**AI PROJECT ON PARAPHRASES DETECTION**

**[Client Project]**

**GROUP ID: PTID-AI-SEP22-1015**

# BUSINESS PROBLEM:

# BASED ON THE TEXT DATA WE NEED TO PREDICT WHICH OF THE PROVIDED PAIRS OF QUESTIONS CONTAINS TWO QUESTIONS WITH THE SAME MEANING

# 

Quora is a place to gain and share knowledge—about anything. It’s a platform to ask questions and connect with people who contribute unique insights and quality answers. This empowers people to learn from each other and to better understand the world.

Over 100 million people visit Quora every month, so it's no surprise that many people ask similarly worded questions. Multiple questions with the same intent can cause seekers to spend more time finding the best answer to their question and make writers feel they need to answer multiple versions of the same question. Quora values canonical questions because they provide a better experience to active seekers and writers, and offer more value to both of these groups in the long term.

## Data Overview

* Data will be in a file Train.csv
* Train.csv contains 5 columns : qid1, qid2, question1, question2, is\_duplicate
* Size of Train.csv - 60MB
* Number of rows in Train.csv = 404,290

## Data fields

* id - the id of a training set question pair
* qid1, qid2 - unique ids of each question (only available in train.csv)
* question1, question2 - the full text of each question
* is\_duplicate - the target variable, set to 1 if question1 and question2 have essentially the same meaning, and 0 otherwise.

Import Necessary Library

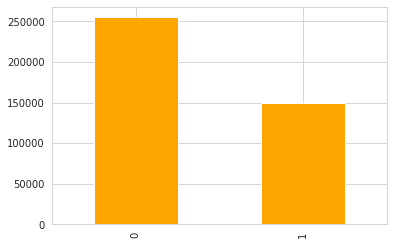
* In this project we are use Tensorflow, Keras, NLTK, Reg-ex, Numpy, Pandas, Matplotlib, Seaborn etc..

LOAD DATA & BASIC CHECKS

* Load training and testing data separately and check the basics of data
* Check head, tail, shape of data
* Getting the information of data
* Checking the unique question present in data

EXPLOTARY DATA ANALYSIS

* Checking the balance of data



Observation:

* Here we are clearly seen the data is imbalanced.

#### USED AUTOMATED PANDAS PROFILING LIBRARY FOR EDA

Dataset statistics

|  |  |
| --- | --- |
| **Number of variables** | 6 |
| **Number of observations** | 404290 |
| **Missing cells** | 3 |
| **Missing cells (%)** | < 0.1% |
| **Duplicate rows** | 0 |
| **Duplicate rows (%)** | 0.0% |
| **Total size in memory** | 103.2 MiB |
| **Average record size in memory** | 267.7 B |

Variable types

|  |  |
| --- | --- |
| **Numeric** | 3 |
| **Categorical** | 3 |

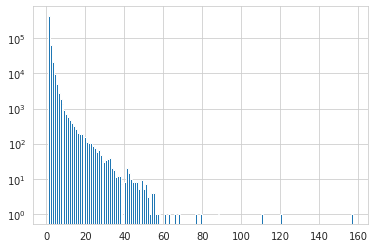
# In question1 there is one missing value and question2 two missing present

# 3 Numerical ad 3 categorical variable is present with a unique feature

#### REPEATED QUESTIONS

* Unique questions: 537933
* Repeated questions: 111780

**Repeated questions histogram**

****

SPLIT DATA INTO TRAINING AND TESTING

* Define X\_train & y\_train in the form of an array
* Create X\_test & y\_test in the form of an array

### DATA PRE-PROCESSING

1. Checking Missing Value: In this data total 3 missing values are present, question1 contain one and the question2 contain two
2. Checking Duplicates: There are no duplicates present in the data

3.Text Pre-Processing:

* Used Keras text processing to convert text data into numerical/vector.

Padding & Sequencing:

* we convert all the questions in columns 'question1' and 'question2',

of both train and test set, into sequences, i.e, in the form of numbers since the machine can only process numbers and not texts.

* We define the maximum length of each question and the questions which contain less than the required length are padded with zeros to make the length of the sentence equal to the mentioned length.

Loading Glove word embedding

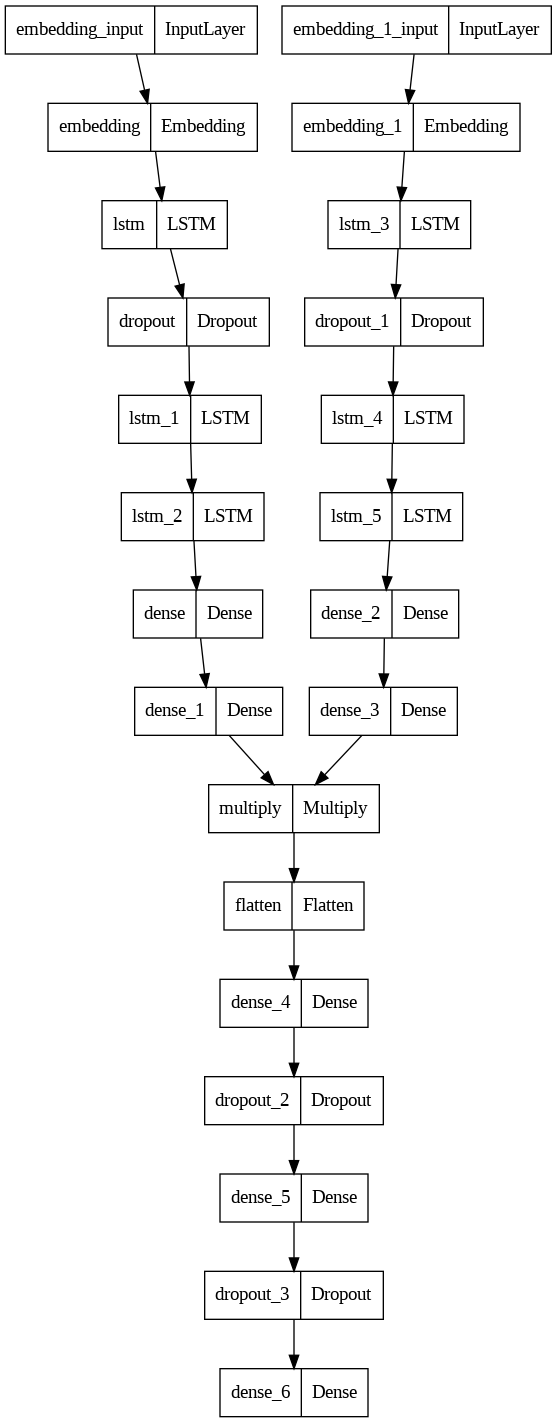
GloVe (Global Vectors) is a model for distributed representations. The model is an unsupervised learning algorithm for obtaining vector representation for words. This is taken care of by mapping words into meaningful space where the distance between the words is related to semantic similarity. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.

### LSTM (Long Short Term Memory)

Long Short-Term Memory is a variation of RNN that is used to eliminate the Vanishing Gradients Problem. It is much more powerful and complex than other variations of RNN, i.e., GRU.

1. Create 1st model for question1
2. Create 2nd model for question2
3. Merging output of the two model i.e(model\_q1, model\_q2)

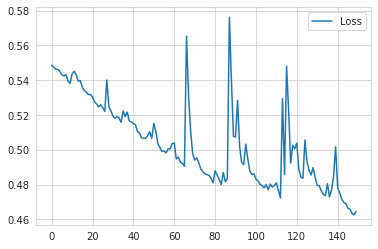
### VISUALISE MODEL

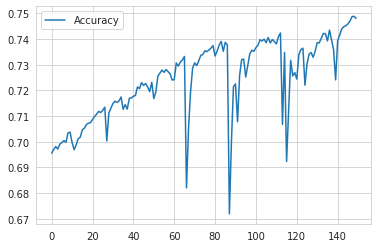


### COMPILE AND TRAIN MODEL

* Used adam optimizer with sparse categorical cross-entropy loss
* Fit model for training and set the batch size of 2000 and train the model on 150 epochs

PLOTTING TRAINING LOSS AND ACCURACY





PREDICTION USING TEST DATA

* Make predictions using pre-process test data

MODEL SAVING

* Save the model using .h5 extension

CHALLENGES FACED:

* Understand the business problem
* Text processing: To choose which technique is better for this type of problem
* Training time: It takes lots of time to training.