Course Project Report (Jul-Dec 2020)

Duplicate Questions Pair Detection

Submitted By

ADITYA KARIA (171IT203)
SHASHANK JAISWAL (171IT239)
THEJASWINI D M (171IT243)

as part of the requirements of the course

Information Retrieval (IT458)

under the guidance of

Dr. Sowmya Kamath S, Dept of IT, NITK Surathkal

undergone at

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL

NOVEMBER 2020

Department of Information Technology

National Institute of Technology Karnataka, Surathkal

<u>CERTIFICATE</u>

This is to certify that the Course project Work Report entitled **"Duplicate Questions Pair Detection"** is submitted by the group mentioned below -

Details of Project Group

Name of the Student	Register No.	Signature with Date
Aditya Karia	171IT203	
Shashank Jaiswal	171IT239	Shashank
Thejaswini D M	171IT243	Thejaswini

as the record of the work carried out by them as part of the course **Information Retrieval (IT458)** during the semester **Jul - Dec 2020**. It is accepted as the Course Project Report submission in the partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Information Technology.**

(Name and Signature of Course Instructor)
Dr. Sowmya Kamath S

DECLARATION

We hereby declare that the project work report entitled **“Duplicate Questions Pair Detection"** submitted by us for the course **Information Retrieval (IT458)** during the semester **July - Dec 2020**, as part of the partial course requirements for the award of the degree of Bachelor of Technology in Information Technology at NITK Surathkal is our original work. We declare that the project has not formed the basis for the award of any degree, associateship, fellowship or any other similar titles elsewhere.

Details of Project Group

Name of the Student	Register No.	Signature with Date
Aditya Karia	171IT203	
Shashank Jaiswal	171IT239	Shashank
Thejaswini D M	171IT243	Thejaswini

Place: NITK, Surathkal
Date: 22 November, 2020

Duplicate Questions Pair Detection

Abstract— **A forum which is growing like Quora comprises a set of queries(questions) and answers created by a user. The users create, edit, and arrange the questions and responses. A huge number of Users on the Quora platform makes it inevitable to have multiple queries of similar intention from different users, this raises the problem of having questions that are redundant. It will be easier to find high-quality answers and save time by identifying duplicate questions efficiently.In this paper, we have exploited the techniques of BoW, TF-IDF and Universal Sentence Encoder for identifying pair of duplicate questions from the publicly available dataset which was released by Quora, and have been able to achieve better results in terms of accuracy than the onces used previously for the same purpose.**

Keywords - BoW, TF-IDF, Universal Sentence Encoder

I. INTRODUCTION

We know that Quora is a social networking website where user questions are posted and answered by experts who provide insights about quality. Via editing questions and suggesting more correct responses to the submitted questions, other users will cooperate. According to statistics given on 17 September 2018 by the Director of Product Management at Quora[29]. Every month,Approximately 300 million unique visitors receive quorum requests, raising the issue of multiple users asking similar questions with the same intention, but in different words.Many similarly worded questions can cause readers to spend more time looking for the right answer and making writers respond to multiple variations of a same query.

Therefore for logically distinct problems, Quora has an essential concept of providing a single query line. Questions such as for instance,' How to become a good photographer? "and What am I supposed to do in order to be a better photographer? are the same because they both have the same meaning and can only be addressed once. Certain questions, such as How old are you?" "and What age is yours? "The wording is not the same. The background remains the same however. Such questions are also often called duplicate questions. Getting separate pages for such questions can be an overhead. Thus, finding and integrating the duplicate questions at Quora allows the exchange of information to be better efficient in several ways.

In this manner, on a single thread, the users can get solutions to all the queries and writers do not have to answer the same solution for the same question at different locations. They will get more users than if the readers are split into different threads. To combine the duplicate questions into one, It is currently using Random-Forest with many handcrafted features.This thesis aims to achieve the same goal by applying advanced Neural Network Architecture to achieve greater accuracy and save time used in engineering of complex feature.

In the scope of this project, we propose to implement various different models and attempt to improve upon the current implementations. Some of the prominent models covered below are: BoW (Bag of Words), TF-IDF (Word-level, Character-level, and n-gram level), XG-Boost Classifier, Word Embedding, Sentence Embedding, and the Universal Sentence Encoder on a popular and publicly available dataset released by Quora itself. Apart from this, we have also covered some pre-built Keras models and reviewed their performance. We have further been able to achieve better accuracy compared to the existing models as prescribed in the base paper. The results of our implementation and comparisons have been discussed in detail below.

II. RELATED WORK

The role of identifying duplicate texts involves profound semantic level comprehension in natural language processing. A duplicate texts is nothing but a re-statement of an article, statement etc. Identification of duplicate texts means recognition of same words or phrases in documents of either arbitrary or similar size. In multiple languages, some of the paraphrase detection implementations provide plagiarism detection for fiction, nonfiction, and science articles.
Social networking networks such as Instagram, Facebook, and answering questions forums such as Quora, Stackoverflow are the places where duplicate texts and their recognition are highly important. For suggestion sites such as Quora, the detection of duplicate sentences may be beneficial because if a user asks any query and if any other customer requests it beforehand, the new question can be answered immediately.

In identifying the sentence similarity, Traditional NLP techniques have less accuracy.
In (Sujith Viswanathan and Soman,) this, the output of 6 s machine learning algorithms that are supervised, in two separate duplicate texts corpus is discussed, and it emphasizes on examining how these algorithms correctly identify sentences as duplicates and non-duplicates present in the corpus. The work done on the corpus of Quora and Twitter suggests that machine learning algorithms work well for detecting redundant phrases between the pair of sentences. Support Vector Machine, Logistic Regression,Naive Bayes, K-Nearest Neighbor, Decision Tree and Ran-dom Forest were the algorithms considered for this work.Among these algorithms considered, no algorithm in both corporas failed to detect paraphrases. The algorithms of the Random Forest

and K-Nearest Neighbor worked equally well in both. But the validation of sentence similarity identification performance is not done.

In (Li,)In this the author has explored how to solve the same problem of matching and de-duplication using a different process, and again as an extension of the classifier, they have addressed the task of de-duplication using Word2Vec and Xgboost.

It also uses Google News corpus, pre-trained by word2vec. In the results The Xgboost achieved 0.77 test accuracy on all the new features we developed, which was lower than the TF-IDF + Xgboost character level at 0.80, but the recall was lifted for duplicate questions from 0.67 to 0.73, which is a substantial improvement, but the increase in the accuracy is still not as up to the mark as what is required.

Simplification of the Clinical Text has since Prospective implementations, such as simplification of applications Patients' case notes for a clearer understanding of their clinical conditions. In-depth learning It has emerged as a popular technique for different types of Pre-conditioned tasks for interpreting natural language with broad annotated datasets.

In (Viraj Adduru,) this the authors suggested a methodology for the development of preliminary Medical Paraphrasing and Clinical Text Datasets. Simplification to encourage Deep Learning Training.

In this a methodology for the production of preliminary Medical Paraphrasing and Clinical Text Datasets Simplification to promote the training of clinical paraphrase generation and simplification models based on deep learning.

The researchers have used Bi-Directional LSTM, a deep learning methodology to perform the same task and have created preliminary data-sets for clinical paraphrasing in order to use it for simplification of sentences in a clinical environment, but this requires a string similarity metrics to identify similar sentence pairs accurately along with that the dataset used is a small one, they still have to explore it on a wide range of dataset to efficiently analyze the results of the proposed model.

In (Elkhan Dadashov,)this two approaches focused on Quora Long Short-Term Memory (LSTM) networks were discussed by the scientists.

Duplicate dataset of questions. A Siamese architecture with the trained representations of a single LSTM operating on both phrases is used in the first model. The second technique utilizes two LSTMs with the Two sentences in order, and the first sentence (word-by-word attention) in the second. Their best ones 79.5 percent F1 with 83.8 percent precision on the test set was achieved by the model.

The authors have used a Siamese architecture with the learned representations from a single LSTM running on both sentences and another method uses two LSTMs, which was suggested as a future work in the first reference.A large part of the results discrepancy is found, which may be due to a disparity in embeddings and a lack of coverage of our pre-trained embeddings. In this the problem is posed with an inherent symmetry between the two sentences and it produces less accuracy, not up to the mark as what expected. All the above mentioned related work provided less accuracy and also required a better word embedding Technique.

III. DATASET AND PREPROCESSING

A. *Dataset*

Quora, an official QA platform , released a public tsv file which consists of question pairs and information about whether the question are duplicate or not.The dataset consist around 404k pairs of questions and their labels. The description of dataset is given in below table.

TABLE I: Dataset Description

Attributes	Description
id	unique identifier for each question pair
qid1	unique identifier for question 1
qid2	unique identifier for question 2
question1	full content of question 1
question2	full content of question 2
is$_{d}uplicate$	binary value0,1

The isduplicate column indicates whether the questions are duplicate or not for that particular question pair. Value 0 indicates that questions are not duplicate and value 1 indicates that questions are duplicate. The question pair example has been shown in the below figure.

TABLE II: Contrbutions of Team Members

question1	question2
How does the Surface Pro himself 4 compare with iPad Pro?	Why did Microsoft choose core m3 and not core i3 home Surface Pro 4?
Should I have a hair transplant at age 24? How much would it cost?	How much cost does hair transplant require?
What but is the best way to send money from China to the US?	What you send money to China?

B. *Preprocessing*

The dataset contains attributes like id, qid1 and qid2 which are not redundant columns for further processing. These columns are removed from the dataset. All the text available in the dataset are converted to their normalized form by removing punctuations, changing them to lower case etc. For further preprocessing , phrases like what's is has been changed to what is, abbreviations like US has been normalized to usa, special symbols are changed to their name and verbs like 've has been changed to have and wasn't to was not.

Not all stopwords can be removed as words like what and why have relevance in this dataset context. Words should not be changed in their root form as same root word can have different signals.

Once all the these above steps are performed, then the dataset is send for further processing.

IV. Proposed Methodology

A. Bag of Words(BoW)

The BoW model is a technique for representing text data while processing text for further use. The model is very simple and flexible and can be easily used for extracting features from set of documents. The BoW model is mainly dependent on two factors:

1) Vocabulary
2) Occurrence of words

The model is called “bag” because any information related to order or structure of word is hidden and is ignored during converting text to vectors. The model only looks for occurrence of the word and totally ignore where the word occur in the document. The below table depicts the vector conversion process of a text using BoW model.

TABLE III: Vector conversion of Text using BoW

Vocabulary 1	The, Sun, More, Bright, Today
Vocabulary 2	The, Sun , Brighter, Today
Text	The Sun Looks More Brighter Today
Vector From Vocab 1	[1,1,0,1,0,1]
Vector From Vocab 2	[1,1,0,0,1,1]

Although it is easy to use and understand, the BoW models has following limitations:

1) Vocabulary Construction impacts the sparsity of the documents.
2) Sparsity increases spaces and time complexity.
3) Contextual meaning of a word is not taken care of

B. TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) (Viswanathan et al., 2019) is technique to assign a score to a particular from the document corpus. Unlike BoW model, TF-IDF model also counts for the relevance of the word in that particular documents.TF-IDF model counts the number of occurrence of a particular word along with the number of documents that contains the word. So , if a word is very rare then its score is high and if a word is common in corpus then its score is low. TF-IDF scores are computed by taking into account following two factors

1) Term Frequency(TF) of a word, character or n-gram

$$tf\,(t, d) = log\,(1 + freq\,(t, d))$$

Fig. 1: Term Frequency Formula

where freq(t,d) is frequency of a keyword t from document d from the corpus

2) Inverse Document Frequency(IDF)

$$idf\,(t, D) = log\,\left(\frac{N}{count\,(d \in D{:}t \in d)}\right)$$

Fig. 2: Inverse Document Frequency Formula

Combining these two terms , the formula for tf-idf weights can be written as:

$$tf\,idf\,(t, d, D) = tf\,(t, d)\,.\,idf\,(t, D)$$

Fig. 3: TF-IDF weight Formula

With the above formula, weights for all the words from document corpus is generated. Then with the word weights, text are converted to vectors. TF-IDF can be further categorised as:

1) **word-level**: frequency of word is taken
2) **character-level**: frequency of character is taken
3) **n-gram level**: frequency of n-gram is taken

C. XG Boost Classifier

XG Boost is an enhanced gradient boosting algorithm and it is one of most used algorithm in machine learning. Once the vectors are created using any of the above two methods discussed, then the vectors are passed to XG Boost Classifier for training and testing.

D. Word Embeddings

Word embedding(Ma and Zhang, 2015) is a form of word representation that enables words to have a similar representation with a similar meaning. Various word embedding techniques are available. The most commonly used techniques are Word2vec, Glove and Google news vector. These techniques are already trained on large sized vocabulary.

E. Sentence Embeddings

Sentence embeddings are created by splitting sentences on words and assigning each word their vector by using any word embeddings technique. But this way of generating vectors would be quite tedious for a sentence which contains lots words. Instead of dealing with words, we can directly deal with sentences using techniques like Universal Sentence Encoder.

F. Universal Sentence Encoder

Universal Sentence Encoder is developed by TensorFlow developer and it has been trained on a very large scale dataset which consists of magazines , news, articles etc. The input to the model is variable length English text and the output is a 512 dimensional vector. The vector representation of texts helps in finding semantic similarity between the questions.

One advantage of this method with respect to other methods is that no pre-processing is required as instead of creating vectors for words , this model creates vectors for sentences.

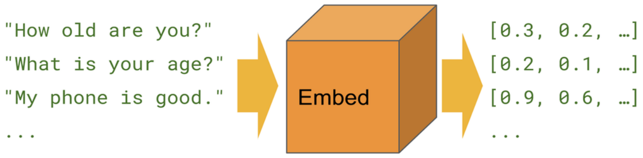


Fig. 4: Universal Sentence Encoder Example

The Universal Sentence Encoder is the hidden layer in Keras model for creating the embeddings of the text. It returns 512 dimensional vectors which is further passed to the output layer of Keras model for training and validating.

G. *Keras Model*

The model is loaded from Keras functional API and then input tensor and output tensor is passed to the model. One hidden layer is created for geenrating embeddings for the text passed as an input to the model. So, there are three layers: input layer, hidden layer and output layer. The model is compiled with all the layers. Then the data in required format is passed to the model.fit function which trains the data for specified number of epochs on training data and then returns the metrics after testing the model on validation data.

V. IMPLEMENTATION SPECIFICS

A. *Algorithm for BoW model*

Algorithm 1 BoW Implementation

```
1) Initialisation
2) Load CountVectorizer from sklearn
3) Fit the text in CountVectorizer
4) Create vectors for the text available in the dataset
5) Load XGBoost classifier from the library xgboost
6) Split the dataset into train and test data
7) Train and test the dataset with XG Boost classifier
8) Compute accuracy, precision, recall and f1-score
```

B. *Algorithm for TF-IDF model*

Algorithm 2 TF-IDF Implementation

```
1) Initialisation
2) Load TFIDFVectorizer from sklearn
3) Fit the text in TFIDFVectorizer
4) Create vectors for the text available in the dataset
5) Load XGBoost classifier from the library xgboost
6) Split the dataset into train and test data
7) Train and test the dataset with XG Boost classifier
8) Compute accuracy, precision, recall and f1-score
```

C. *Algorithm for Universal Sentence Encoder*

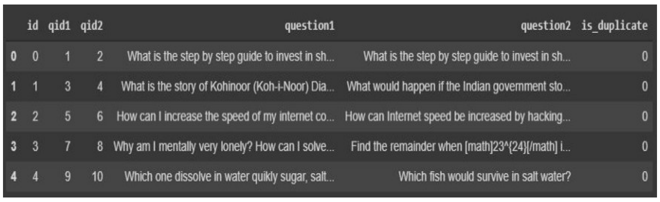
Algorithm 3 Universal Sentence Encoder Embedding Implementation

```
1) Initialisation
2) Load Universal Sentence Encoder.
3) Load Keras Model
4) Create embeddings for all the questions from the dataset
5) Split the dataset into train and test data
6) Train the keras model with training data for 20 epochs
7) Test the model with the testing data
8) Compute accuracy, precision, recall and f1-score
```

VI. EXPERIMENTAL RESULTS AND ANALYSIS

A. *Dataset*

Quora question pairs have been used for training and testing our model. The dataset was released by Quora itself.



	id	qid1	qid2	question1	question2	is_duplicate
0	0	1	2	What is the step by step guide to invest in sh...	What is the step by step guide to invest in sh...	0
1	1	3	4	What is the story of Kohinoor (Koh-i-Noor) Dia...	What would happen if the Indian government sto...	0
2	2	5	6	How can I increase the speed of my internet co...	How can Internet speed be increased by hacking...	0
3	3	7	8	Why am I mentally very lonely? How can I solve...	Find the remainder when [math]23^{24}[/math] i...	0
4	4	9	10	Which one dissolve in water quikly sugar, salt...	Which fish would survive in salt water?	0

Fig. 5: Representation of Quora Question Pair Dataset

The following analysis has been done on the dataset to get graphs of normalized value of shared words and common words in the dataset.

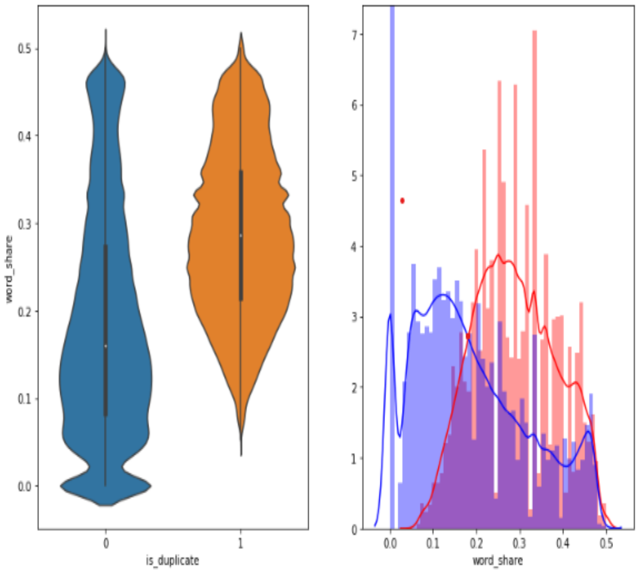


Fig. 6: Shared Words

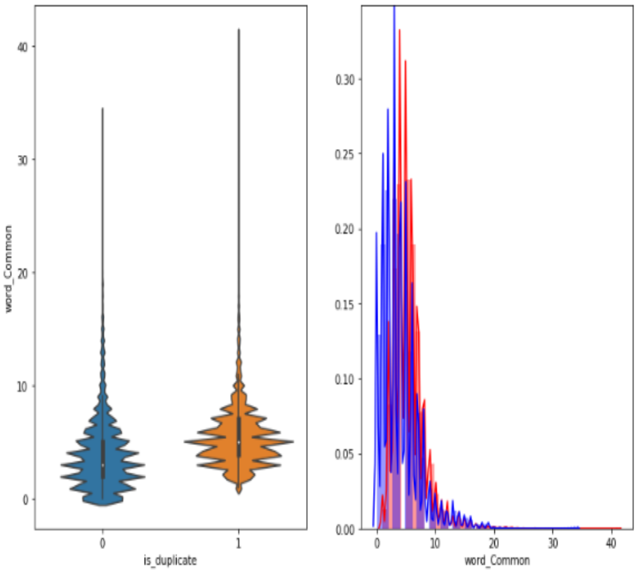


Fig. 7: Common Words

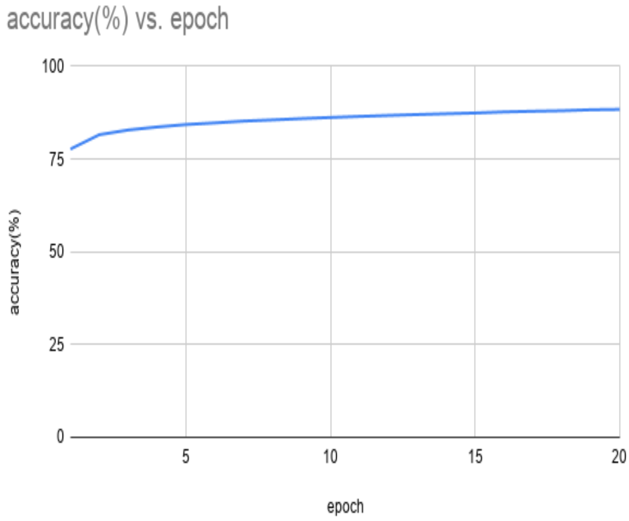


Fig. 8: Epoch Vs Accuracy graph

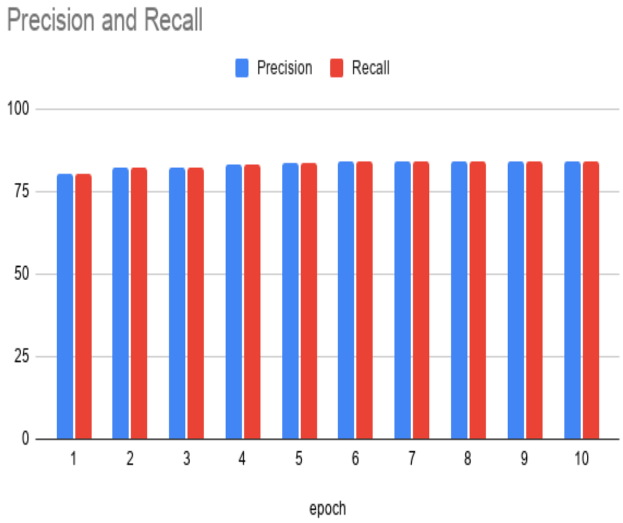


Fig. 9: Precision Recall Plot For 10 epochs

As it can be seen from the fig 6 and fig 7 that common words should be removed and common words can not be a criteria to identify duplicate pairs as for both labels 0 and 1 graph is almost similar.

The dataset is splitted between training and testing dataset. The training dataset consists around 303k values where as the testing dataset consists around 101k values.

B. Training Details

The model is trained with the training dataset for 20 number of epochs. After 20 epochs, the model seems to be converging to a fixed value. Paddings are created for the text whose size is less than the required size. We have used adam optimizer and tried different learning rates.

C. Proposed Model Results

In the final set of experiments, sentence vectors are created using BoW model, TF-IDF model and Universal Sentence Encoder. The vectors created by using BoW model and TF-IDF model are then trained and tested by using XG Boost Classifier. The sentence encodings created by Universal Sentence encoder are then further processed by keras model for training and testing dataset. The algorithm is ran for 20 epochs. Fig8 shows the changes in accuracy as model trains more and more. After 17 epochs the model seems to be converging to a value i.e. 84.83. The below table summarises the results obtained by our proposed model.

The table shows that one of our proposed model i.e. Universal Sentence Encoder(USE) model outperforms all other models. The USE model gives around 88.4 percent accuracy which is quite impressive as compared to the base paper model which uses Siamese MaLSTM model.

D. Blind Study Findings

After going through the testing dataset, we founded that if the subject of one question is the identifying component of other question, then sometime model is not able to assign correct label to the question pair. Below given are some examples to illustrate this.

Q1. Were the Clintons paid by the Clinton Foundation?
Q2. Was Chelsea Clinton really paid $700,000 a year working for the Clinton Foundation ?
is marked Positive for being the same. But the two questions

TABLE IV: Summary of Results

Model	Precision	Recall	F1-Score	Accuracy
BoW	78.0	75.0	76.0	79.0
TF-IDF Word Level	79.0	75.0	76.0	79.0
TF-IDF Character Level	82.0	79.0	80.0	82.0
TF-IDF n-gram Level	74.0	67.0	68.0	74.0
Siamese MaLSTMs	73.0	86.8	79.3	83.8
USE Model	84.82	84.83	84.83	**84.83**

are different in context.

Another observation is that, sometimes model is not able to recognise the audience for the question. Same question can be for different audience and thats why it should be treated as different question. Below example illustrates this.

Q1. Why India is more successful in Cricket than Hockey?
Q2. Why in India cricket is more famous than hockey?

The questions are quite different. The first question is player related whereas the second question is audience related. This question pair is marked Positive for being the same.

VII. DISCUSSIONS

By introducing a model that better recognizes the similarity of sentences, the duplicate detection tasks of these two companies can be expanded. Some remarkable results can be obtained by conventional approaches, such as TFIDF. That's one of the reasons why Google has long been using TFIDF to evaluate the relevance of a given keyword to a given page in indexing and data retrieval.

Lots of preprocessing is required for cleaning of dataset. There are lots of rows which needs to be removed from the dataset as it has more number of columns than other rows. Lots of preprocessing steps are reuired for processing text before passing it to BoW and TF-IDF model. With Universal Sentence Encoder, no preprocessing steps are required. For creating sentence embeddings, it requires less amount of time as compared to word embedding techniques discussed in the above sections.

In this paper we have implemented and evaluated a new model against the existing models and our proposed model provides better accuracy. We have used accuracy as performance metrics.

We have obtained 84.83% accuracy which is much higher than other models that we are using to compare our proposed model with. Our model outperforms the rest of them.
The challenges that we faced while implementing this are machine capability, where the machine's we used to implement and evaluate the proposed model was not supposed to exhaust the resources, We overcame that issue by including smart techniques in the model while evaluating and executing.

VIII. CONCLUSIONS AND FUTURE WORK

This work proposed 3 models i.e. BoW, TF-IDF and Universal Sentence Encoder embeddings to create vectors for text of all questions available in Quora Question Pair dataset and identify the duplicate questions from the dataset. The proposed embeddings technique results in better accuracy as compared to already used model Siamese MaLSTMs. More importantly the embeddings generated by Universal Sentence Encoder doesn't require any preprocessing steps which helps in saving a lot of time. The embedding technioque also focuses on contextual meaning of a sentence. Creating word embeddings for the whole Quora Question Pair by using any of word embeddings techniques like glove , word2vec etc requires around 5-6 hours which is quite more as compared to the proposed embedding technique which requires around 1.5 to 2 hours.

The future work involves using other sentence embeddings techniques like sentence bert and comparing their results with techniques used in our work. Also training of sentence embeddings model can be done on quora context so as to get better embedding for sentences.

IX. INDIVIDUAL CONTRIBUTIONS

TABLE V: Contrbutions of Team Members

Aditya Karia	Shashank Jaiswal	Thejaswini D M
Dataset Cleaning	Dataset Preprocessing	Dataset Preprocessing
TF-IDF	Dataset Analysis	BoW
word2vec	Sentence encoding	XG Boost Classifier
Keras Model	Universal Sentence Encoder	Blind Study Findings

REFERENCES

Elkhan Dadashov, Sukolsak Sakshuwong, K. Y. *Quora Question Duplication.*

Li, S. *Finding Similar Quora Questions with Word2Vec and Xgboost.*

Ma, L. and Zhang, Y. (2015). *Using Word2Vec to process big text data.*

Sujith Viswanathan, Nikhil Damodaran, A. S. A. G. M. A. K. and Soman, K. P. *Detection of Duplicates in Quora and Twitter Corpus: Proceedings of ICBDCC18.*

Viraj Adduru, Sadid A. Hasan, J. L. Y. L. V. D. K. L. A. Q. O. F. *Towards dataset creation and establishing baselines for sentence-level neural clinical paraphrase generation and simplification.*

Viswanathan, S., Damodaran, N., Simon, A., George, A., Kumar, M., and Kp, S. (2019). *Detection of Duplicates in Quora and Twitter Corpus: Proceedings of ICBDCC18.*