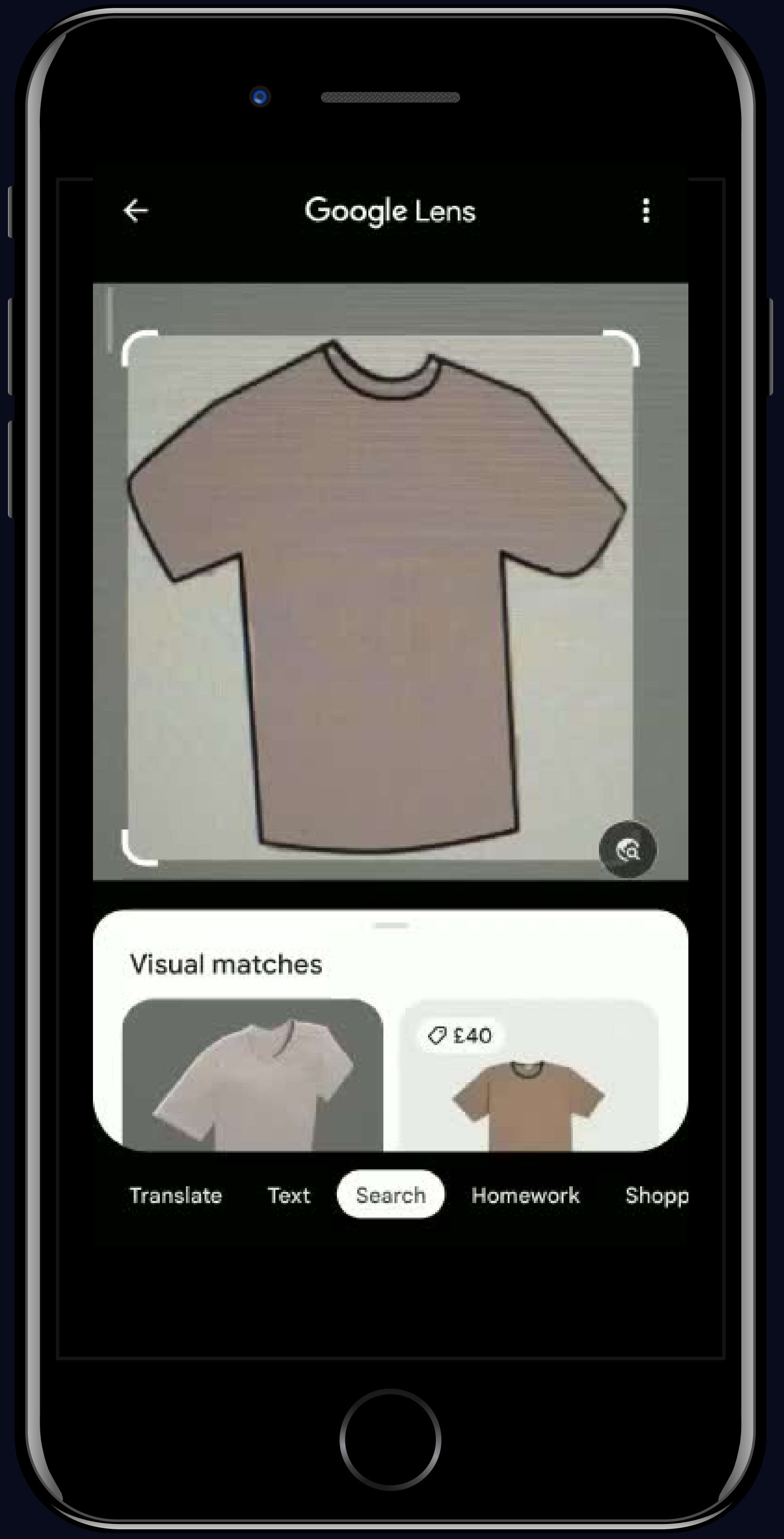
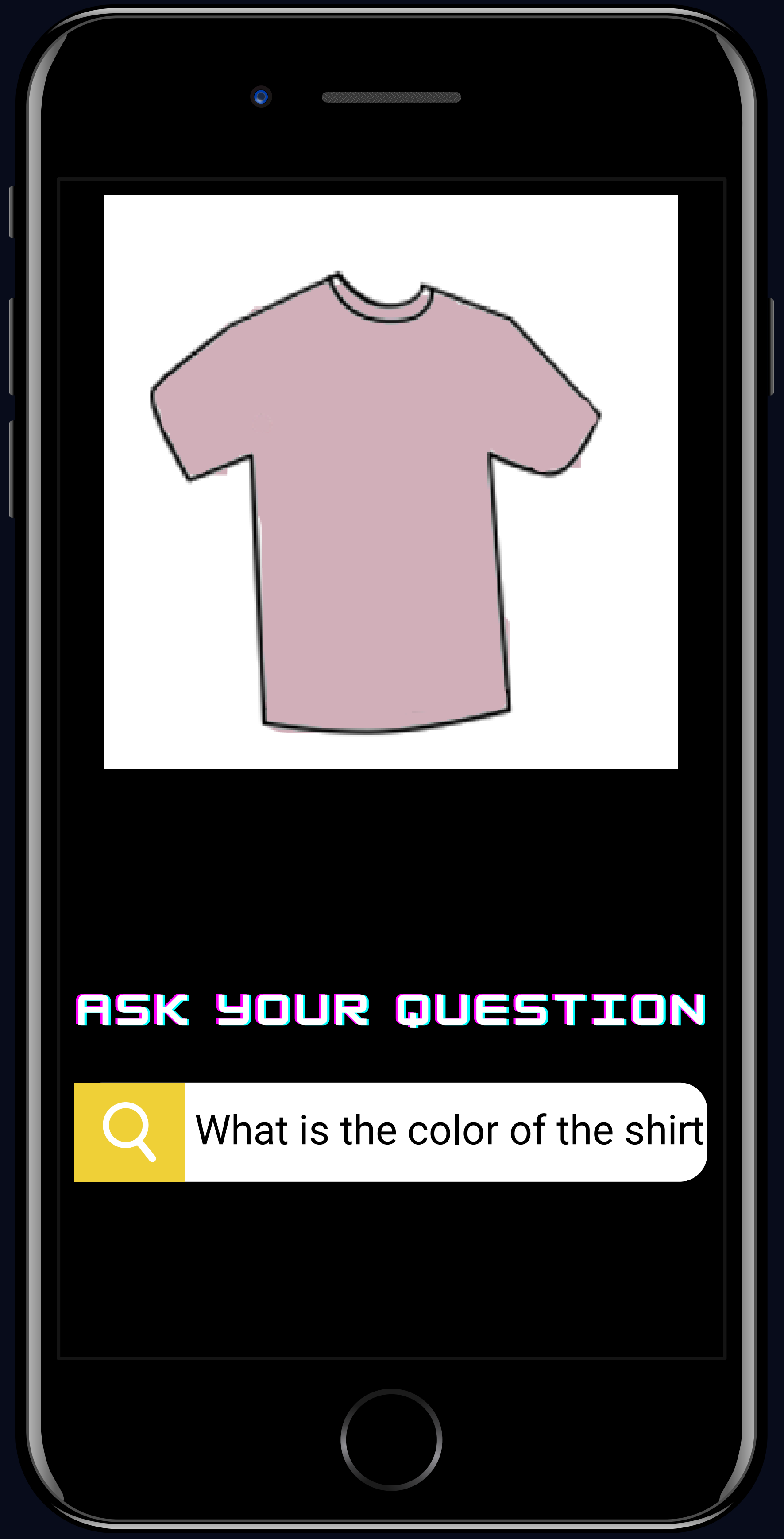
# VISUAL QUESTION ANSWERING

BARATH NIRANJAN S A

# PROBLEM STATEMENT

The project is an Android application aimed to help the visually impaired by giving them the ability to take a picture, ask questions about it and the application will provide them with the answers using machine learning techniques and tools.


Google Lens
Visual matches
£40
Translate
Text
Search
Homework
Shopp

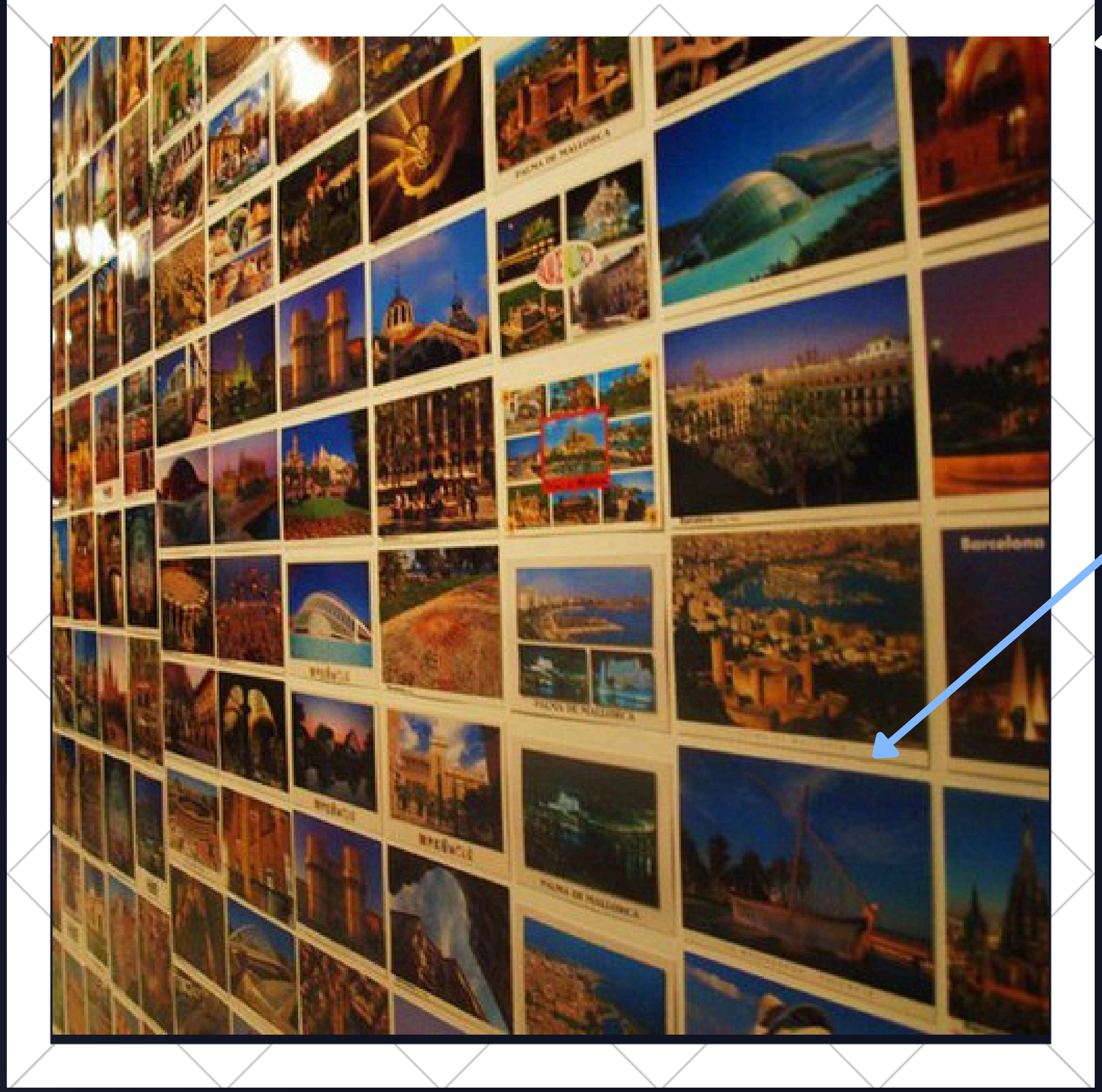


ASK YOUR QUESTION

What is the color of the shirt

STAMPED CANDY

# DATASETS USED



0.25M Images

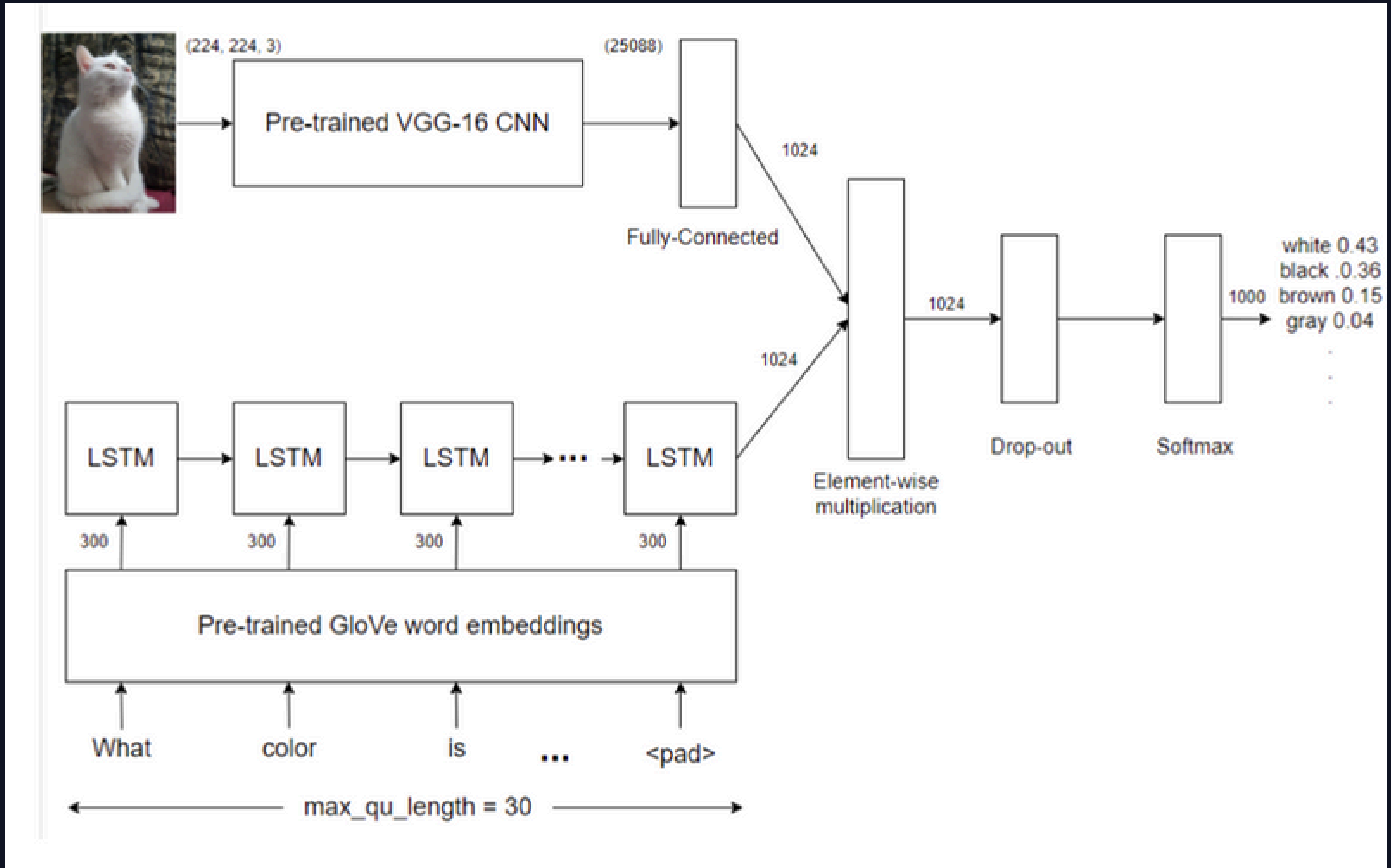
What is in the image?

0.76M Questions

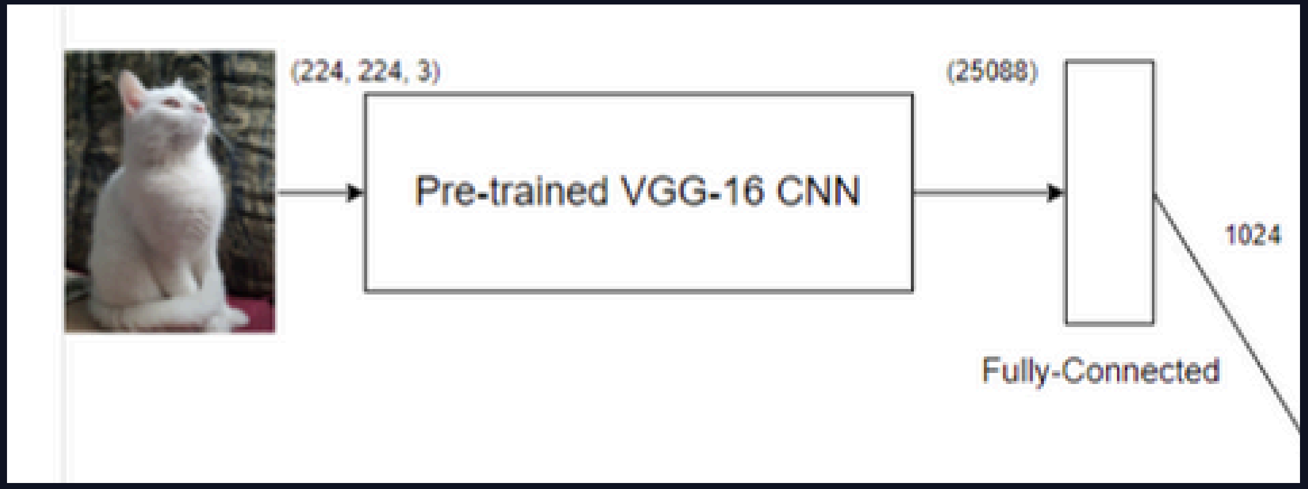
Boat

10M Answers

# MODEL ARCHITECTURE



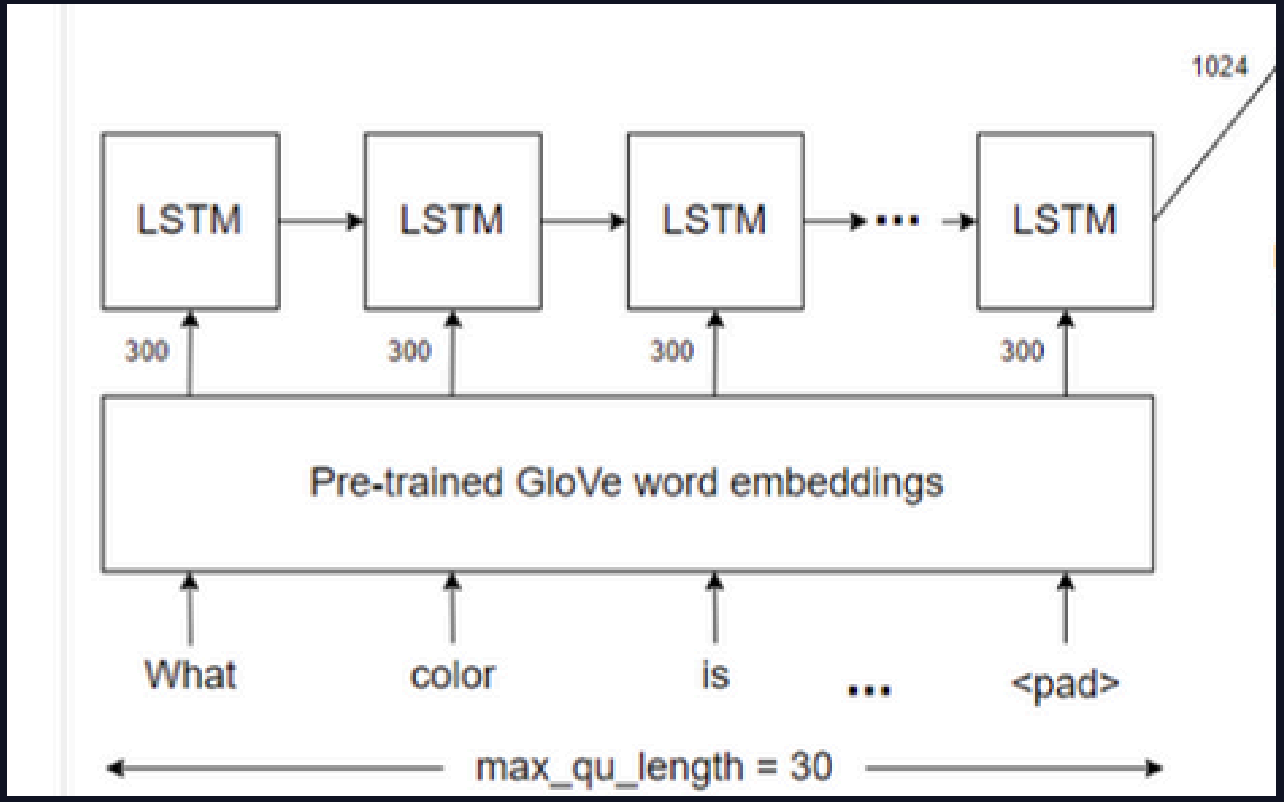
# MODEL ARCHITECTURE: IMAGE ENCODER



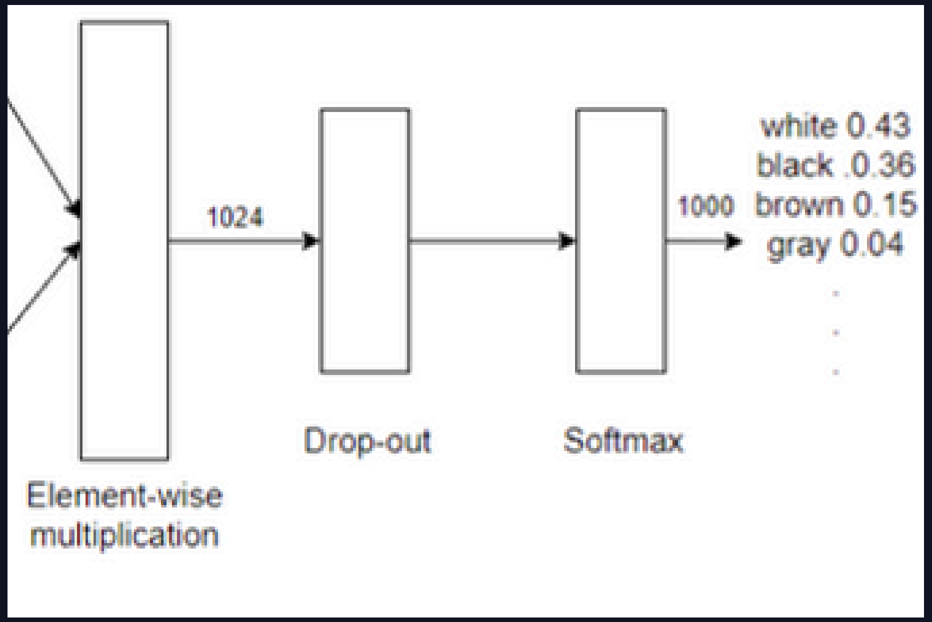
It takes the pre-processed image features extracted using VGG-16 convolution neural network. These features were stored with the shape of (49, 512). The model flattens the image features and then feeds them to a fully connected layer with 1024 neurons and uses the RELU activation function. This part outputs a 1024-dim embedding of the image.

# MODEL ARCHITECTURE: QUESTION ENCODE

It takes a padded tensor of the vocabulary indices for each word in the question sentence. This tensor has the length of max_qu_length = 30. It uses an embedding layer initialized using pre-trained GloVe-300 word embeddings to replace each word in the sentence with its representative 300-dim vector.



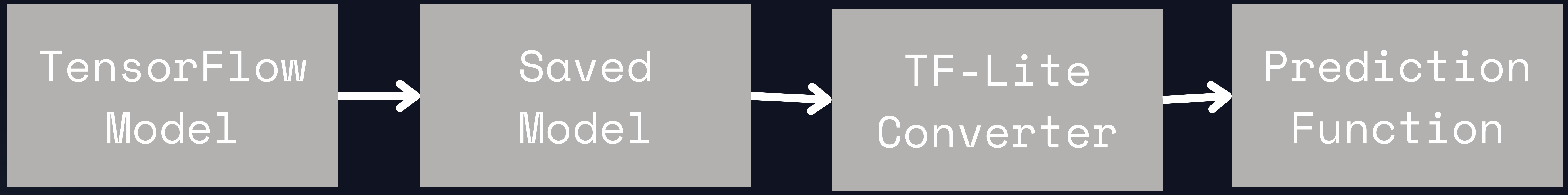
# MODEL ARCHITECTURE:ANSWER PREDICTOR



In this part, the image embedding and the question embedding are fused together using element-wise multiplication. The results of the multiplication are then fed into a drop-layer with a drop rate of 20%, which helps to prevent overfitting, and then into a fully connected layer of K = 1000 neurons and it uses softmax activation function to provide a probability distribution over K answers

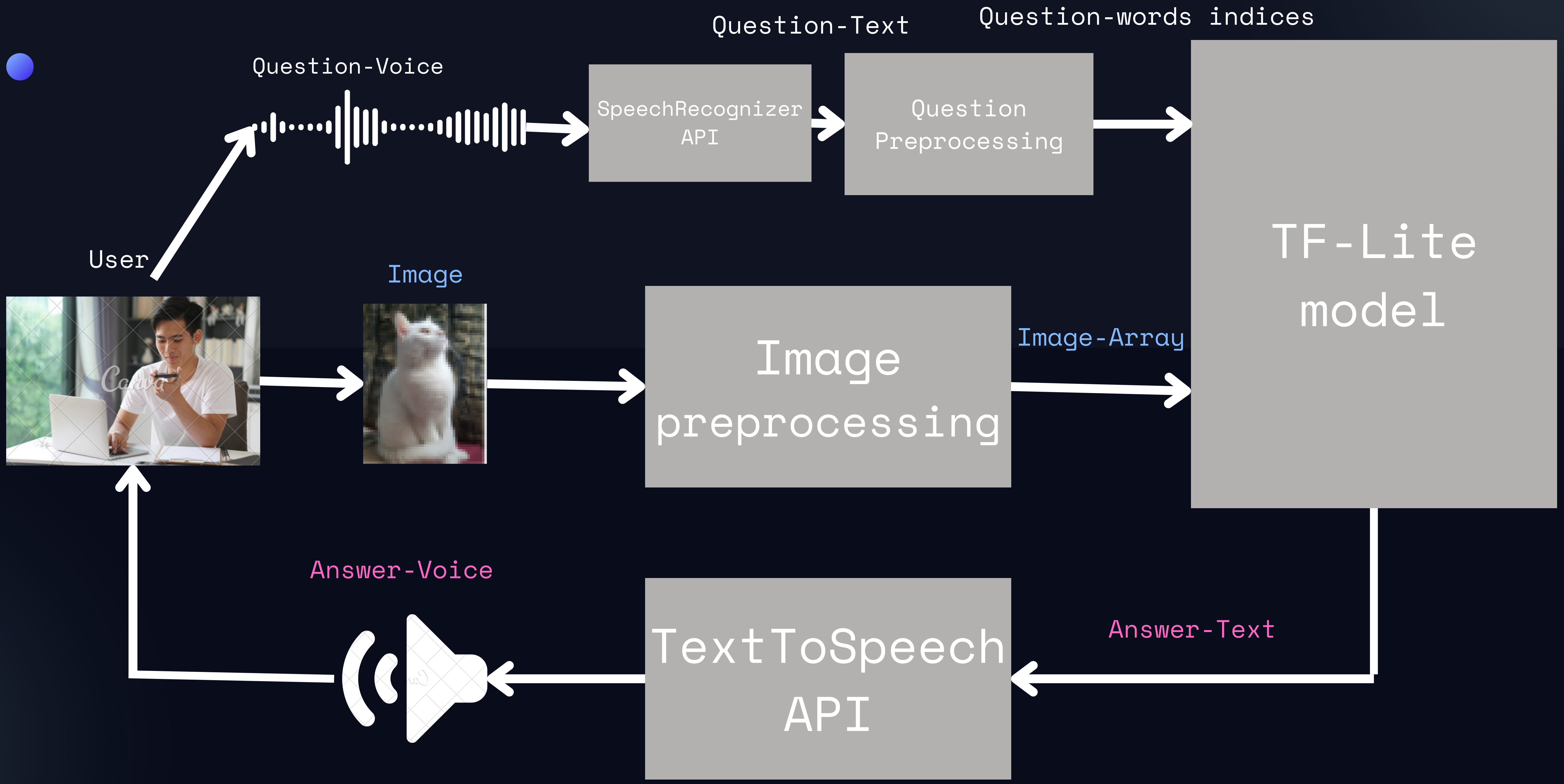
# MODEL DEPLOYMENT

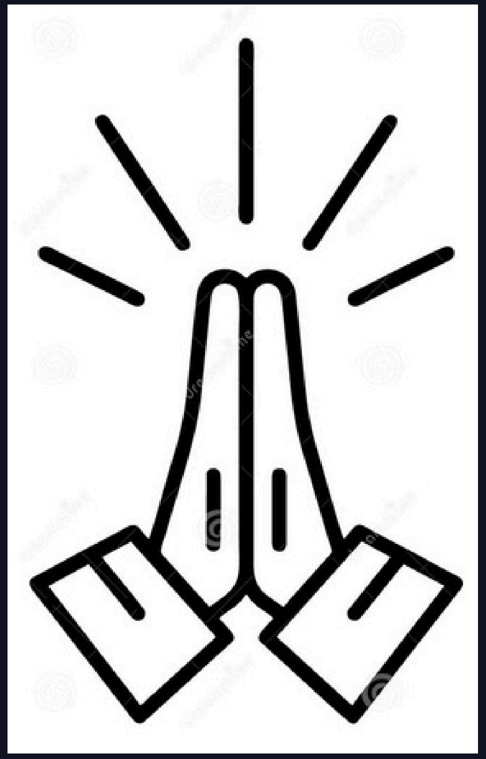
We used the TF-Lite library to convert the already trained VQA model into a tflite format that can be used for inference on our Android application.



TensorFlow Model → Saved Model → TF-Lite Converter → Prediction Function

# MODEL DEPLOYMENT: ON THE ANDROID APP





# THANK YOU