CHATBOT USING PYSPARK

BIG DATA PROJECT REPORT

Submitted by

NAME	SRN
ROSHINI BHASKAR	PES2201800122
VISHWAJEET SHIVAJI HOGALE	PES2201800128
SHARIKA VALLAMBATLA	PES2201800432

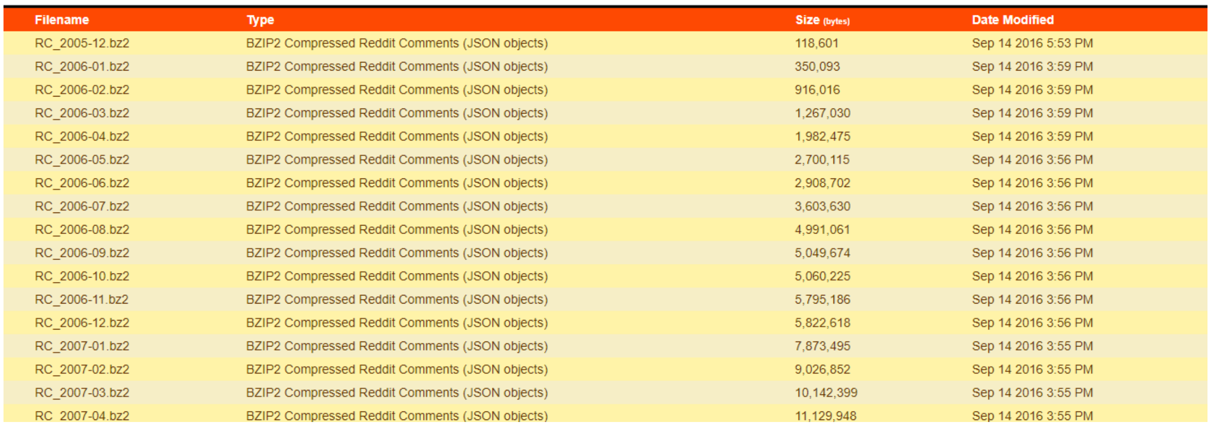
INTRODUCTION

We live in the digital era where we get access to everything at the touch of a screen. Be it a money transfer, shopping online, buying groceries and many more. Thanks to technological advances that have helped us find the most comfortable way to solve our daily problems. Humans are constantly fascinated with auto-operating gadgets. The latest trend that is getting the attention of most of the tech industry is chatbots. This conversational technology is expanding rapidly through virtual assistants like Siri, Alexa, Google Mini, etc. taking human-like engagement to a new dimension. Just like mobile apps, chatbots are becoming popular among companies to enhance the customer experience.

Chatbots are the best AI agents that answer users' queries, offer help and information, whenever required. Since they are automated and can handle multiple queries at a time, it saves time and efforts of human workers allowing them to focus on tasks that only humans can do. Believe it or not, chatbots are the best collector of big data that handle queries and process the information at a greater speed than humans. After bots collect the data, these data are analyzed to provide better services and boost customer experiences. In this project we will use reddit data to make a general conversational chatbot using nlp techniques on the hadoop ecosystem, using pyspark.

DATASET

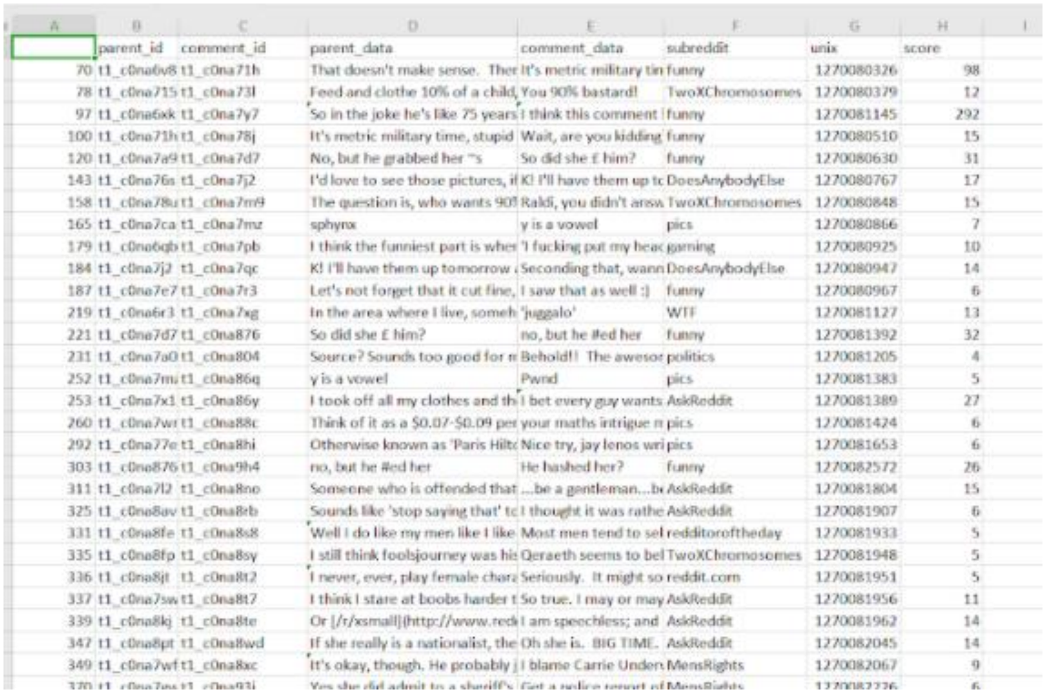
Link:https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/?st=j9udbxta&sh=69e4fee7



Filename	Type	Size (bytes)	Date Modified
RC_2005-12.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	118,601	Sep 14 2016 5:53 PM
RC_2006-01.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	350,093	Sep 14 2016 3:59 PM
RC_2006-02.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	916,016	Sep 14 2016 3:59 PM
RC_2006-03.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	1,267,030	Sep 14 2016 3:59 PM
RC_2006-04.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	1,982,475	Sep 14 2016 3:59 PM
RC_2006-05.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	2,700,115	Sep 14 2016 3:56 PM
RC_2006-06.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	2,908,702	Sep 14 2016 3:56 PM
RC_2006-07.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	3,603,630	Sep 14 2016 3:56 PM
RC_2006-08.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	4,991,061	Sep 14 2016 3:56 PM
RC_2006-09.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	5,049,674	Sep 14 2016 3:56 PM
RC_2006-10.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	5,060,225	Sep 14 2016 3:56 PM
RC_2006-11.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	5,795,186	Sep 14 2016 3:56 PM
RC_2006-12.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	5,822,618	Sep 14 2016 3:56 PM
RC_2007-01.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	7,873,495	Sep 14 2016 3:55 PM
RC_2007-02.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	9,026,852	Sep 14 2016 3:55 PM
RC_2007-03.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	10,142,399	Sep 14 2016 3:55 PM
RC_2007-04.bz2	BZIP2 Compressed Reddit Comments (JSON objects)	11,129,948	Sep 14 2016 3:55 PM

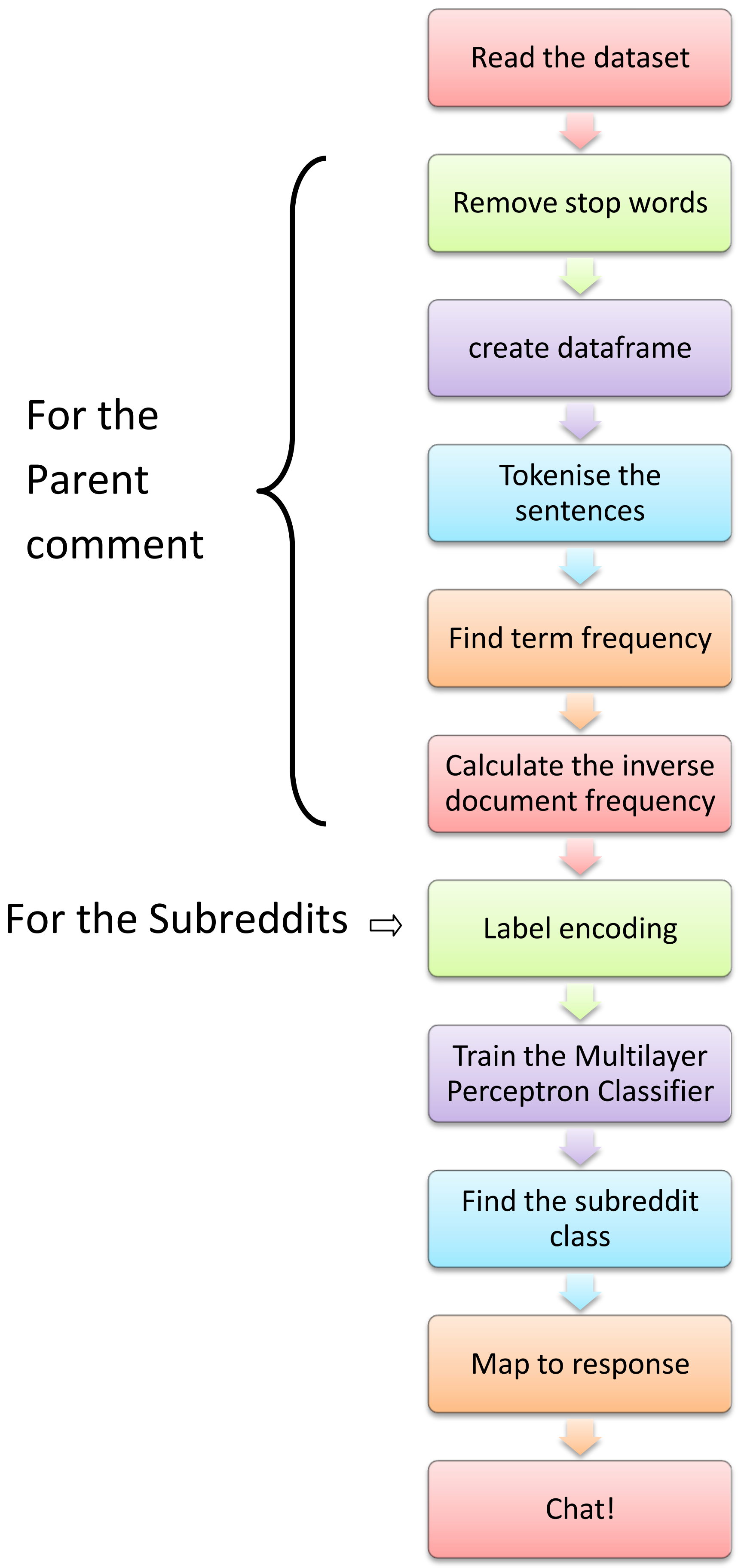
The structure of Reddit is in a tree-form. The parent comments are linear, but replies to parent comments branch out. The structure we need for deep learning is input-output. So we really are trying to get something more along the lines of comment and reply pairs. The comment is the input, the reply is the desired output. Now with Reddit, not all comments have replies, and then many comments will have many replies. The other thing we need to consider is that, as we iterate over this file, we may find a reply, but then we might find a better reply later. The way we did this is to go off of upvotes and then we set limits for scores.

Now we have the filtered file having parent_id, comment_id, parent_data, comment_data, subreddit, unix and score. We put this data on hadoop ecosystem.



	parent_id	comment_id	parent_data	comment_data	subreddit	unix	score
70	t1_c0na6v8	t1_c0na71h	That doesn't make sense. Ther	It's metric military tin	funny	1270080326	98
78	t1_c0na715	t1_c0na73l	Feed and clothe 10% of a child,	You 90% bastard!	TwoXChromosomes	1270080379	12
97	t1_c0na6xk	t1_c0na7y7	So in the joke he's like 75 years	I think this comment	funny	1270081145	292
100	t1_c0na71h	t1_c0na78j	It's metric military time, stupid	Wait, are you kidding	funny	1270080510	15
120	t1_c0na7a9	t1_c0na7d7	No, but he grabbed her "s	So did she £ him?	funny	1270080630	31
143	t1_c0na76s	t1_c0na7j2	I'd love to see those pictures, if	K! I'll have them up tc	DoesAnybodyElse	1270080767	17
158	t1_c0na78u	t1_c0na7m9	The question is, who wants 90!	Raldi, you didn't answ	TwoXChromosomes	1270080848	15
165	t1_c0na7ca	t1_c0na7mz	sphynx	y is a vowel	pics	1270080866	7
179	t1_c0na6qb	t1_c0na7pb	I think the funniest part is wher	'I fucking put my heac	gaming	1270080925	10
184	t1_c0na7j2	t1_c0na7qc	K! I'll have them up tomorrow a	Seconding that, wann	DoesAnybodyElse	1270080947	14
187	t1_c0na7e7	t1_c0na7r3	Let's not forget that it cut fine,	I saw that as well :)	funny	1270080967	6
219	t1_c0na6r3	t1_c0na7xg	In the area where I live, someh	'juggalo'	WTF	1270081127	13
221	t1_c0na7d7	t1_c0na876	So did she £ him?	no, but he #ed her	funny	1270081392	32
231	t1_c0na7a0	t1_c0na804	Source? Sounds too good for n	Behold!! The awesor	politics	1270081205	4
252	t1_c0na7mi	t1_c0na86q	y is a vowel	Pwnd	pics	1270081383	5
253	t1_c0na7x1	t1_c0na86y	I took off all my clothes and th	I bet every guy wants	AskReddit	1270081389	27
260	t1_c0na7wr	t1_c0na88c	Think of it as a $0.07-$0.09 per	your maths intrigue n	pics	1270081424	6
292	t1_c0na77e	t1_c0na8hi	Otherwise known as 'Paris Hiltc	Nice try, jay lenos wri	pics	1270081653	6
303	t1_c0na876	t1_c0na9h4	no, but he #ed her	He hashed her?	funny	1270082572	26
311	t1_c0na7l2	t1_c0na8no	Someone who is offended that	...be a gentleman...b	AskReddit	1270081804	15
325	t1_c0na8av	t1_c0na8rb	Sounds like 'stop saying that' tc	I thought it was rathe	AskReddit	1270081907	6
331	t1_c0na8fe	t1_c0na8s8	Well I do like my men like I like	Most men tend to sel	redditoroftheday	1270081933	5
335	t1_c0na8fp	t1_c0na8sy	I still think foolsjourney was his	Qeraeth seems to bel	TwoXChromosomes	1270081948	5
336	t1_c0na8jt	t1_c0na8t2	I never, ever, play female chara	Seriously. It might so	reddit.com	1270081951	5
337	t1_c0na7sw	t1_c0na8t7	I think I stare at boobs harder t	So true. I may or may	AskReddit	1270081956	11
339	t1_c0na8kj	t1_c0na8te	Or [/r/xsmall](http://www.redd	I am speechless; and	AskReddit	1270081962	14
347	t1_c0na8pt	t1_c0na8wd	If she really is a nationalist, the	Oh she is. BIG TIME.	AskReddit	1270082045	14
349	t1_c0na7wf	t1_c0na8xc	It's okay, though. He probably j	I blame Carrie Unden	MensRights	1270082067	9
370	t1_c0na7ws	t1_c0na93i	Yes she did admit to a sheriff's	Get a police report of	MensRights	1270082226	6

FLOWCHART


Read the dataset
Remove stop words
create dataframe
Tokenise the sentences
Find term frequency
Calculate the inverse document frequency
For the Parent comment
For the Subreddits ⇨
Label encoding
Train the Multilayer Perceptron Classifier
Find the subreddit class
Map to response
Chat!

CODE

IMPORTING ALL LIBRARIES

```
from pyspark.sql import SparkSession

from pyspark.sql.functions import array

from pyspark.ml.feature import HashingTF, IDF, Tokenizer

from pyspark.ml.feature import Tokenizer, RegexTokenizer

from pyspark.ml.feature import OneHotEncoder

from pyspark.ml.feature import StopWordsRemover

from pyspark.sql.functions import udf, col

from pyspark.sql import Row

from pyspark.sql.types import StringType, StructType, StructField, ArrayType,IntegerType

from pyspark.ml.classification import MultilayerPerceptronClassifier

from pyspark.ml.evaluation import MulticlassClassificationEvaluator

from pyspark.ml.feature import OneHotEncoder, StringIndexer

import random
```

STOP WORDS LIST->

```
li =['i','i\'d', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll",
"you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers',
'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who',
'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have',
'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until',
'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before',
'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further',
'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more',
'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't',
'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren',
"aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't",
'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't",
'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn',
"wouldn't"]
```

THE FUNCTION TO RETURN THE RESPONSE BELONGING TO A SUBREDDIT

```
# input- mapping of label to subreddit, predicted label, intents(contains, subreddits and respective responses), spark object.

#returns a response under the respective predicted class.

def output(mapping,pred,intents,spark):

   subreddit = mapping[pred]

   otpt = list(intents.toPandas()[subreddit])

   return [random.choice(otpt[0]),subreddit]
```

A FUNCTION TO REMOVE STOP WORDS

```
# input- A row of the dataframe

#returns the cleaned parent comment along with the rest of the fields

def remove_stop(x,li):

   tep=[]

   for j in str(x[3]).split(" "):

      if(j.lower() not in li):

         tep.append(j.lower())

   return (x[3],x[5]," ".join(tep))
```

A FUNCTION TO REMOVE STOP WORDS FOR THE USER INPUT

```
# input- sentence and  a list of stopwords

#returns the cleaned sentence

def new_stop(x,li):

   tep = []

   for j in x.split(" "):

      if j.lower() not in li :

         tep.append(j.lower())
```

```
    return " ".join(tep)
```

PROCESSING THE USER INPUT TO THEN PASS THROUGH THE MODEL

#input is the user input text with all the required preprocessors

#returns the subreddit class the model predicted

```
def bag_of_words(inp,hashfunc,idfmod,tokenizer,model):
    x = [(Row(new_stop(inp,li)))](remove stop words)
    schema = ["Filtered_data"]
    # # Apply the schema to the RDD.
    sP = spark.createDataFrame(x, schema) (make it a dataframe)

    yellow = sP.select("Filtered_data")
    yellow.show()
    sP = tokenizer.transform(sP) (tokenize the input)
    sP = hashfunc.transform(sP) (find term frequency)
     rDD = idfmod.transform(sP) (get the inverse document frequency)
    result = model.transform(rDD) (pass it through the model)
    result.show()
    predictionAndLabels = result.select("prediction")(get the predicted class)
    return predictionAndLabels.toPandas()['prediction'][0]
```

MAIN FUNTION

```
if __name__ == "__main__":
```

Spark session is a unified entry point of a spark application from Spark 2.0.

```
    spark = SparkSession \
        .builder \
        .appName("Chatbot") \
        .config("spark.some.config.option", "some-value") \
    .getOrCreate()
```

A SparkContext represents the connection to a Spark cluster, and can be used to create RDDs, accumulators and broadcast variables on that cluster.

```
sc = spark.sparkContext

lines = sc.textFile("/Input/test_file1.tsv")

parts = lines.map(lambda l: l.split("\t"))(we separated using tab space, as the columns had a lot of commas in it )

parts1 = parts.filter(lambda p:p!=None) (remove empty columns)

parts2 = parts1.map(lambda x:remove_stop(x,li)) (remove stop words)

schemaString = "parent_data Subreddit Filtered_data"(required dataframe columns)

fields = [StructField(field_name, StringType(), True) for field_name in schemaString.split()]

schema = StructType(fields)

# # Apply the schema to the RDD.

schemaPeople = spark.createDataFrame(parts2, schema)

yellow = schemaPeople.select("Filtered_data")(select only the non-stopwords parent comments column)

yellow.show()

tokenizer = Tokenizer(inputCol="Filtered_data", outputCol="words")(tokenize the sentences)

countTokens = udf(lambda words: len(words), IntegerType())(function to count the number of words in the sentence)

schemaPeople = tokenizer.transform(schemaPeople)

schemaPeople.select("Filtered_data", "words")\

    .withColumn("tokens", countTokens(col("words")))

hashingTF = HashingTF(inputCol="words", outputCol="rawFeatures", numFeatures=1500)

schemaPeople = hashingTF.transform(schemaPeople)(gets the term frequency)

idf = IDF(inputCol="rawFeatures", outputCol="features")(gets the inverse document frequency)

idfModel = idf.fit(schemaPeople)

rescaledData = idfModel.transform(schemaPeople)
```

rescaledData.select("Subreddit","features").show()

stringIndexer = StringIndexer(inputCol="Subreddit", outputCol="label"))(label encodes the subreddit classes)

model = stringIndexer.fit(rescaledData)

indexed = model.transform(rescaledData)

get_dict_label = list(indexed.toPandas()['label'])

get_dict_subreddit = list(indexed.toPandas()['Subreddit'])

new_df = indexed.select('label','features'))(creates a new dataframe with only the inputs ie the sentence vector and the subreddit label)

splits = new_df.randomSplit([0.6, 0.4], 1234))(split it into train test, 60-30 % split)

train = splits[0]

test = splits[1]

layers = [1500, 250, 184, 120])(define the number of neurons in each layer, input is 1500 neurons(the number of features) and output layer is of 120 neurons as we have 120 classes)

# create the trainer and set its parameters(100 iterations and a block size of 128)

trainer = MultilayerPerceptronClassifier(maxIter=100, layers=layers, blockSize=128, seed=1234)

# train the model

model = trainer.fit(train)

# try it for the test set

result = model.transform(test)

result.show(1600)

predictionAndLabels = result.select("prediction", "label")

#as spark doesn’t allow user input, we write our questions in a text file and submit to the model

questions = sc.textFile("/Input/Ap.txt")

le_name_mapping = {}

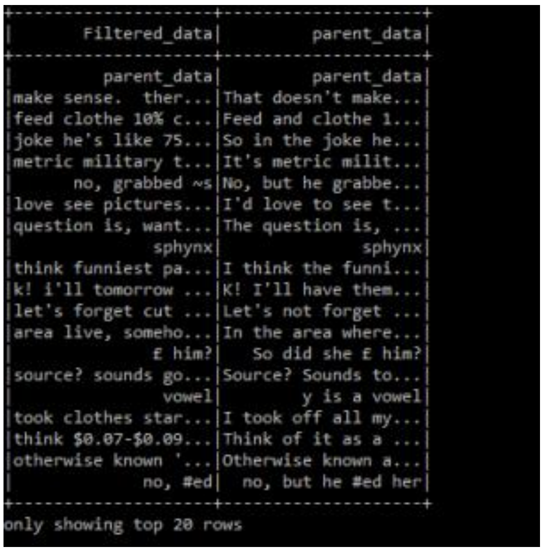
for i,j in zip(get_dict_label,get_dict_subreddit): (maps the subreddit class to its label)

```
    le_name_mapping[i] = j
```

```
path = "/Input/intents.json" (intents is a file where each subreddit class has a set of responses which we will output given a class)

intents = spark.read.json(path)

replies = []

sub = []

q= []

for i,j in enumerate(questions.collect()):

    pred = bag_of_words(j,hashingTF,idfModel,tokenizer,model)(pass each question to the model)

    q.append(j)

    o = output(le_name_mapping,pred,intents,spark)

    replies.append(o[0])

    sub.append(o[1])

for i,j,k in zip(replies,q,sub): (print the output along with its class)

    print(j+" : "+i+"--"+k+"--")

spark.stop()
```

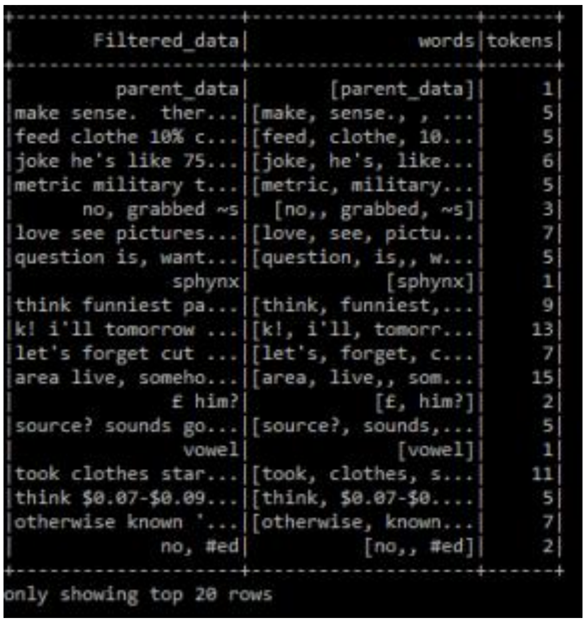
OUTPUTS

BEFORE AND AFTER REMOVING STOPWORDS



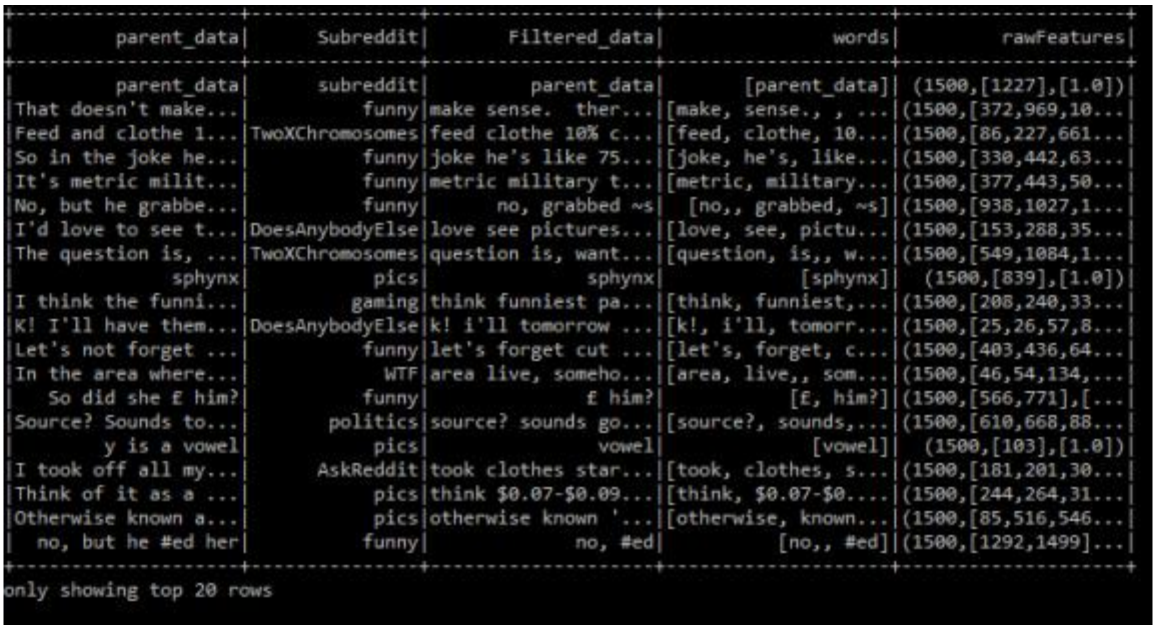
```
+--------------------+--------------------+
|       Filtered_data|         parent_data|
+--------------------+--------------------+
|         parent_data|         parent_data|
|make sense.  ther...|That doesn't make...|
|feed clothe 10% c...|Feed and clothe 1...|
|joke he's like 75...|So in the joke he...|
|metric military t...|It's metric milit...|
|      no, grabbed ~s|No, but he grabbe...|
|love see pictures...|I'd love to see t...|
|question is, want...|The question is, ...|
|              sphynx|              sphynx|
|think funniest pa...|I think the funni...|
|k! i'll tomorrow ...|K! I'll have them...|
|let's forget cut ...|Let's not forget ...|
|area live, someho...|In the area where...|
|              £ him?|   So did she £ him?|
|source? sounds go...|Source? Sounds to...|
|               vowel|         y is a vowel|
|took clothes star...|I took off all my...|
|think $0.07-$0.09...|Think of it as a ...|
|otherwise known '...|Otherwise known a...|
|           no, #ed|  no, but he #ed her|
+--------------------+--------------------+
only showing top 20 rows
```

AFTER TOKENIZATION



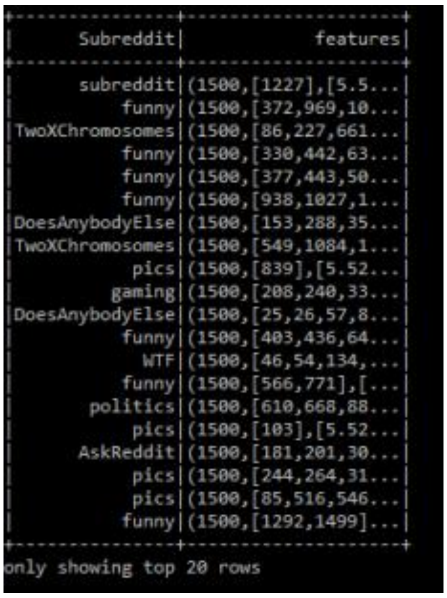
```
+--------------------+--------------------+------+
|       Filtered_data|               words|tokens|
+--------------------+--------------------+------+
|         parent_data|       [parent_data]|     1|
|make sense.  ther...|[make, sense., , ...|     5|
|feed clothe 10% c...|[feed, clothe, 10...|     5|
|joke he's like 75...|[joke, he's, like...|     6|
|metric military t...|[metric, military...|     5|
|     no, grabbed ~s|   [no,, grabbed, ~s]|     3|
|love see pictures...|[love, see, pictu...|     7|
|question is, want...|[question, is,, w...|     5|
|              sphynx|            [sphynx]|     1|
|think funniest pa...|[think, funniest,...|     9|
|k! i'll tomorrow ...|[k!, i'll, tomorr...|    13|
|let's forget cut ...|[let's, forget, c...|     7|
|area live, someho...|[area, live,, som...|    15|
|              £ him?|           [£, him?]|     2|
|source? sounds go...|[source?, sounds,...|     5|
|               vowel|             [vowel]|     1|
|took clothes star...|[took, clothes, s...|    11|
|think $0.07-$0.09...|[think, $0.07-$0....|     5|
|otherwise known '...|[otherwise, known...|     7|
|             no, #ed|          [no,, #ed]|     2|
+--------------------+--------------------+------+
only showing top 20 rows
```

AFTER VECTORIZING



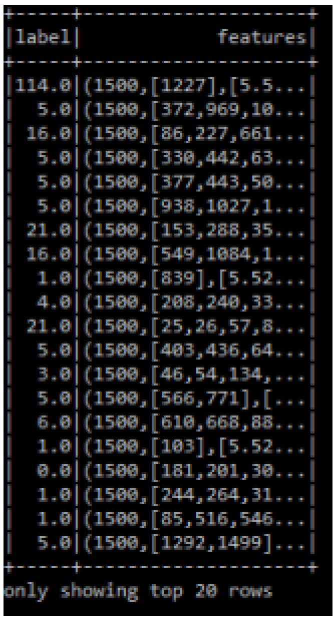
```
+--------------------+---------------+--------------------+--------------------+--------------------+
|         parent_data|      Subreddit|       Filtered_data|               words|         rawFeatures|
+--------------------+---------------+--------------------+--------------------+--------------------+
|         parent_data|      subreddit|         parent_data|       [parent_data]| (1500,[1227],[1.0])|
|That doesn't make...|          funny|make sense.  ther...|[make, sense., , ...|(1500,[372,969,10...|
|Feed and clothe 1...|TwoXChromosomes|feed clothe 10% c...|[feed, clothe, 10...|(1500,[86,227,661...|
|So in the joke he...|          funny|joke he's like 75...|[joke, he's, like...|(1500,[330,442,63...|
|It's metric milit...|          funny|metric military t...|[metric, military...|(1500,[377,443,50...|
|No, but he grabbe...|          funny|      no, grabbed ~s|  [no,, grabbed, ~s]|(1500,[938,1027,1...|
|I'd love to see t...|DoesAnybodyElse|love see pictures...|[love, see, pictu...|(1500,[153,288,35...|
|The question is, ...|TwoXChromosomes|question is, want...|[question, is,, w...|(1500,[549,1084,1...|
|              sphynx|           pics|              sphynx|            [sphynx]|  (1500,[839],[1.0])|
|I think the funni...|         gaming|think funniest pa...|[think, funniest,...|(1500,[208,240,33...|
|K! I'll have them...|DoesAnybodyElse|k! i'll tomorrow ...|[k!, i'll, tomorr...|(1500,[25,26,57,8...|
|Let's not forget ...|          funny|let's forget cut ...|[let's, forget, c...|(1500,[403,436,64...|
|In the area where...|            WTF|area live, someho...|[area, live,, som...|(1500,[46,54,134,...|
|   So did she £ him?|          funny|              £ him?|           [£, him?]|(1500,[566,771],[...|
|Source? Sounds to...|       politics|source? sounds go...|[source?, sounds,...|(1500,[610,668,88...|
|         y is a vowel|           pics|               vowel|             [vowel]|  (1500,[103],[1.0])|
|I took off all my...|      AskReddit|took clothes star...|[took, clothes, s...|(1500,[181,201,30...|
|Think of it as a ...|           pics|think $0.07-$0.09...|[think, $0.07-$0....|(1500,[244,264,31...|
|Otherwise known a...|           pics|otherwise known '...|[otherwise, known...|(1500,[85,516,546...|
|  no, but he #ed her|          funny|             no, #ed|          [no,, #ed]|(1500,[1292,1499]...|
+--------------------+---------------+--------------------+--------------------+--------------------+
only showing top 20 rows
```

SUBREDDIT CLASS AND PARENT COMMENT VECTORS



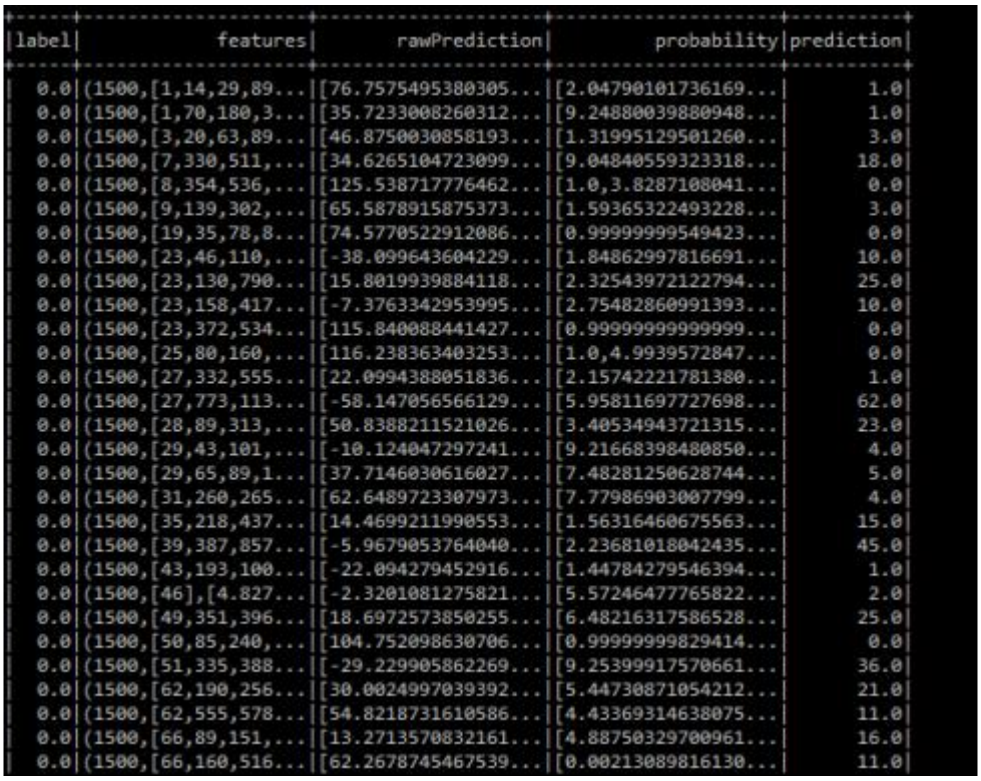
```
+---------------+--------------------+
|      Subreddit|            features|
+---------------+--------------------+
|      subreddit|(1500,[1227],[5.5...|
|          funny|(1500,[372,969,10...|
|TwoXChromosomes|(1500,[86,227,661...|
|          funny|(1500,[330,442,63...|
|          funny|(1500,[377,443,50...|
|          funny|(1500,[938,1027,1...|
|DoesAnybodyElse|(1500,[153,288,35...|
|TwoXChromosomes|(1500,[549,1084,1...|
|           pics|(1500,[839],[5.52...|
|         gaming|(1500,[208,240,33...|
|DoesAnybodyElse|(1500,[25,26,57,8...|
|          funny|(1500,[403,436,64...|
|            WTF|(1500,[46,54,134,...|
|          funny|(1500,[566,771],[...|
|       politics|(1500,[610,668,88...|
|           pics|(1500,[103],[5.52...|
|      AskReddit|(1500,[181,201,30...|
|           pics|(1500,[244,264,31...|
|           pics|(1500,[85,516,546...|
|          funny|(1500,[1292,1499]...|
+---------------+--------------------+
only showing top 20 rows
```

AFTER LABEL ENCODING THE SUBREDDIT LABELS



```
+-----+--------------------+
|label|            features|
+-----+--------------------+
|114.0|(1500,[1227],[5.5...|
|  5.0|(1500,[372,969,10...|
| 16.0|(1500,[86,227,661...|
|  5.0|(1500,[330,442,63...|
|  5.0|(1500,[377,443,50...|
|  5.0|(1500,[938,1027,1...|
| 21.0|(1500,[153,288,35...|
| 16.0|(1500,[549,1084,1...|
|  1.0|(1500,[839],[5.52...|
|  4.0|(1500,[208,240,33...|
| 21.0|(1500,[25,26,57,8...|
|  5.0|(1500,[403,436,64...|
|  3.0|(1500,[46,54,134,...|
|  5.0|(1500,[566,771],[...|
|  6.0|(1500,[610,668,88...|
|  1.0|(1500,[103],[5.52...|
|  0.0|(1500,[181,201,30...|
|  1.0|(1500,[244,264,31...|
|  1.0|(1500,[85,516,546...|
|  5.0|(1500,[1292,1499]...|
+-----+--------------------+
only showing top 20 rows
```

TESTING THE MODEL ON TEST SET

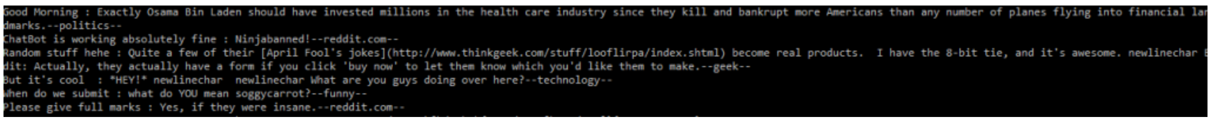


```
+-----+--------------------+--------------------+--------------------+----------+
|label|            features|       rawPrediction|         probability|prediction|
+-----+--------------------+--------------------+--------------------+----------+
|  0.0|(1500,[1,14,29,89...|[76.7575495380305...|[2.04790101736169...|       1.0|
|  0.0|(1500,[1,70,180,3...|[35.7233008260312...|[9.24880039880948...|       1.0|
|  0.0|(1500,[3,20,63,89...|[46.8750030858193...|[1.31995129501260...|       3.0|
|  0.0|(1500,[7,330,511,...|[34.6265104723099...|[9.04840559323318...|      18.0|
|  0.0|(1500,[8,354,536,...|[125.538717776462...|[1.0,3.8287108041...|       0.0|
|  0.0|(1500,[9,139,302,...|[65.5878915875373...|[1.59365322493228...|       3.0|
|  0.0|(1500,[19,35,78,8...|[74.5770522912086...|[0.99999999549423...|       0.0|
|  0.0|(1500,[23,46,110,...|[-38.099643604229...|[1.84862997816691...|      10.0|
|  0.0|(1500,[23,130,790...|[15.8019939884118...|[2.32543972122794...|      25.0|
|  0.0|(1500,[23,158,417...|[-7.3763342953995...|[2.75482860991393...|      10.0|
|  0.0|(1500,[23,372,534...|[115.840088441427...|[0.99999999999999...|       0.0|
|  0.0|(1500,[25,80,160,...|[116.238363403253...|[1.0,4.9939572847...|       0.0|
|  0.0|(1500,[27,332,555...|[22.0994388051836...|[2.15742221781380...|       1.0|
|  0.0|(1500,[27,773,113...|[-58.147056566129...|[5.95811697727698...|      62.0|
|  0.0|(1500,[28,89,313,...|[50.8388211521026...|[3.40534943721315...|      23.0|
|  0.0|(1500,[29,43,101,...|[-10.124047297241...|[9.21668398480850...|       4.0|
|  0.0|(1500,[29,65,89,1...|[37.7146030616027...|[7.48281250628744...|       5.0|
|  0.0|(1500,[31,260,265...|[62.6489723307973...|[7.77986903007799...|       4.0|
|  0.0|(1500,[35,218,437...|[14.4699211990553...|[1.56316460675563...|      15.0|
|  0.0|(1500,[39,387,857...|[-5.9679053764040...|[2.23681018042435...|      45.0|
|  0.0|(1500,[43,193,100...|[-22.094279452916...|[1.44784279546394...|       1.0|
|  0.0|(1500,[46],[4.827...|[-2.3201081275821...|[5.57246477765822...|       2.0|
|  0.0|(1500,[49,351,396...|[18.6972573850255...|[6.48216317586528...|      25.0|
|  0.0|(1500,[50,85,240,...|[104.752098630706...|[0.99999999829414...|       0.0|
|  0.0|(1500,[51,335,388...|[-29.229905862269...|[9.25399917570661...|      36.0|
|  0.0|(1500,[62,190,256...|[30.0024997039392...|[5.44730871054212...|      21.0|
|  0.0|(1500,[62,555,578...|[54.8218731610586...|[4.43369314638075...|      11.0|
|  0.0|(1500,[66,89,151,...|[13.2713570832161...|[4.88750329700961...|      16.0|
|  0.0|(1500,[66,160,516...|[62.2678745467539...|[0.00213089816130...|      11.0|
```

INTENTS

```
{"funny": ["It's metric military time, stupid American.", "I think this comment broke everything. ", "Wait, are you
kidding? newlinechar  newlinechar I hope so...", "So did she \u00a3 him?", "I saw that as well :)", "no, but he #ed her",
"He hashed her?", "It crashes Firefox 3.6 too :(", "Hey! Did you know that you can put a &gt; at the beginning of a line to
quote people? Like this: newlinechar  newlinechar &gt; Unless you built a radio with frequencies to listen to the emotional
detachment of polar bears", "What does the 'stat' part mean anyway? I know it means 'right now', but what's its
etymology?", "Should've used Opera or Chrome.", "Etymology 1 newlinechar  newlinechar From Latin statim
(\u201c\u2018immediately\u2019\u201d). newlinechar  newlinechar -from wiki", "That's also known as a pound symbol. At
least, it is in some to all of America. I don't know about the rest of the world.  newlinechar  newlinechar  newlinechar
newlinechar (Go on, make the ignorant American joke. You know you want to)", "Your sarcasm is as eloquent as it is loud.",
"IT IS IMPOSSIBLE FOR ME TO THANK YOU WITHOUT BETRAYING THE SPIRIT OF MY NOVELTY ACCOUNT", "...or the story takes place
some time in the past", "Do you seriously not know?", "Don't downvote this man. Some people don't know how to format, and
we all gotta learn somehow.", "^ newlinechar  newlinechar | newlinechar  newlinechar | newlinechar  newlinechar That's the
post of the day right there folks!~", "...or very right.", "Are you one of the writers from the show Lost?", "An
unfortunate talent. ", "SMALLER THAN A FACT BUT JUST AS FILLING.", "And that's presuming the dude had sex the same year he
was born.", "Sad, You beat Thraxil by 1 minute with the same message, but he's getting more upvotes. Well, have an
upvote.", "At worst Chrome will say 'yo, bro, this process is going slow, you wanna kill it?' and be done with it.", "No,
that's just a diet fact.", "Well... his penis *was* in a vagina.", "YOU'RE A DIET FACT.", "The world is a cruel place.",
"You should have just said 'THANKS'", "YOU CLEARLY HAVE A FIRM GRASP OF THE PURPOSE OF THIS NOVELTY ACCOUNT", "Reddit is a
cruel place", "So, actually, there are no virgins...  newlinechar  newlinechar and, if you're a 'virgin', the last girl you
'had' was your mother...", "Reddit is my world.", "So's your face.", "As an ignorant American joke enjoying American I
approve this request.", "Whoa man, don't make me lower my speakers.", "Reddit needs a bus like that.", "Here's some bleach.
```

RESULT



```
Good Morning : Exactly Osama Bin Laden should have invested millions in the health care industry since they kill and bankrupt more Americans than any number of planes flying into financial lar
dmarks.--politics--
ChatBot is working absolutely fine : Ninjabanned!--reddit.com--
Random stuff hehe : Quite a few of their [April Fool's jokes](http://www.thinkgeek.com/stuff/looflirpa/index.shtml) become real products.  I have the 8-bit tie, and it's awesome. newlinechar E
dit: Actually, they actually have a form if you click 'buy now' to let them know which you'd like them to make.--geek--
But it's cool  : *HEY!* newlinechar  newlinechar What are you guys doing over here?--technology--
When do we submit : what do YOU mean soggycarrot?--funny--
Please give full marks : Yes, if they were insane.--reddit.com--
```

FURTHER IMPROVEMENTS

The model we made was trained on only a part of the acquired dataset as storage and the computing power was limited. Moreover the dataset classes were not uniformly distributed hence we got very biased results. Currently the reply to a question is a random selection of a reply from the respective predicted class. Inorder to improve the selection of replies to the questions, a similarity check (cosine similarity) can be performed on sentence vectors obtained using TFIDF with all the responses belonging to that class. Moreover using Auto encoders to generate responses would provide even better results. Here instead of building from predefined responses, they are trained using a large number of previous conversations (questions and answers), based upon which responses to the questions are generated.