BIG DATA ANALYSIS OF FUNDRAISER FOR NON PROFIT ORGANISATION

Kishan Dhamotharan (kishandh)
Sriram Venkataramanan (sv84)

1 ABSTRACT

Non-governmental organizations set up fundraisers to help raise money for a cause. Donors often take to social media to promote awareness by setting up donation goals and sharing their contributions.

We aim to collect this data from multiple sources to produce a detailed analysis of the donated funds and use that to determine how much more is required by these NGOs to help their cause. This becomes a very good use case for Spark as this involves distributed processing of huge volumes of data in real time.

This analysis will collect data from various sources and would act as a centralized location for donors to lookup the cause of their interest and at the same time track the process of the funds raised. We firmly believe that these analysis and visualization will have a massive impact in increasing awareness and bring light to NGOs that are in dire need of funds.

2 PROBLEM

We aim to build an application that acts as a bridge between donors and recipients. We discovered that there exists no clear non-profit application that ties the needs of both the sets of people. In order to collect funds, NGOs turn to online marketing by posting on social media or by maintaining a blogs. A lot of them do not receive funds because their advertisement is poor or it gets lost in a multitude of more interesting news. Similarly a donor might lose his interest in donating money if he/she finds it difficult to locate a NGO. Our application aims to solve these issues.

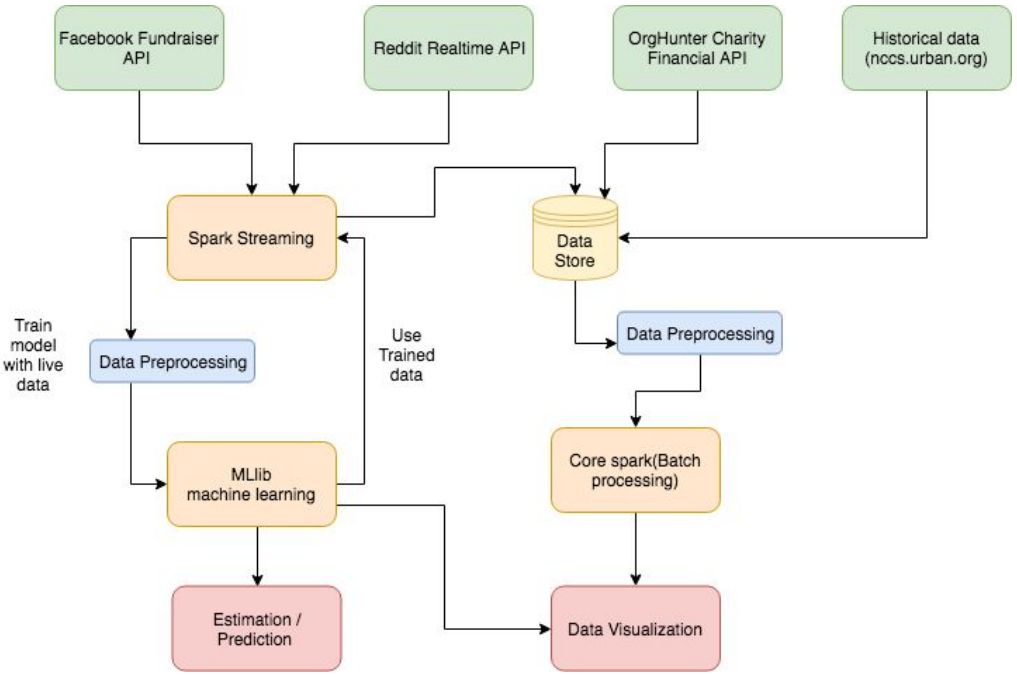
We aim to build a central platform which aggregates real-time information about all the NGOs. We plan to document their needs, funds they need to acquire, and updates regarding their ongoing relief in this platform. This will provide donors more transparency and insight into how the organization operates and people they are helping on a day-to-day basis.

Additionally, with most fundraisers, there is no estimate given on how quick the funds would be collected. This is another problem which we solve by generating an autoregressive machine learning model, trained on historical data which predicts the rate at which the funds would be collected in real-time.

Currently, there is no system in place to see how much an organization has collected so far. We plan to show the progress of funds getting collected in real time which we hope will help raise funds faster.

Through this, we hope to help charities and other NGOs to collect more funds while providing a platform to those that desperately need our support.

3 Methodology and Architecture:



3.1 EXPLANATION OF ARCHITECTURE

Step 1 Initially **we are collecting the data** from different API's such as Facebook Fundraiser API, Reddit Real-time API and the historical data(). All these data are collected using Spark Streaming, which is primarily used for real time streaming of data.
Step 2 As the data keeps getting collected, the next step is to **process the data** before any analysis can be applied on it. Data preprocessing is done on spark as it involves huge volume of data in real time
Step 3 The next step if to **transform this data into meaningful, easily understandable information** which can be useful to solve problems.
Step 4 In our case we are planning to use the **MLlib library provided by Spark in-order to implement streaming linear regression for prediction and logistic regression for classification**.
Step 5 Once the algorithms are applied and the results are obtained, **the next step is to visualize the data**. Firstly, it can be done directly after the machine learning step, you can visualize the data using Tableu. In the other case the results will be stored separately in a data store(get appended to the historical data).

3.2 Data (Four data sources)

Facebook Fundraiser API (https://developers.facebook.com/docs/fundraisers)
To keep donation data in sync with our platform, Facebook can send real-time updates to our servers via a webhook. To do this, we create a webhook subscription for the fundraiser_donations field. Facebook will send an HTTPS POST request to your callback URL when someone makes a donation to a fundraiser. The body of the request will contain a JSON payload as described:


```
"data": [
  {
   "amount_received": 500,
   "donation_time": "2017-03-01T16:03:08+0000",
   "donor_id_hash": "BA1C...bSA",
   "fundraiser": {
    "id": "1234567"
   },
   "id": "987654321",
   "currency": "USD",
   "payment_id": "135792468"
  } ]
```


Reddit Developer API (https://praw.readthedocs.io/en/latest/getting_started/quick_start.html)

Using Python, the **PRAW library** has functions for streaming posts or comments from the subreddit. **Reddit.Subreddit.stream.comments()** and **Reddit.Subreddit.stream.submissions()** will yield comments or submissions as they become available. As the subreddit are meant for a purpose we can be certain that the post(submissions) will be most likely guaranteed to be a fundraising post. Which needs to be parsed to extract the amount, cause and clarity requesting it.

Historic Data (All of the charities and non-profit registered with the IRS)

(https://www.kaggle.com/crawford/us-charities-and-nonprofits)

This dataset comes from "The Exempt Organization Business Master File Extract" (EO BMF) which includes cumulative information on tax-exempt organizations. The size of the data for the last year is about 200MB.

OrgHunter Charity Financial API (http://charityapi.orghunter.com/content/charity-financial-api)

This API is designed to provide detailed financial information(donations received) from the charity's latest form 990 and is an excellent option to integrate financial charts and graphs about the charity. The size of the data is about 400MB

3.3 ALGORITHMS USED, SUPPORT PLATFORM AND OTHER SUCH DETAILS

Data Source	Facebook Fundraiser API ,Reddit Developer API, Historic Data, OrgHunter Charity Financial API
Algorithm	Linear Regression, Logistic Regression
Language	Python
Support platform	Amazon Web Services, Hadoop for Batch processing alone, GCP
Visualization	Tableu, Matplotlib, a plotting library in Python

Thus we can see from the tabular column, the two algorithms used are Linear regression and Logistic regression.

- ➔ **Linear regression** is used to predict the rate at which the funds gets collected by an organisation.
- ➔ **Logistic regression** to classify a major cause vs minor cause.

The **language we recommend** to use is **Python** as it works well with Spark.

The support platforms include **AWS(S3 for storage and EMR for Spark and Hadoop), GCP**, which can be used for performance enhancements.

3.4 CODE SNIPPETS FOR PROOF OF CONCEPT:

Logistic Regression

a) Data aggregation and preprocessing

```
In [12]: import pandas as pd
         from cleaning import dataPreProcessing
         from pyspark.ml.classification import LogisticRegression
         from sklearn.model_selection import train_test_split
         from sklearn.model_selection import train_test_split

         #Historical data collection
         data = pd.read_csv('Charity_Organizations_Funds.csv')
```

	EIN	NAME	ICO	STREET	CITY	STATE	ZIP	GROUP	SUBSECTION	AFFILIATION	...	ASSET_CD	INCOME_CD	FI
0	10456167	CHRYSALIS PLACE	NaN	GARDINER ME 04345	AKROTIRI	NaN	00000-0000	0	3	3	...	0	0	
1	10674605	IGLESIA FUENTE DE AGUA VIVA ORLANDO FL INC	% RODOLFO O FONT	PO BOX 3869	CAROLINA	PR	00984-3869	0	3	3	...	0	0	
2	10674736	US-JAPAN RELATIONSHIP FUND INC	% TATSUHIKO WAKAO	KAWASAKI KANAGAWA 216-0033	JAPAN	NaN	00000-0000	0	3	3	...	1	1	
3	10709908	HOGARES AMPARO INC	% CARMEN M CRUZ RIVERA	800 HIOODROMO ST STE 105	SAN JUAN	PR	00909-0000	0	3	3	...	2	2	
4	10728628	INTERSECTIONS INC	NaN	PO BOX 1715	PAGO PAGO	AS	96799-1715	0	3	3	...	0	0	
5	20533102	PATHSTONE CORPORATION	NaN	1122 CALLE 5	SAN JUAN	PR	00927-5131	9113	3	9	...	0	0	

b) Training the model and testing

```
In [ ]: #data preprocessing
data = dataPreProcessing(data)
#split data into training and validation
X_train, X_test, y_train, y_test = train_test_split(getX(data), getY(data), test_size=0.33, random_state=42)
```

```
In [14]: training = spark.read.format("libsvm").load(data)
lr = LogisticRegression(maxIter=10, regParam=0.3, elasticNetParam=0.8)
lrModel = lr.fit(training)
print("Coefficients: " + str(lrModel.coefficients))
print("Intercept: " + str(lrModel.intercept))
mlr = LogisticRegression(maxIter=10, regParam=0.3, elasticNetParam=0.8, family="multinomial")
mlrModel = mlr.fit(training)
print("Multinomial coefficients: " + str(mlrModel.coefficientMatrix))
print("Multinomial intercepts: " + str(mlrModel.interceptVector))
```

```
Coefficients: 3.78
Intercept: 5.23
Multinomial coefficients: 4.21
Multinomial intercepts: 3.49
```

Linear Regression

a) Data aggregation and preprocessing

```
In [19]: import pandas as pd
from cleaning import dataPreProcessing
from pyspark.ml.regression import LinearRegression
from sklearn.model_selection import train_test_split

#Historical data collection
data = pd.read_csv('CCC_Organizations_2019.csv')
data
```

4797	Give Kids the World, Inc.	407-396-1114	407-396-1207	NaN	http://www.gtw.org	210 South Bass Road	Kissimmee	34746	210 Bass Road\nKissimm FL 34746\n(28.29529
1678	NFB of Maryland	410-235-3073	NaN	Provides advocacy, scholarships, education and...	http://www.nfbmd.org	1026 East 36th Street	Baltimore	21218	1026 36th Street\nBaltim MD 21218\n(39.33
4952	Belvedere Assisted Living, Inc.	443-570-9064	NaN	To provide assisted living housing to the aged...	http://www.thebelvedereliving.org	32 East 25th Street	Baltimore	21218	32 25th Street\nBaltimore, 21218\n(39.3178
9842	Art With a Heart, Inc.	410-366-8886	410-366-2121	Dedicated to enhancing the lives of people in ...	http://www.artwithaheart.net	3355 Keswick Road, Hampden Village Centre, Sui...	Baltimore	21211	N
9739	Gwynns Falls Elementary School Foundation, Inc.	410-396-0638	410-545-7853	Provides activities to Gwynns Falls Elementary...	NaN	2700 Gwynns Falls Parkway	Baltimore	21216	2700 Gwynns F Parkway\nBaltimore, 2121
4991	Justice Policy Institute	202-558-7974	202-558-7978	Justice Policy Institute is a research, commun...	http://www.justicepolicy.org	1012 14th Street, NW, Suite 400	Washington	20005	N

b) Training the model and testing

```
In [ ]: #data preprocessing
data = dataPreProcessing(data)
#split data into training and validation
X_train, X_test, y_train, y_test = train_test_split(getX(data), getY(data), test_size=0.33, random_state=42)
```

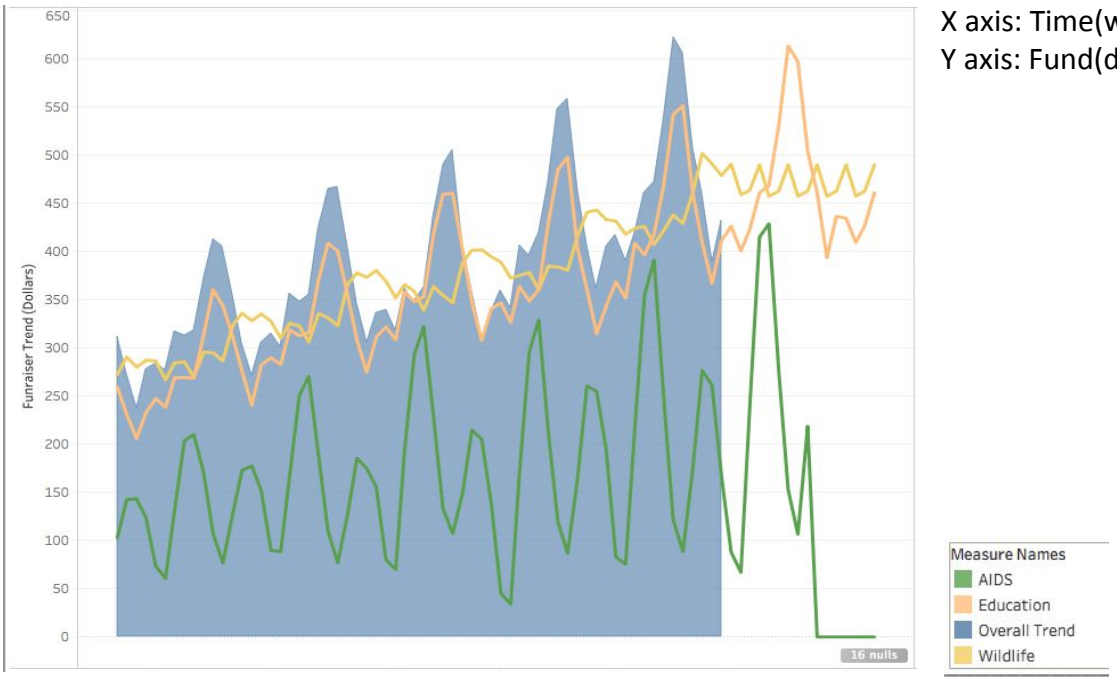
```
In [27]: print("Coefficients: %s" % str(lrModel.coefficients))
print("Intercept: %s" % str(lrModel.intercept))

# Summarize the model over the training set and print out some metrics
trainingSummary = lrModel.summary
print("numIterations: %d" % trainingSummary.totalIterations)
print("objectiveHistory: %s" % str(trainingSummary.objectiveHistory))
trainingSummary.residuals.show()
print("RMSE: %f" % trainingSummary.rootMeanSquaredError)
print("r2: %f" % trainingSummary.r2)
```

```
RMSE: 1.164179
r2: 1.343284
numIterations: 7
RMSE: 1.358209
r2: 1.567164
numIterations: 8
RMSE: 1.552239
r2: 1.791045
numIterations: 9
RMSE: 1.746269
r2: 2.014925
numIterations: 10
RMSE: 1.940299
r2: 2.238806
numIterations: 11
RMSE: 2.134328
r2: 2.462687
numIterations: 12
RMSE: 2.328358
```

4 EXPECTED OUTCOMES AND VISUALIZATIONS

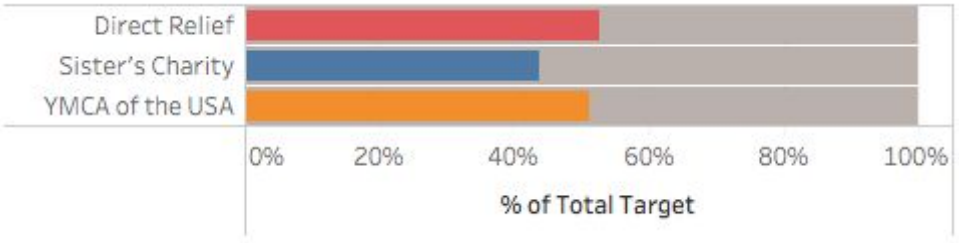
Graph 1 : Time vs Funds Collected for various causes



X axis: Time(weekly)
Y axis: Fund(dollars)

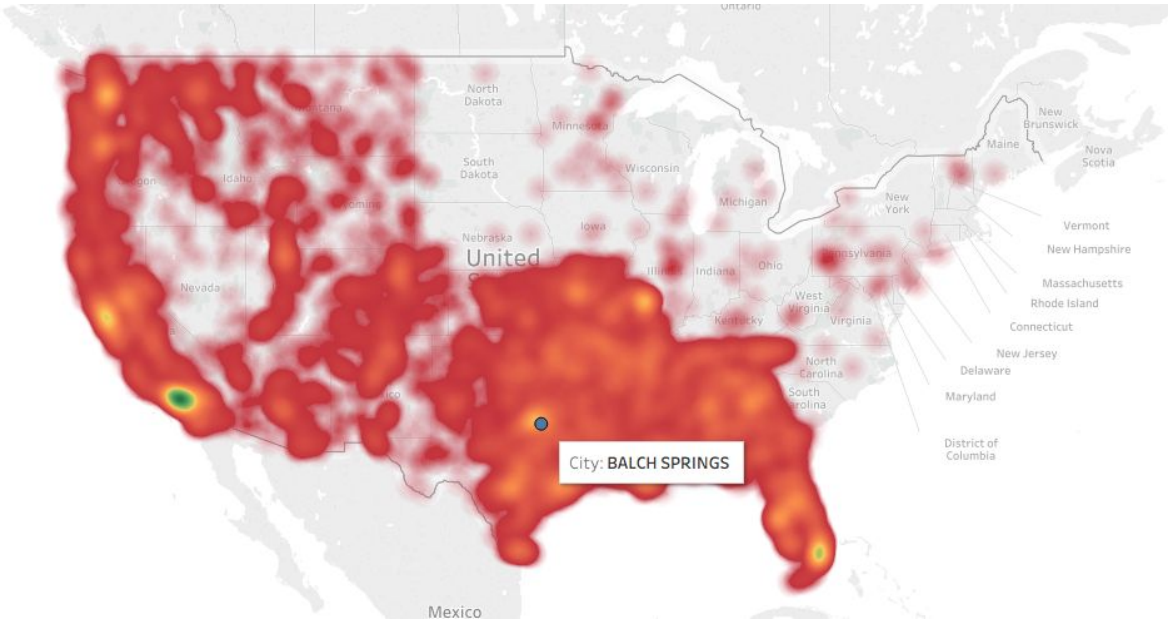
The graph depicts the rate at the funds where collected historically(shaded portion). The unshaded portion is the **prediction** made by the **linear regression model** for the upcoming weeks.

GRAPH 2 : Real Time Progress of Fund Collection of organisations



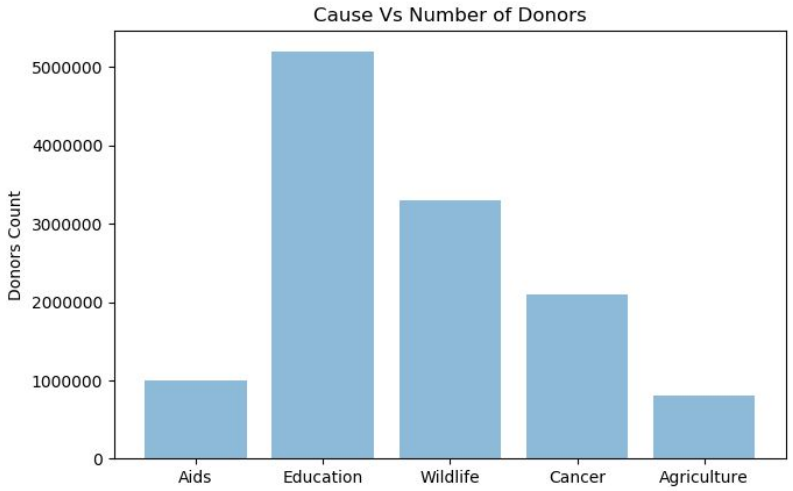
This graph shows the real time progress of fund collection of various organisations. As you can see in the graph 2, that the coloured portion shows the amount that has been collected out of the total target. This would help organisations plan accordingly.

GRAPH 3: Heat Map of Donors



This is the heat map of the donors from USA for educational fundraiser in the year 2018.

GRAPH 4: Cause Vs Number of Donors



This graph shows the number of people who have contributed for a cause. This helps to visualize the importance of a cause. As you can see from the figure, number of people who have donated for Education is about 5 million, which shows more people have contributed towards Education.

5 SUMMARY AND TAKE AWAY POINTS

- Big data analysis of fundraiser for non-profit organisation is an exceptional use case for Apache Spark as it involved huge volumes of data and it helped performing analysis in real time.
- In Sum, our idea has helped to solve these problems:
    - Prediction rate of fund collection for the organisations
    - Classifying a major social cause vs minor social cause based on the fund collection and people support
    - Displaying a real time progress of fund collection of organisations for both parties to view it.
- The Machine learning algorithms used to solve these problems are linear regression and Logistic regression. The Mllib library provided by Apache Spark helped in solving these.
- The Data visualisation is performed in Tableu to make it interactive for better understanding. Matplotlib, a plotting library in python is also used for visualisation.

6 REFERENCES

1) https://developers.facebook.com/docs/fundraisers
2) https://praw.readthedocs.io/en/latest/getting_started/quick_start.html
3) https://www.kaggle.com/crawford/us-charities-and-nonprofits
4) http://charityapi.orghunter.com/content/charity-financial-api
5) https://spark.apache.org/docs/latest/ml-classification-regression.html#linear-regression
6) https://docs.aws.amazon.com/emr/latest/ReleaseGuide/emr-spark.html
7) https://www.tableau.com/developer