King Saud University
Collage of Computer Science and Information

SWE 485
2ND Semester Spring 2020
Final Report

Team #6		
Name	**ID**	Section
Asma AlJudaya	437201017	54132
Fatimah AlQahtani	437200517	54132
Shaden AlRomi	437200223	54132
Ghedaa AlAjaji	437202722	54132
Mona AlFayyadh	435202319	54132
Ruba AlSmail	437201821	54132

Supervisors: L.Bayan AlArifi / L.Mona Hakami

Phase 1: Data Collection

STC Company

Is a Saudi telecommunications company that founded on 21/4/1998 by the name" Saudi telecom company", It offers landline, mobile, internet service and computer networks for residential and business customers in Saudi Arabia as well as Kuwait and Bahrain.

Problem Description

STC fiber service is using the FTTH (Fiber to the home) Technology, which is replacing existing copper infrastructure such as telephone wires and coaxial cable with a newer network of high-speed fiber optic cables that are better able to deliver high-speed data across greater distances, resulting in much faster download speeds.

We found that customers who are using STC fiber service complaining about it, where customers claiming that STC not doing their job by providing what they are supposed to provide.

Objective

Our objective of this analysis is to find out what is the main problem of this service that users are facing, also we want to know if all customers who purchased this service are facing the same problem or not. If not, we want to know what the common factor between those customers who are facing this problem. And therefore, finding the root cause of this problem, which will then help us to design the solution of the problem, that will enhance the service reputation and make the customers have the best experience.

Process & Tools

We will use the Anaconda distribution platform which is The World's Most Popular Python/R Data Science Platform. Moving on to the tools Jupyter Notebook is the tool we have used. we have used Tweepy and GetOldTweets3 libraries to access the twitter API,

NumPy library to do mathematical and statical operations, Pandas library to do the data structure(tables).

Extracted Tweets Description:

We used two extraction tweets methods:

Method 1	
Tool used	Anaconda, Pandas, Tweepy and NumPy.
Duration	11/02/2020 - 22/02/2020
Keywords	”, الأليافstc,الالياف”, “stc,الياف”, “stc,الفايبر”, “stc,ربيافstc, “ ”اس تي سي,الياف” , “اس تي سي,الفايبر” , “اس تي سي ,فايبر” ,“أليافstc, “@stccare_ksa”)الالياف,اس تي سي“ ,
Number of tweets extracted	around 1750 tweets
Format for each tweet	(Tweet text, ID, Created date, Source, Favorite count, Retweet count)
File extension	.xls

Method 2	
Tool used	GetOldTweets3 0.0.11
Duration	27/09/2012 - 22/02/2020
Keywords	”, الأليافstc,الالياف”, “stc,الياف”, “stc,الفايبر”, “stc,ربيافstc, “ ”اس تي سي,الياف” , “اس تي سي,الفايبر” , “اس تي سي ,فايبر” ,“ أليافstc,)الالياف,اس تي سي“ ,
Number of tweets extracted	2134 tweets
Format for each tweet	(Tweet text, Created date, User name, To, Number of replies, Favorite count, Retweet count)
File extension	.csv

Phase 2: Data Preprocessing

Introduction:

After we read the dataset we noticed some tweets that uses the keyword (ألياف STC, ربياف STC الفايبر stc, الالياف اس تي سي, الياف اس تي سي, اس تي سي فايبر , الياف_stc), but it is irrelevant to the our subject. But also, we found that some of the customers are explaining their problem and others either curse or complain. Moreover, some of the customers tweets are actually demanding STC to provide the fiber service in their neighborhood or their cities since the service not available to them yet. furthermore, we found that STC lately launched a competition, where customers have contributed to it but there are no opinions about the service itself. The following table illustrate the issues and explanation about them:

Issue	Explain	How we will fix the issue
STC competitions	There were several tweets related to STC competitions using hashtags #انترنت_بيتي_لامحدود #فايبر_نص_ميجا #دنكيو_STC	Looping through the tweets which contains the word "انترنت بيتي لا محدود" and drop them using the following methods: data_df[data_df['Tweets'].str.contains('#دنكيو_STC |انترنت_بيتي_لامحدود_STC.|#ويكند#فايبر_نص_ميجا|#انترنت_بييت__stc#|فايبر_نص_ميجا#دودحملا', case=False)== False]
Zain/Mobily fiber service	There were several tweets related to other companies' fiber service.	Looping through the tweets which contains the word "Zain" or the word "Mobily " and drop them using the following methods: data_df[data_df['Tweets'].str.contains('zain|mobily|موبايلي|نيز', case=False)== False] then we print the result and check it manually to ensure all tweets are unrelated to our subject. ex: some tweets contains comparison between STC fiber service and other companies' fiber service.
Duplicated tweets	There were several duplicated tweets since we used two methods to extract data, hence each method may retrieve the same tweets.	data_df = data_df.drop_duplicates(subset='Tweets', keep='first')

Cleaning text	remove additional signs/characters.	looping through tweets and cleaning each one using the following method. tweet = re.sub(removeChar, ' ', tweet)
Advertisements	There were several tweets that are actually, advertisements but use the same keywords.	We fixed this issue manually because this kind of issue does not have a specific keyword, that we can use.

We also double checked the cleaned sheets manually to ensure that the data are actually cleansed.

With regard to the issues above, we had decided to drop the tweets that are considered as **STC Competitions** and **Advertisements** since they are not a real people opinion they are just bots and it will affect the accuracy of our analysis, same with **Duplicated tweets** it will also affect the accuracy since it's duplicated so what is the point of having duplicated tweets? We decided to drop them too, moving on to the tweets that are considered as **Zain/Mobily fiber service,** we drop it the because we need only opinions about STC fiber, not their Competitors.

Phase 3: Data Analysis and Modeling

Introduction:

At this stage we will describe the data descriptively and predictively using some models and tools," tool to help us classify the data into two different كجازمfor example we started with using the "classes "positive, negative". Which will help us in understanding the data more as well as using the
classes in our model.

Descriptive Analysis:

Most frequent word:

The following picture shows the most 10 frequent and the repetitions of the word in the analyzed tweets:

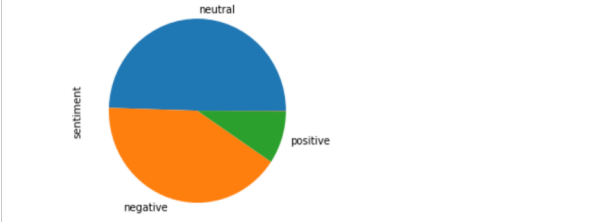
```
[50]: # show the most common words
      word_counter.most_common(10)

[50]: [('84 ,'حي),
       ('81 ,'المشكله),
       ('77 ,'الخدمه),
       ('71 ,'شركه),
       ('62 ,'موبايلي),
       ('61 ,'سرعه),
       ('56 ,'باقه),
       ('56 ,'زين),
       ('53 ,'الاتصالات),
       ('52 ,'السرعه),
```

From the above picture we can see the word since neighborhood takes the most repeated word, we expect that the neighborhoods may be facing some infrastructure issues hence why users tweeted about their neighborhood. Also, the word problem was mentioned a lot in the tweets where we assume the users are explaining the problems they were facing. Moreover, we can see the word speed which indicates that users my pain from the speed of the fiber service. some other words were mentioned such as Zain and Mobily, where we think users are comparing STC fiber service with those
companies.

Distribution of Sentiment:

The following picture shows the distribution of sentiment:

```
[78]: tweets_by_stance = data_df['sentiment'].value_counts()

[79]: # visualize the results
      tweets_by_stance.plot(kind='pie')

[79]: <matplotlib.axes._subplots.AxesSubplot at 0x24a3cb92308>
```



As you can see from the picture there is a significant problem since the negative part represents a large amount compared to the positive part which tells people opinions about the fiber service is negative.

Predictive Analysis:

Since the result of the model is a discrete variable either “positive” or “negative”, the logistic regression model “binary classification” was appropriate for us.

Applying the model:

1- Import the needed libraries and read and store transformed tweets in the “data” variable:

```
In [1]:
import os
import pandas as pd
from pandas import DataFrame, read_csv
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import cross_val_score, cross_val_predict, KFold
from sklearn.metrics  import confusion_matrix, classification_report
from sklearn import metrics


data = pd.read_excel(r'C:\Users\gheda\Desktop\clean.xlsx')
```

2- Remove the natural class (to focus on the negative and positive opinions of the customers)

```
In [6]:
# remove the "Neutral" class
data_f=data[data['sentiment'] != "neutral"]
```

3- In the pre-processing step, read the “Sentiment” column and convert the labels from strings to

binary values [“negative” = 0, “positive” = 1].

```
In [7]:
# change values to numeric
data_f['sentiment'] = data_f['sentiment'].map({'positive': 1, 'negative': 0})
```

4-Save the “tweets” column in a variable called **data** and the binary labeling “sentiment” column in the **target** variable.

```
In [8]:
# idneitfy the data and the labels

target= data_f['sentiment']
data = data_f['tweets']
```

5- Convert the data to numerical form

```
In [14]:
# Use TfidfVectorizer for feature extraction (TFIDF to convert textual data to numeric form):
tf_vec = TfidfVectorizer()
X = tf_vec.fit_transform(data)
X.shape

Out[14]: (1066, 4637)
```

6- Diagnose the model using holdout data technique, the data was split into two parts 0.7 for training and 0.3 for testing:

```
In [106]:
# Training Phase
X_train, X_test, y_train, y_test = train_test_split(X_sm, y_sm, test_size=0.30, random_state=0)
```

7-Make a prediction on test sample:

```
In [68]:
#Probability Threshold = 0.5 (default)
prob_test  = trained_model_lr.predict_proba(X_test)
pred_test  = trained_model_lr.predict(X_test)
```

8-Apply Logistic Regression model to the dataset in order to train it by simply calling [LogisticRegression()] which was imported in step 1

```
In [66]:
model_name = 'Logistic Regression'
logreg = LogisticRegression(C=0.5)
trained_model_lr = logreg.fit(X_train, y_train.values.ravel())
```

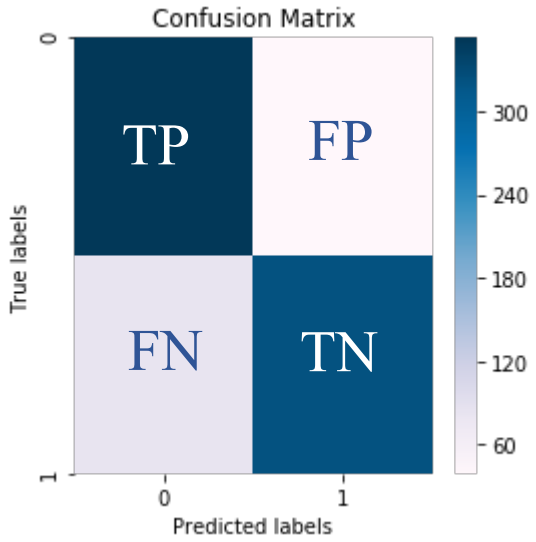
9-Display the result accuracy for both sample:

```
In [70]:
#Calculate train and test accuracy
train_acc = accuracy_score(y_train.values.ravel(), pred_train)
test_acc  = accuracy_score(y_test.values.ravel(), pred_test)
print ("\nTrain Accuracy :: ", train_acc)
print ("\nTest Accuracy :: ", test_acc)

Train Accuracy ::  0.9374217772215269

Test Accuracy ::  0.8460575719649562
```

10-. Calculate and display the results of the confusion matrix. The results of the confusion matrix and the value of (FPR, FNR, Precision, and Recall) were high, which means that the model classification have high accuracy.

- As you can see from the below figure that the color of true positive and true negative is dark blue describing high value where for the false positive and false negative is light pink color describing low value with that, we conclude that our model has high accuracy.

```
In [114]:
confusion_matrix(y_test, pred_test)

Out[114]:
array([[206,  32],
       [ 25, 217]], dtype=int64)

In [115]:
import pylab as pl
import seaborn as sns

cm = confusion_matrix(y_test, pred_test)
pl.matshow(cm)
ax= plt.subplot()
sns.heatmap(cm, ax = ax, cmap=plt.cm.PuBu); #annot=True to annotate cells

# labels, title and ticks
ax.set_xlabel('Predicted labels');ax.set_ylabel('True labels');
ax.set_title('Confusion Matrix');
```



- The below display that precision with 0.91 for negative 0.89 for positive and the recall 0.89 negative 0.92 positive and the accuracy is 0.90 all the previous value is high which prove that the accuracy is high for our model.

In [116]:
```
#Calculate classification model evaluation metrics like precision, recall, f1 score
report = classification_report(y_test, pred_test)
precision,recall,fscore,support = precision_recall_fscore_support(y_test,pred_test,average='weighted')
print("\n Classification report (weighted average across classes) ::\n", classification_report(y_test, pred_test))
```

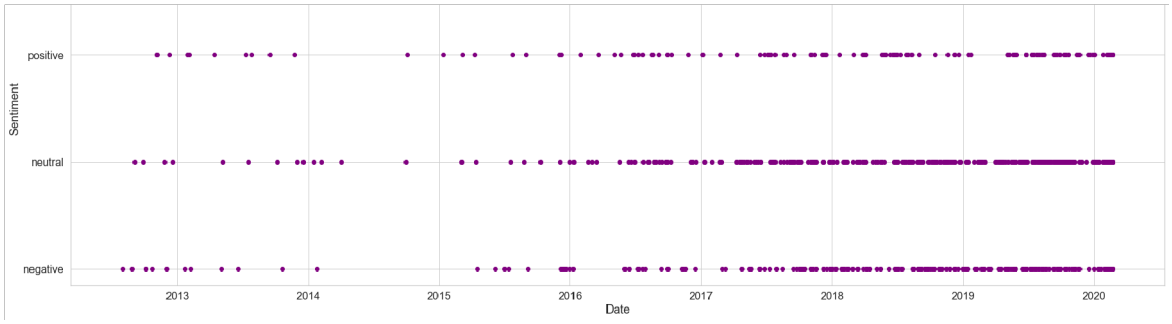
```
 Classification report (weighted average across classes) ::
              precision    recall  f1-score   support

         0.0       0.91      0.89      0.90       238
         1.0       0.89      0.92      0.90       242

    accuracy                           0.90       480
   macro avg       0.90      0.90      0.90       480
weighted avg       0.90      0.90      0.90       480
```

Phase 4: Data Visualization and Findings

Introduction:

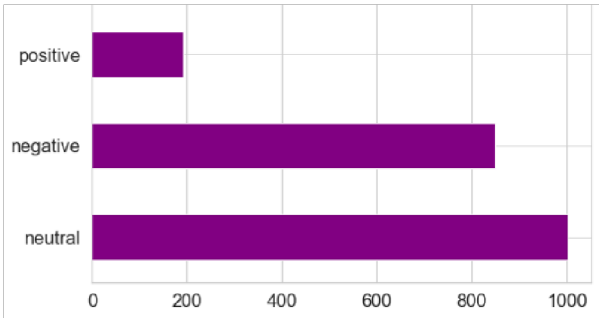
Data visualization is a quick, easy way to convey concepts in a universal manner. It gains insight into an information space by mapping data into graphical shapes. We will represent our findings in graphics to promote creative data exploration. In this document, we explained our data into charts and graphs to help us understand our data that we have collected as well as to help us to make the best decisions and give the write recommendations.

- **Figure One:**



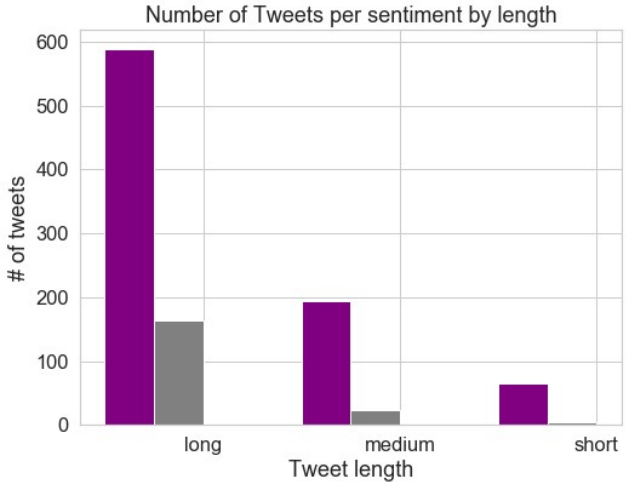
The figure shows as a time series analysis of the sentiments as you can see people started tweeting about STC fiber service in 2013, where in that year people opinions seem to be a few and have different opinions towards the service, however, we think that the reason they are a few is because it was a new service.
leading with the years from 2015 to 2016 we can see that the negative tweets and positive tweets are approximately equal, however at the end of 2016 STC decided to make an offer is given modem and installation for free as well as STC started to make more promotions for the service, hence why we think the tweets started to increase.
Although, the graph shows that with the increase of users using the service there is an increase in the negative tweets comparing to the positive tweets, after going through the tweets we found out that users suffer from the speed of the fiber STC which we think it indicates a problem in the infrastructure of the service.

- **Figure Two:**



The figure shows the number of tweets for each sentiment as you can see that neutral takes the most part, in which users did not necessarily complain or like the service, but they maybe have an inquiry or a question about the service.
the negative tweets take the second frequent part, which consists with information mentioned in the previous picture, in which we noticed that negative opinions represent a huge part of user's opinions and a few of them actually like the service.

- **Figure Three:**



The figure shows the length of the tweets in which the ***purple*** color represents negative tweets and the ***gray*** represents the positive tweets.

- more than 500 negative tweets considered as long and less than 200 positive tweets considered as long.
- about 200 negative tweets considered as medium and less than 100 positive tweets considered as medium.
- about 100 negative tweets or less considered as short and 0 positive tweets considered as short.

the above data shows to us that users write a lot when they writing something negative about the service however the opposite thing happen when they are writing something positive, which

is normal because when people like a service they do not go deep into the details but, when it comes to a problem they are facing they will write more about it, we noticed in the most of the negative tweets that they say that the fiber speed is an issue as well as they trying to fix the issue but they cannot.

- **Figure Four:**

AUC - ROC Curve it summarizes the performance of a binary classification model on the positive class. by shows the trade-offs between the true positive rate (TPR) and the false positive rate (FPR) given a test set and a model. The result of plotting (TPR & FPR) where the vertical axis represents the true positive rate and the horizontal axis represents the false positive rate is the area under the curve which is a measure of the model accuracy [14].
for our graph in the below figure it shows that the area under the curve 0.95 which is close to one and therefore consider as perfect accuracy and the prediction we conclude as accurate.

Recommendations:

- First STC should handle users' complaints properly in order to not destroy the reputation of the service.
- Second, Since we couldn't decide the root cause of the fiber speed problem form our dataset, there is no solution in regard to fiber speed.

Framework and Libraries:

- **Anaconda:** distribution platform which is The World's Most Popular Python/R Data Science Platform.
- **Jupyter Notebook** is the tool we have used.
- **Tweepy and GetOldTweets3 libraries:** to access the twitter API.
- **NumPy library:** to do mathematical and statical operations.
- **Pandas library:** to do the data structure(tables), remove duplicate tweets, and remove tweets that contains a specific string.
- **Re library:** for regular expression and to clean up the tweets from additional signs/characters we use re library.
- **Sklearn:** Rich library used for Regression, Classification, Clustering, Model selection And Preprocessing.
- **Matplotlib.pyplot:** provides a MATLAB-like way of plotting. pyplot is mainly intended for interactive plots.
- **Earthpy:** help to make working with temporal data and time series analysis.

Development files and Description:

File	Description
FULL_DATASET.csv	The original dataset from phase 1.
CLEANED_FULL_DATASET.csv	Contains the clean dataset after manipulations ، which the model applied to it.
phase2 cleaning.ipynb	Contains the code manipulations.
FINAL_DATASET.csv	Final dataset after manipulations.
phase3_predictive.ipynb	logistic regression model code
phas4.ipynb.ipynb	Code for generates figures (1,2 and 3) and visualize final dataset on phase 4.
regression_ph4.ipynb	Code for generates figure 4 on phase 4.

References:

[1] “What is FTTH: DiamondNet, OK - Official Website,” What is FTTH | DiamondNet, OK - Official Website. [Online]. Available: https://www.diamondnetok.com/128/What-is-FTTH. [Accessed: 22-Feb-2020].

[2] Stc.com.sa. (2020).stc-[online]Available at: الصفحة الرئيسية https://www.stc.com.sa/wps/wcm/connect/arabic/individual/individual [Accessed 19 Feb. 2020].

[3] Underconsideration.com. (2020). Push Forward. [online] Available at: https://www.underconsideration.com/brandnew/archives/new_logo_and_identity_for_stc_by _interbrand.php [Accessed 18 Feb. 2020].

[4] Pandas.pydata.org. 2020. Pandas.Series.Str.Contains — Pandas 1.0.2 Documentation. [online] Available at: <https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.Series.str.contains.html> [Accessed 15 March 2020].

[5] Pandas.pydata.org. 2020. Pandas.Dataframe.Drop_Duplicates — Pandas 1.0.2 Documentation. [online] Available at: <https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.drop_duplicates.html?highlight=drop_duplicates#pandas.DataFrame.drop_duplicates> [Accessed 15 March 2020].

[6] Pandas.pydata.org. 2020. Pandas.Dataframe.Drop — Pandas 1.0.2 Documentation. [online] Available at: <https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.drop.html?highlight=drop#pandas.DataFrame.drop> [Accessed 15 March 2020].

[7] Drive.google.com. 2020. Sharedfolder – Google Drive. [online] Available at: <https://drive.google.com/open?id=1ANnepiUMumMoUZYZ5kFNEOrQOsr3GBuo> [Accessed 15 March 2020].

[8] Drive.google.com. 2020. Sharedfolder – Google Drive. [online] Available at: <https://drive.google.com/open?id=1ANnepiUMumMoUZYZ5kFNEOrQOsr3GBuo> [Accessed 15 March 2020].

[9] GitHub. 2020. Advaitsave/Churn-Classification-Model-Selection. [online] Available at: <https://github.com/advaitsave/Churn-Classification-Model-Selection/blob/master/Code/Churn%20Classification%20and%20Model%20Selection%20in%20Python.ipynb> [Accessed 1 April 2020].

[10] Brownlee, J., 2020. ROC Curves And Precision-Recall Curves For Imbalanced Classification. [online] Machine Learning Mastery. Available at: <https://machinelearningmastery.com/roc-curvesand-precision-recall-curves-for-imbalanced-classification/> [Accessed 19 April 2020].

[11] Matplotlib.org. 2020. Matplotlib.Pyplot — Matplotlib 3.1.2 Documentation. [online] Available at: <https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.html> [Accessed 19 April 2020].

[12] Earth Data Science - Earth Lab. 2020. Work With Datetime Format In Python - Time Series Data. [online] Available at: <https://www.earthdatascience.org/courses/use-data-open-source-python/usetime-series-data-in-python/date-time-types-in-pandas-python/> [Accessed 19 April 2020].

[13] PyPI. 2020. Earthpy. [online] Available at: <https://pypi.org/project/earthpy/> [Accessed 19 April 2020].

[14] John Wiley & Sons, Inc, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, 2015. [E-book] Available: Google e-book.