Deliverable 1

Project: Quick Draw

Team members: Anson Ng, Abigail Wolfensohn

1. Choice of dataset

For our project, we are going to be using the dataset provided by Google that the company uses for its own Quick Draw game, seen here. It features 50 million drawings – in the form of timestamped vectors as well as some metadata – across 345 categories of items. We chose this dataset because its large volume of consistent and accessible data will be highly useful for training; additionally, because it has already been preprocessed it will be simpler to use to train our model.

2. Methodology

a. Data Preprocessing:

   i. As was mentioned previously, the data has already been preprocessed by Google. However, a model that needs to be trained using 345 different categories may be beyond the scope of this course, so we figured it would be more reasonable to choose only 20 or so categories for our model (exact ones TBD). The fact that the data comes in the form of timestamped vectors (meaning it contains information about the types and directions of strokes used to draw the images) is very useful in a drawing context, since while drawing, the strokes a person makes and their order/direction can reveal a lot about the object being illustrated. Training the model on data like this will allow for real-time predictions as the user themselves is drawing.

b. Machine learning model:
    i. Our goal is to train a machine learning model to be able to predict what a person is drawing from a list of 20 prompts, based on the qualities of their real-time brushstrokes.
    ii. Our exact model and implementation has yet to be finalized, as it requires further research and discussion with our TPM. However, based on what we know right now, we think a recurrent neural network (RNN) would be the most effective model to choose: this is a type of artificial neural network that is capable of understanding sequential, data, meaning it has the potential to be very effective at parsing our timestamped brustroke data. However, a con of this method is that it might be complex to train, although we do hope that reducing the number of prompts from 345 to 20 will make our model much less computationally demanding and simpler to refine.

c. Evaluation metric:

More research and consultation needs to be done in order for us to finalize evaluation metrics. For now, our main options are as follows:
    i. Precision
    ii. Recall
    iii. AUC
    iv. F1

3. Application

We have two main ideas in mind for how we can integrate our model. In both cases, the user will be given a prompt, and will have a short amount of time to draw a doodle of the item they were told; based on their drawing, the model will offer a prediction of which item it was. As for the precise methods, we have two main ideas for implementations in mind:

1. Webcam: the user holds up something to the camera and gestures out their drawing)
2. Digital canvas: the user draws onscreen.

We were also thinking of making the model provide its output prediction in the form of an emoji.