Homework-7: TensorFlow and Kafka

Deadline: November 21st, 11:59PM ET.

In this homework, you will practice TensorFlow and Apache Kafka.

Your homework submission should be on GitHub. Use the following GitHub classroom to access the assignment: https://classroom.github.com/a/o76lRJSO

You should submit the URL for your GitHub repository on Canvas. Grading penalty will be applied if otherwise.

Cite any external sources you use. External sources shouldn't exceed more than 30% of the final solution.

Q1. (50%) You will use TensorFlow to build neural networks to predict not just whether there is an attack, but also the type of attack for the NSL-KDD problem. In other words, this is a multi-class classification problem with 5 possible categories (normal, DOS, R2L, U2R, probing)

Q1.1 (10%) Prepare the data and convert them to appropriate Tensor formats needed for TensorFlow. You may use KDDTrain+.txt as the training dataset, 50% of KDDTest+.txt as the validation dataset, and the remaining 50% as the test dataset.

Q1.2 (20%) Build a Neural Network using tf.keras, and conduct training and validation. Select the appropriate loss function and choose at least one metric. Set the appropriate number of epochs. After training, evaluate your trained model on the test data.

In your submission, include a screenshot of the output of the fit function and the evaluate function.

Q1.3 (20%) Display the results in TensorBoard. In your submission, include screenshots of the loss and the metrics for both the training and the validation run.

Q2. (50%) YouTube Video Analysis Based on Comment Popularity:

Enhance the YouTube demonstration from the lecture by developing a feature that identifies the most popular video out of up to five videos. This will be determined by calculating which video has the highest total number of likes on its comments.

Design your application to prompt the user to input up to five YouTube video IDs.

Ensure your application aggregates the total number of likes from the comments on each video and displays the title of the video that has accumulated the most likes on its comments.

You may assume that calling the YouTube API once is sufficient for retrieving all the comments that you need to consider for your analysis.

Ensure that all communications with the YouTube API are confined to the Producer script to avoid redundant API calls from other areas of the code.

You can print the output on a terminal shell or a webpage.

Your application must use Apache Kafka (or Confluent Kafka).

Hint: You may find the YouTube API reference useful:
https://developers.google.com/youtube/v3/docs/

For Q2., submit the following:

1. (20%) All the code files you developed/used in a working state.

2. (15%) Screenshot of the “Messages” tab of your Kafka topic with sample messages from your Kafka topic. (15%)

Hint: open the “Messages” tab before you start the Producer to view the messages properly.

3. (15%) Screenshot of the output of the consumer.

If your output is not printed in a Jupyter notebook, take a screenshot of the output and make sure your name is written in a Text Editor in the screenshot (Don’t photoshop your name).