we plan to use cross-validation to ensure model robustness. Additionally, we will analyze the model's performance under stable and unstable network conditions to ensure its accuracy in real-world scenarios.
- Finally, we will evaluate our results with the metrics and benchmarks used in the *Puffer* project.