DonorsChoose Recommendation Engine – Final Paper

DonorsChoose is a crowdsourcing website that utilizes its networking strength to bring public to the public schools. The goal of this organization is to connect donors from all over the world to the numerous projects organized which they support through donations. It also aims at strengthening the donor-project connection and maximizing the number and amount of donations at the same time. Public school teachers post their projects and project requirements on DonorsChoose.org. Public can make donations to the specific projects. Once the donation goal is reached, DonorsChoose procure the listed material and ship it directly to the school.

This team project attempts to build a sophisticated and comprehensive recommendation system which when deployed will help DonorsChoose to target their existing donor base with the most relevant projects so as to achieve a high donation rate as well as attract new donors and increase their conversion rate in the long run.

The data sources employed for this project fall into one of the following categories:

1. Operations Data – This includes data on classroom materials, equipment and human resources. Inventory data indicates the quantity of items to be procured.
2. Logistics Data – This encompasses delivery and supply chain data specifically delivery of equipment, raw materials etc. to the schools.
3. Financial Data – Data related to donations and expenditures is recorded as financial data to gauge the performance of the project. This data would enable the organizers to employ measures if the necessary funds are not able to be raised for a project.

The data required for building a recommender system can be collected from a number of sources few of which are listed below:

1. **Surveys**
   An effective way of gathering both objective as well as detailed information are surveys. The organization creates and requires existing donors as well as new donors to fill the surveys before donations. They may also send out surveys to existing donors from time to time in order to record and gauge their (changing) interests. They may also direct surveys through their email campaigns to potential donors which they think are likely to make donations to their projects. This mode of data collection attempts to capture basic donor details as well as details like hobbies, interests, inclination or attitude towards social service etc.
2. **Interviews with school representatives**
   This activity is intended to gather information regarding needs of the classrooms in particular from teachers, principals or other representatives. These are often detailed sessions where the representatives would elaborate on their exact needs, purpose of requesting items, end goal etc. Such information is useful in attracting donors who would resonate with these requirements and would actively donate towards to these projects.
3. **Social Media**
   This is another avenue which would give us ‘unfiltered’ information about donors. Surveys only capture the necessary basic details about the donors but social media platforms like Facebook, Twitter etc. throw light on the personal aspect of the donors. This might reveal information (such as what areas they feel very strongly about, places

they visit frequently etc.) which surveys might not be able to uncover and might prove useful in targeting additional projects towards them.
For the purpose of this POC, we will not be using Social Media as a data source but in future, we plan to make use of this channel as it will help improve our recommendation system and target donors better.

4. **Ledger Data**
   The amount of money raised for projects, expenditure for delivery vehicles and other logistics also needs to be collected in order for the initiative to be self-sustaining. This data needs to be tracked and any anomalies need to be scrutinized. At rare occasions, DonorsChoose needs to pay the amount from its operative funds. This will contain detailed financial transaction level data for each donation.

Data from different sources is already being stored in different relational databases by DonorsChoose. Using Apache Sqoop, the data from different sources is extracted, transformed into CSV files and loaded into HDFS.

Why Apache Sqoop?

We have chosen Apache Sqoop as the ETL tool for our project as it is specifically built to extract data from RDBMS and load to HDFS. Sqoop can execute the data transfer in parallel resulting in quick and cost-effective execution. Our dataset comes from multiple data sources (ranging from 73,000 to 7.2 million records) which makes parallel data processing necessary. Moreover, Sqoop works well with unstructured data also. Going forward, we will be including social media data for improving our recommendation system. At that point, Sqoop will come in handy and we would not need to look for other ETL tools to accommodate this change in the nature of data sources. Lastly, the Finance Management System database will contain detailed information about all the donations received. When a new donation is received, the earlier donation data need not be loaded again into HDFS Having to load the whole table each time there is a new donation will be computationally expensive. Therefore, using Sqoop will allow us to perform incremental loads of this data whenever the table is updated. This holds true for projects related data also.

Why HDFS?

HDFS utilizes commodity hardware which is very cheap and can be used with no licensing and support costs. This greatly reduces the storage costs for the company. Also, HDFS has the capability to transfer data to the computing nodes of the Hadoop cluster at a very high speed (exceeding 2 gigabits per second). This is beneficial for the amount of data that we will be using. HDFS (which is also the default storage mechanism for Hadoop) ensures data availability by replicating each block of data (entire data is divided into a number of blocks). By default, it makes 3 replicas of each block but this number can be increased based on needs. Since computation of large amounts of data can be expensive, ensuring fault-tolerance will reduce the probability of losing data or processing results to a great extent.

Extraction from data sources

The data that we are collecting from the above-mentioned data sources is being stored in different kinds of relational databases. In order perform any kind of cleaning and analysis, we

need to gather this data in CSV format at a single storage location which is HDFS (Hadoop Distributed File System) in our case.

The information that we will be capturing from each of these data sources is listed below:

Data Source	Information Captured
Surveys	Surveys will be used to capture details about the donors such as Name, State, City, ZIP etc. These are required to build a basic profile of the donor which will be stored in the system and shared with school projects for sending gratitude notes.
Interviews	Interviews with teachers will provide details about the projects that they want to be funded like project type/category, need statement, short description, an essay justifying the need for it to be funded and what purpose will it solve once the funds are raised and type of resources required. Information about the schools (like name, location, school metro – urban/suburban/rural, grade level of the students etc.) in which the project will be organized and the teachers who will organize the project (like teacher name and primary and secondary focus areas) will also be gathered from interviews.
Ledger Data	• All information regarding the donations will be gathered from the finance management system of DonorsChoose. Some examples of this information are: vendor shipping charges, sales tax, fulfillment labour materials, payment processing charges, optional donations. • Information about the donors and donations based on a project that have been received will also be collected from ledger data. This will mainly include details like total donations received, number of donors who donated, project status, funding status, date posted, date completed, project expiry date etc.

Data Cleaning and Transformation

The first step for data cleaning is to structure the data in a way that all related data is stored at a single location and segregated from rest of the data.
The data from all the above sources will need to be exported into CSV files from the relational databases supporting the respective front-end systems. However, this will result in separate files being generated from each of the systems. For example, the ledger data will be exported into a single file, project, schools and teachers related data will be obtained in another file and so on. We need to segregate the data based on its nature. That is, schools related data, projects data, donors and donations data, each should be stored in separate files. As part of the ETL process, we will perform the following transformations:

1. Extra data fields which would not contribute towards the recommendation system or metrics will be removed. Examples of such fields are: school latitude and longitude, district, county, poverty level, donation payment method etc.
2. For our purposes, we are not concerned with the breakdown of charges for a project. We are only concerned about the total cost of the project and how much donations were raised for it. Therefore, we will be combining the variables vendor shipping charges,

sales tax, fulfillment labor materials, payment processing charges, optional donations into a single variable called 'Project_Cost'.

3. Information regarding each of the resources (like unit price, quantity etc.) will be stored in a separate file as these are different from the total price calculated above. Although the total resource price will be included in calculating the total price, this file will give the breakdown of price, quantity to purchase of each resource. Since this detailed breakdown is not needed for the recommendation system, it will be stored separately for potential future use or metrics calculation.
4. The field names in our dataset consist of spaces. This makes the processing of data difficult. Hence, we will be replacing the spaces with underscore in order to maintain consistency and error-free data processing.

Dataset Overview

After extracting, cleaning and transforming the data, our entire dataset comprises of six CSV files. A brief description of each of these files is given below:

CSV File	Fields	Number of Rows
Donations	Project_ID, Donation_ID, Donor_ID, Donation_Included_Optional_Donation, Donation_Amount, Donor_Cart_Sequence, Donation_Received_Date	4.7 million
Donors	Donor_ID, Donor_City, Donor_State, Donor_Is_Teacher, Donor_Zip	2.1 million
Projects	Project_ID, School_ID, Teacher_ID, Teacher_Project_Posted_Sequence, Project_Type, Project_Title, Project_Essay, Project_Short_Description, Project_Need_Statement, Project_Subject_Category_Tree, Project_Subject_Subcategory_Tree, Project_Grade_Level_Category, Project_Resource_Category, Project_Cost, Project_Posted_Date, Project_Expiration_Date, Project_Current_Status, Project_Fully_Funded_Date	1.1 million
Resources	Project_ID, Resource_Item_Name, Resource_Quantity, Resource_Unit_Price, Resource_Vendor_Name	7.2 million
Schools	School_ID, School_Name, School_Metro_Type, School_Percentage_Free_Lunch, School_State, School_Zip, School_City, School_County, School_District	73,000
Teachers	Teacher_ID, Teacher_Prefix,	402,900

	Teacher_First_ Project_Posted_Date	

We plan to use the following features of the dataset to build our recommendation system. We will also look at increasing or decreasing the number of features depending on their relevance and accuracy of the recommendation engine. We will include more details on this in the paper on Data Analysis and Visualization.

Text Based Features

1. Project_Title
2. Project_Essay

Non-Text Based Features

1. Project_Subject_Category_Tree
2. Project_Subject_Subcategory_Tree
3. Project_Grade_Level_Category
4. Project_Resource_Category
5. School_State
6. Teacher_Prefix

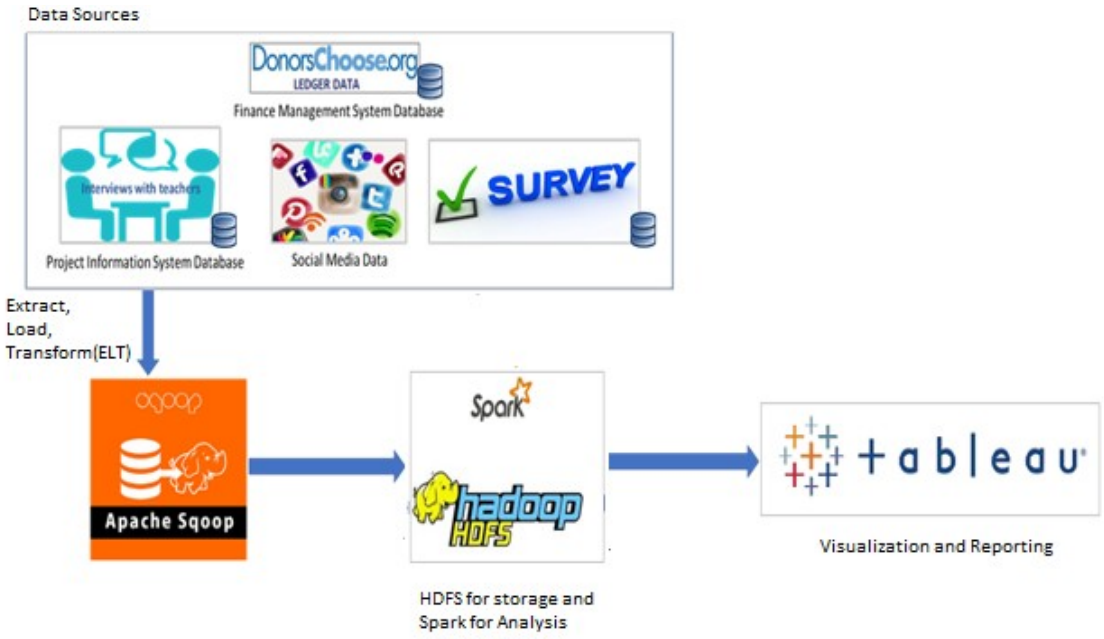
In the next few sections, we will discuss about the data storage and architecture utilized for our project.

Importance of Big Data Architecture

Big data architecture is the foundation of big data analytics. To formulate an effective big data architecture, the primary challenges that need to be overcome are data storage, analytics and visualizations. The entire process of big data analysis follows a well-defined architecture that begins with data storage. Choosing the right storage is not only important for real time analytics, but also for data mining in future. Some of the essential components of big data architecture are as follows:

- Data Ingestion: This is the entry point of raw data from various sources into big data platform.
- Data Storage: This will evaluate the data and organize it in different sections. The data storage platform should be able to handle large volume of data, have the ability of scaling to keep up with growth and should also have the ability to support data analytics.
- Data Analysis: In this step, data will be analyzed to derive meaningful insights.
- Visualization: In this step, the analysis will be made available to end users for aiding them in making business decisions.

Data Architecture for our project



In our project, we are dealing with multiple data sources such as Surveys, Interviews and Ledger data. Scaling is an important aspect in our application as we plan to include social media data going forward, which will increase the data complexity and size to a large extent. Therefore we have chosen Hadoop Distributed File System (HDFS) as our data storage platform. HDFS is the data storage system used in Hadoop applications. It is the right choice in our application because it is cost effective and highly scalable. It has low latency and fault tolerance. In our project, we will use Extract, Load and Transform (ELT) process and not the traditional Extract, Transform and Load (ETL) process, because in the former process, data is immediately loaded after extraction. This will enable unlimited access to all the data at any time and saves time and efforts for Business Intelligence users and analysts.

ELT process in our architecture

For transferring data from sources to big data platform, we will use Apache Sqoop as the ELT tool. It is a tool in the Hadoop ecosystem which is used for data transfer between RDBMS and Hadoop. Sqoop is the right choice for us because the data sources in our application are available in multiple relational databases. Sqoop works as follows:

1. Sqoop uses export and import commands to transfer data to HDFS. Internally, it uses Map Reduce program for data storage in HDFS. Map task is associated with retrieving data from external databases, while Reduce task will be used for placing the retrieved data into HDFS.
2. Sqoop has the capability of connecting various data sources to Hadoop and offers parallel processing along with fault tolerance.
3. For connecting to our data sources, Sqoop will use API connectors. If needed, we can also create custom connectors to make our database interactions better. The steps in the ELT process are:

4. Sqoop will analyze the external database and will formulate metadata in order to retrieve the data.
5. In the next step, it will use a Map task to load data into HDFS by using the metadata. It splits the input data into multiple map tasks. Data will be imported to a file related to the retrieved table. Sqoop also allows file name to be specified by the user. This file, which is now available in HDFS, will have comma separated values (CSV) for each column, and each row will be placed in a new line. Based on the table size, more files may be created. For importing data, Sqoop supports avro format also, but in our project, data will be loaded as CSV files.
6. In our application, batch processing will be used to extract and load the data, thereby ensuring that input is collected for a specified interval of time and transformations are run on it in a scheduled way.
7. The data transformation takes place after the data is loaded into Hadoop. This step consists of multiple data manipulations, such as moving, splitting, translating, merging, sorting, pivoting, and more. For example, in our project, we will sort the Project Data based on Project_Expiration_Date and the total Resource price will be aggregated into a single field by multiplying Resource_Quantity and Resource_Unit_Price.

Cluster Planning and Network requirements in Hadoop

Cluster planning plays an important role in choosing the right hardware configuration for Hadoop components. For big data analytics, the current recommended machines are dual CPU with 4 to 8 cores each, at least 48GB of RAM up to 512GB and standard rack-mountable 19″ servers. In our application, we will follow the standard cluster size to achieve good analytical capabilities. Network is also an important consideration while planning data architecture using Hadoop ecosystem. A fast network not only allows data to be exported and imported quickly, but also improves the performance for analytics and computations. A 10 Gigabit Ethernet network provides a simple, cost-effective solution. Research conducted by Intel has shown that using 10 Gigabit Ethernet rather than 1 Gigabit Ethernet in a Hadoop cluster can improve performance by up to 4 times using conventional hard drives.

Data Storage in HDFS

When HDFS takes in data, it breaks the information into blocks and distributes them to various nodes in the cluster, thereby allowing parallel processing. HDFS also achieves fault tolerance by replicating each piece of data multiple times. In our project, each information block will be of 128 MB size and we will use a replication factor of 3. Therefore, the first copy can be found in the same rack, but on a different node while the second copy will be available in another rack altogether. This replication process will ensure that processing will continue, even if there is a node failure. On a Hadoop cluster, the data within HDFS and the processing unit are housed on every machine in the cluster. This has two advantages:

- It adds redundancy to the system in case one machine in the cluster goes down
- It achieves data locality by bringing the data processing software into the same machines where data is stored, thereby increasing the speed of information retrieval.

NameNode is the central node in an HDFS system. NameNode has the directory tree of all the files in HDFS filesystem. It manages data placement and monitors server availability. Whenever Client applications want to locate a file, or when they want to add/copy/move/delete/modify a file, they can talk to the NameNode. The NameNode responds to such requests by returning a list of relevant DataNode servers where the data lives. In HDFS, capacity and performance can be scaled and tuned by adding more DataNodes.

Data Processing Framework: Spark

In our project, we will be using Apache Spark as the data processing framework. Spark is an open source cluster computing framework. Spark has the following characteristics:

- It provides parallel processing and fault tolerance.
- It is 100 times faster than MapReduce due to its in-memory computing capabilities. Resilient Distributed Dataset (RDD) is the core concept in Spark framework. RDDs are immutable and support two types of operations: Transformation and Action. They are also fault tolerant because they can recreate and recompute the datasets.
- Spark includes MLlib, a library that provides a growing set of machine algorithms which we will using in our project to build a recommendation system.

In our project, we will be using PySpark as the programming language. We have chosen Pyspark for our application because it makes it easy to combine local and distributed data transforming operations, while significantly accelerating analytics without increasing computing costs.

Analysis and Reporting using Tableau

The primary benefit of a reporting and visualization tool is that it enables decision makers to better comprehend complex data and understand the data through interactivity. It also allows the business leaders to visually discover hidden relationships between day-to-day processes and business performance. The goal of our big data solution is to provide insights into the data and build a recommendation system. We also need to visualize the impact of our recommendation system. We have chosen Tableau as our reporting tool because:

- It can tell stories with visualizations, thereby making the results clearly understandable to business stakeholders.
- It also has the ability to connect to multiple data sources. For our project, we will use Spark SQL connector to connect to the big data platform.

Next, we discuss about the actual data science used in our project that is, which algorithms will be used to implement the recommendation engine and how they will be utilized.

We will be implementing the three approaches for recommending projects to donors which are explained in the below sections.

Content-Based Filtering

We plan to perform content-based filtering in the following two ways:

1. Using text-based features

In this approach, we will be utilizing the project essay feature available in the dataset. This feature contains a detailed description about the project being undertaken, the goal that needs to be reached. It is primarily a paragraph containing all these details. We will be doing natural language processing (NLP) on the values for this feature to determine the topic that a particular project most relates to.

2. Using non-text-based features
   We have identified the below listed features to be non-text-based features that we will be using in our algorithm. This means that if a particular project belongs to Arts category and Donor 1 has donated to this project in the past and there is another project which we identify as belonging to the Arts category then, Donor 1 would probably like to donate to the second project also. Moreover, we will not be using just the Project Category and Project Subcategory features in the dataset. We will utilize Grade Level, Resource Category and School State to find similar projects and then recommend them to the donor.

For the text-based features approach, we will be performing a series of steps to get to the recommendations.

First, we will be cleaning (or preprocessing) our dataset in the following ways:

1. We will be renaming the column names to replace spaces with underscore and convert them to lower case for easy processing of data.
2. We will also be merging the 3 separate CSV files (for donors, donations and projects) into a single dataset so as to consolidate all the information at a single place such that we do not need to perform joins at every step-in order to get donations data for the particular donor in order to produce recommendations.
3. Due to storage and computation power limitations, we have reduced our dataset to 20,000 rows. Therefore, the result of the recommendation algorithm may not be as opposed to considering 4 million rows.
4. In the merged dataset, we will also be converting all text in the Project Essay column into lower case so that the algorithm does not treat upper and lower-case letters differently.
5. We have also converted the Project IDs in the projects and donations files from ‘long’ strings to integers so as to perform merging of the datasets faster.
6. Next, we will be filling all missing values in the Project Essay column with blanks so that this does not get a different treatment by the algorithm.
7. We will remove stop words (like the, is, are etc.) from the project essay text since these words do not have any significance in classifying the projects. Doing so will also make the text shorter and help to reduce the text processing time. Besides the standard English stop words, we will also add more stop words like ‘donotremoveessaydivider’, ‘student’, ‘help’ etc. since these words either do not add any meaning to the sentences or appear too commonly in all project essays.
8. We will also utilize stemming in order to convert the variants of a single word into the stem word. For example, the words run, ran and running represent the same word in different tenses. For text processing, it does not add value to process all variants of the

same word. Therefore, we reduce these words into their root or stem (run in this case) and then remove the multiple occurrences of the root word.

9. From experience and some research, we realized that stemming does not work very well in all cases since it does not take morphological analysis into consideration. For example, it converts 'bullying' to 'bulli' which is not a word in the English dictionary and does not make sense. To avoid this, we will use lemmatization which will convert the variants of the word into the actual root word.
10. After these cleaning steps, we plan to assign strengths (or weights) to all the projects on the basis of their donation amounts. This is known as smoothening where we apply a logarithmic function to all values. Before doing this, we will calculate the sum of the amounts for all donations made by a donor to a single project. This is because it is possible that a donor makes multiple donations to the same project. The smoothening will be performed on each donation amount per donor per project.
11. We will use the **TFIDFVectorizer** library in Python in order to generate the TF-IDF matrix for all words in the Project Essay column. The TF-IDF matrix gives the probability of a particular word appearing in a particular project essay. A simple word count of each word in each essay will not be useful because there will be instances when certain unimportant words (like and, the, in etc.) will appear in almost every essay and therefore we would not be able to distinguish the essence or topic of each essay properly. TF-IDF overcomes this limitation since it incorporates the removal of English stop words. It also defines a threshold above which if the frequency of a word occurs, that word will be ignored. This is so that frequently appearing words (across documents) do not produce unnecessary bias and the essays do not get classified into unrelated topics.

The essence of our text-based content-based filtering algorithm is that we build the donor and project profiles and then determine the similarity between them using Cosine Similarity.

A project profile will be obtained using the TF-IDF matrix. On the other hand, a donor profile is determined on the basis of the projects to which they have donated. That is, first we obtain the profiles for all projects to which the donor has donated and then we calculate the product of the donor project profiles and their strengths which we had calculated in Step 6 in the previous section. We then add all these products together and divide this by the total number of profiles. It will be something like below:

weighted average = $a_1x_1 + a_2x_2 + \ldots + a_nx_n / (a_1+a_2+\ldots+a_n)$

This calculation is necessary because we want to ensure that the projects to which a donor has donated say, 5 times get more importance than the ones to which they have donated only once or twice. The projects to which they have donated more number of times are more likely to be candidates for future donations and therefore should appear at the top of the recommendation list.

Using Cosine Similarity, we calculate the similarity of the donor profile with all project profiles and then rank them in order of their similarity level.

If time permits, we plan to implement the recommendation engine based on non-text-based features as well.

Why we chose TF/IDF for our project?

There are other methods such as Naïve Bayes algorithm for creating text-based recommendation system, but TF/IDF works better for our project because it gives definite answers to the question "which Projects match", whereas techniques like Naïve Bayes will cluster the projects into broader groups, which may reduce the accuracy of the recommendation system.

Collaborative Filtering

Collaborative Filtering is a technique which is used for making predictions (filtering) about the interests of a user by collecting preferences of other users with similar interests (collaborating). In our project, we will use Collaborative Filtering to recommend projects to a donor by collecting preferences of other donors. This approach will be used to predict what a particular donor will donate to, based on the projects that similar donors have donated in the past. This approach works on the assumption that if Donors A and B have donated to the same Projects, then donor B is more likely to match A's preference for a given project, than that of any other random donor.

Latent Factor Model and Singular Value Decomposition (SVD)

In our project, we will use a latent factor model which compresses donor-project matrix into a low-dimensional representation using latent factors. Latent factors are variables which are not directly observed, but they are inferred using mathematical techniques. The reduced presentation using such latent factors is utilized for creating project-based neighborhood searching algorithms. In our Project we will use a latent factor model known as Singular Value Decomposition (SVD).

The Singular-Value Decomposition is a matrix decomposition method which is used for stripping a matrix to its constituent parts. This makes subsequent matrix calculations simpler. The following formula is used in SVD:

A = U. Sigma. V^T, where A is the real m x n matrix that we wish to decompose, U is m x m matrix, Sigma is m x n diagonal matrix, and V^T is the transpose of n x n matrix where T is a superscript. The diagonal values in the Sigma matrix are known as the singular values of the original matrix A. The columns of the U matrix are called the left-singular vectors of A, and the columns of V are called the right-singular vectors of A.

The steps for collaborative filtering are:

1. We will create a sparse matrix with donor ids in rows and project ids in columns. A sparse matrix is a matrix in which most of the columns are zero.
2. We will use SVD to get latent factors from the sparse matrix created. In our project, we will obtain a number of latent factors using SVD technique. We will fit and evaluate model with different number of factors and select the one that performs best on our test set.

3. After the factorization, we will reconstruct the matrix by calculating the dot product of the factors. This will give us a resulting matrix, which is not sparse anymore, but it will consist of the generated predictions of projects which the donors have not yet donated to.

Why we chose SVD for our project?

There are other methods for building a Collaborative Filtering recommendation system such as Principle Component Analysis (PCA), but we have chosen SVD because it works well with large and variable data sets. Also, it is very efficient even when it is applied to large matrices, unlike other algorithms.

Hybrid Filtering

Hybrid recommendation system is based on combination of content based and collaborative filtering. It provides recommendation based on the combination of what content a user liked in the past as well as what similar users like. Research has shown that a Hybrid Recommendation System could be more effective than Content based or Collaborative Filtering in some cases. They are also used to overcome some of the common problems in a recommendation system such as Cold Start. Cold Start problem arises when there is a lack of information about the users or items. In our project, we are going to use a very simple Hybridization technique wherein we multiply the Content Based Score and the Collaborative-Filtering score, and then rank the resulting hybrid score to derive the recommendation system.

Comparison of Content Based, Collaborative Filtering and Hybrid Recommendation System

In Content-Based Model, new projects without any donation can get recommended to donors, while in the Collaborative Filtering Model, only projects which have already received at least one donation will get the opportunity to be recommended. Ideally, we would want combination of both, hence Hybrid based approach may be better. We will compare the model performances of all these techniques to find out which one suits our project the most.

Model Evaluation - Top-N accuracy metrics

In our project we will work with Top-N accuracy metrics, which evaluates the accuracy of the top recommendations provided to a user, comparing to the items that the user has already interacted with. The evaluation model works on the following principle:

- For each donor,
    - For each project that the donor has donated to, sample 100 other projects that the donor has not donated to.
    - Using the recommendation system, generate a ranked list of recommended projects, from a set composed of one donated project and 100 non-interacted (not donated) projects.
    - Compute the Top-N accuracy metrics for this donor and project, from the recommendation ranked list.
- Aggregate the global Top-N Accuracy metrics.

The accuracy comparison will be visualized using Tableau.

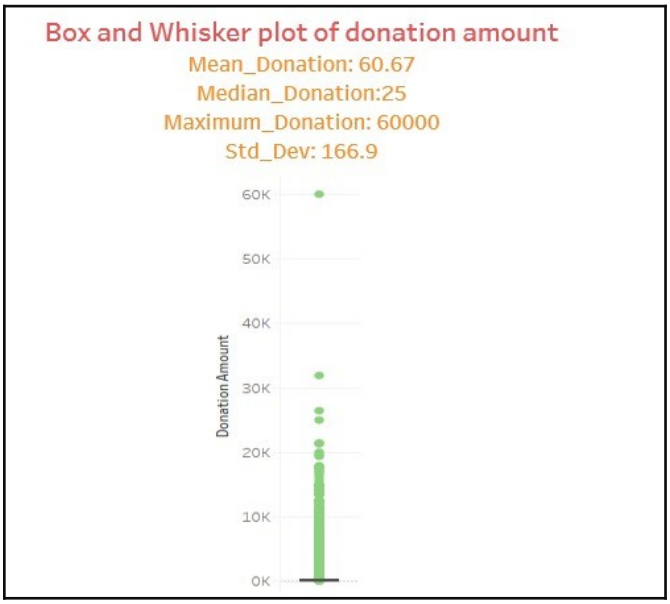
Data Exploration using Tableau

Following table created using Tableau shows the top 10 projects which received the most number of donations. These projects have a higher probability of being recommended when using the Collaborative Filtering recommendation systems.

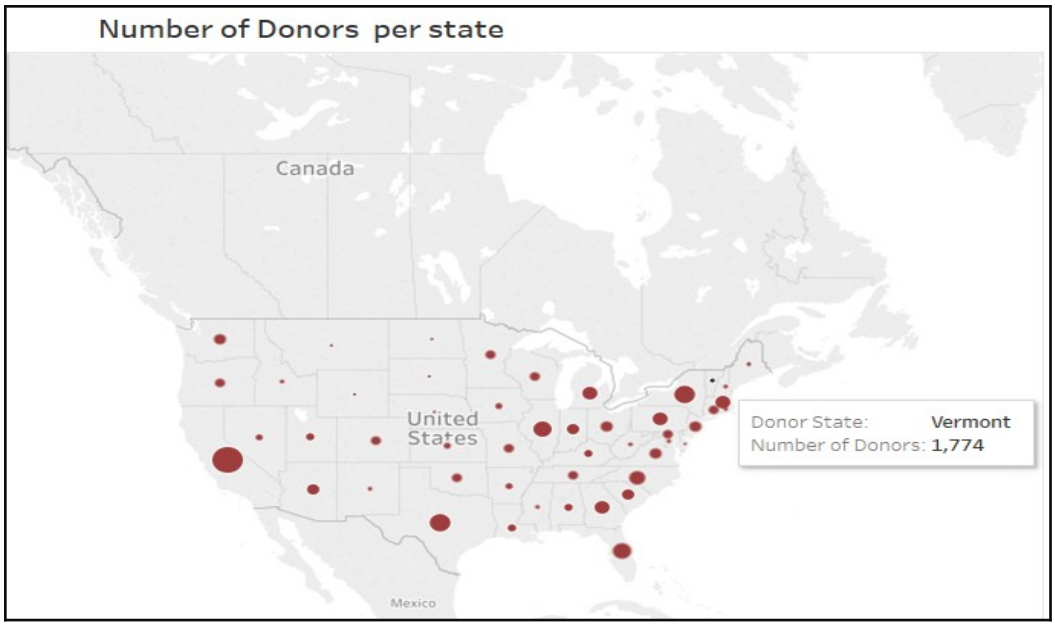
Top 10 Projects with the most number of donations

Project ID	Count_of_Donations	Donation Amount
c5582cb6dc5d45a7f4d9a..	474	556
bd89d8f499ff23ce7a421..	472	572
ea99d0493c7c668890eca..	538	917
a720b2e32df79c52f8926..	631	1,904
31c9862995b9f65b0e92d..	499	2,425
a7028bc2602104e658ef7..	600	6,223
2450ee51be5cb2443ef52..	476	17,078
d6a260b9099aabdac7f5c..	663	47,334
132141bfb266d8deedf08..	536	51,948
c34218abf3fecd36be013..	863	108,248

Following plot which has been created using Tableau shows the distribution of different donation amounts. This visualization has been created using Box and Whisker Plot. This helps us in finding the outliers in our data.



The following image created using Tableau shows the distribution of donors across the different states.



Recommendation Results

Note: Based on the feedback for our presentation, we updated the algorithm for Hybrid filtering by replacing the multiplication of the content-based and collaborative scores, we took an average of the two. By doing so, we noticed that the recommendation strengths using the Hybrid model improved and the bias towards Content-based model reduced but the recall still remains the same.

In order to verify if the recommendations provided by our models are in fact relevant to the donors, we found out the tokens which are relevant to each donor. In order to do this, we have built the project and donor-project profiles and based on them we have identified the top ten tokens which hold the maximum relevance for the particular donor. This shows us the nature of projects which the donor has donated to the most number of times. We have then displayed the recommendations provided by each model. These are shown below.

Token-Relevance table for Donor 1:

	token	relevance
0	puzzle	0.27
1	wooden puzzle	0.14
2	wooden	0.12
3	families	0.10
4	fine motor skills	0.10
5	motor skills	0.09
6	fine motor	0.09
7	motor	0.09
8	fine	0.09
9	olds	0.08

Content-Based Recommendations for Donor 1:

```
In [43]: cbr_model = ContentBasedRecommender(projects)
         cbr_model.recommend_projects(mydonor1)
```

Out[43]:

	rec_strength	project_id	project_title	project_essay
0	1.00	50751655c488db95985f464e99a84a56	electric keyboards needed to rock out in jazz ...	the jazz program at the middle school is growi...
1	0.15	4e78ceb6fc071faf6b26b50ea0efd31f	jepson school needs jazz!	"jazz is played from the heart. you can even l...
2	0.10	e4dda3c0bf01a74fe9384b1135c7c882	bang! boom! crash!	there are so many different styles and genres ...
3	0.07	e3498e0205078a3496c651c17cc2050f	how can the band play without music?	my students love to play in the band. the prob...
4	0.07	cbfe19eab57321fe1e1b9c6831a214e7	materials for disadvantaged music students	whether it was singing in the school play or ...

Collaborative Filtering Based Recommendations for Donor 1:

```
In [47]: cfr_model = CFRecommender(cf_preds_df, projects)
         cfr_model.recommend_projects(mydonor1)
```

Out[47]:

	rec_strength	project_id	project_title	project_essay
0	4.70	50751655c488db95985f464e99a84a56	electric keyboards needed to rock out in jazz ...	the jazz program at the middle school is growi...
1	0.01	c9d82483aa71c9ec178286c4e9562143	help! we need basic supplies for building lite...	can you imagine trying to learn how to write w...
2	0.01	0a69dc93b48520f16a5d9c1804a3e55d	worker bees	elbert hubbard said, "the best preparation for...
3	0.01	55bdd3141b8e8c811b3f72682446d9f8	uke troupe! new music group! whoop! whoop!	the ukulele is a noble little instrument... an...
4	0.00	d3fc101ea24e26443dbfe9bc7560d08c	flutter by butterfly	i teach first grade at an amazing public schoo...

Hybrid Recommendations for Donor 1:

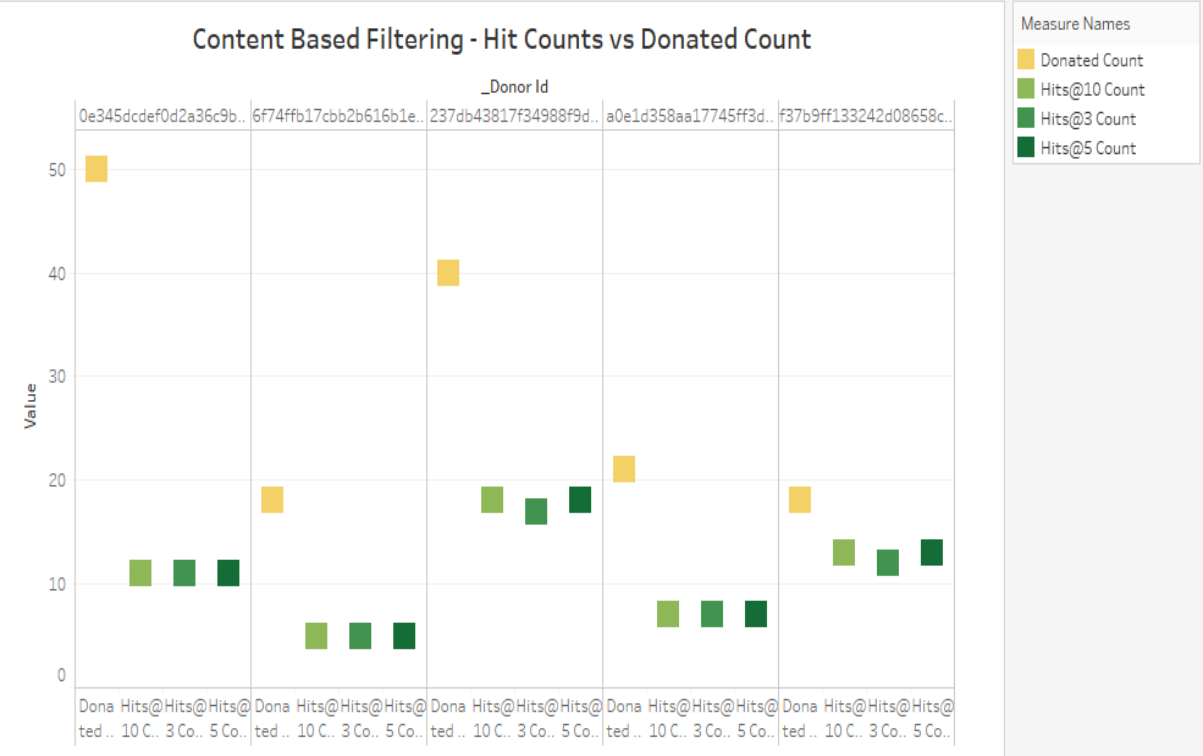
```
In [50]: hybrid_model.recommend_projects(mydonor1)
```

Out[50]:

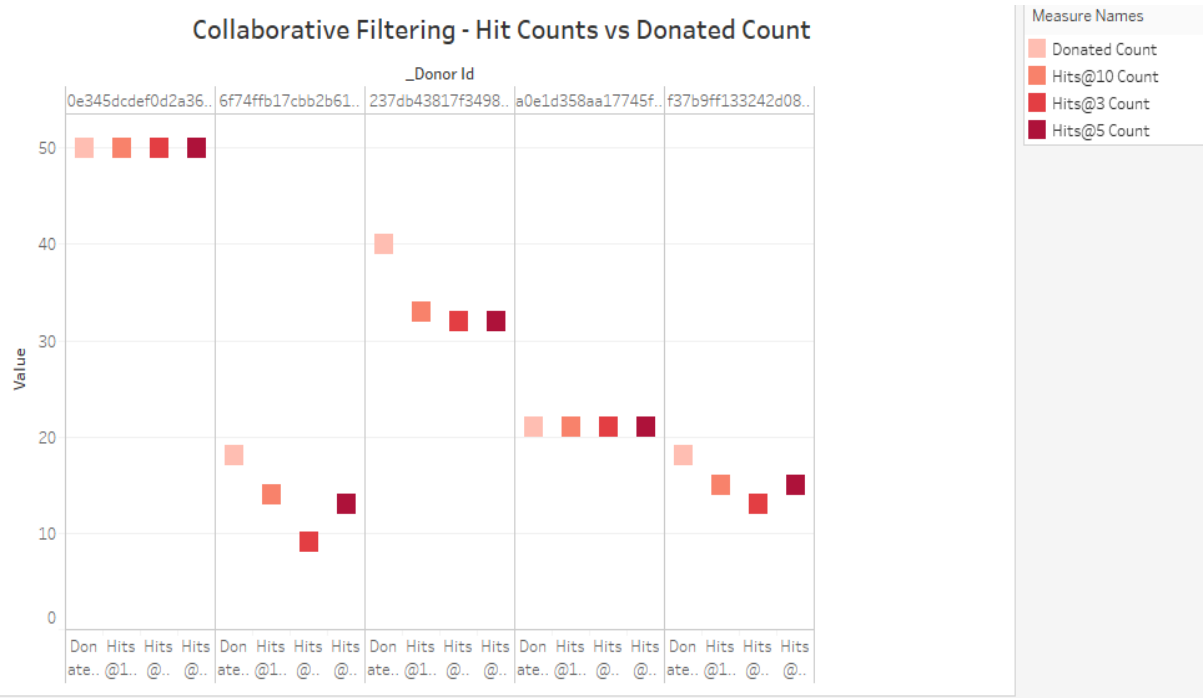
	rec_strength_hybrid	project_id	project_title	project_essay
0	2.85	50751655c488db95985f464e99a84a56	electric keyboards needed to rock out in jazz ...	the jazz program at the middle school is growi...
1	0.08	4e78ceb6fc071faf6b26b50ea0efd31f	jepson school needs jazz!	"jazz is played from the heart. you can even l...
2	0.05	e4dda3c0bf01a74fe9384b1135c7c882	bang! boom! crash!	there are so many different styles and genres ...
3	0.04	e3498e0205078a3496c651c17cc2050f	how can the band play without music?	my students love to play in the band. the prob...
4	0.03	cbfe19eab57321fe1e1b9c6831a214e7	materials for disadvantaged music students	whether it was singing in the school play or ...

After running our recommendation model on the dataset, we arrived at the following results. The below graph shows the top 3, top 5 and top 10 hit counts for each algorithm for 5 random donors. The donors have been obtained by performing a head () on the recommendations output therefore, they are not ordered in any way.

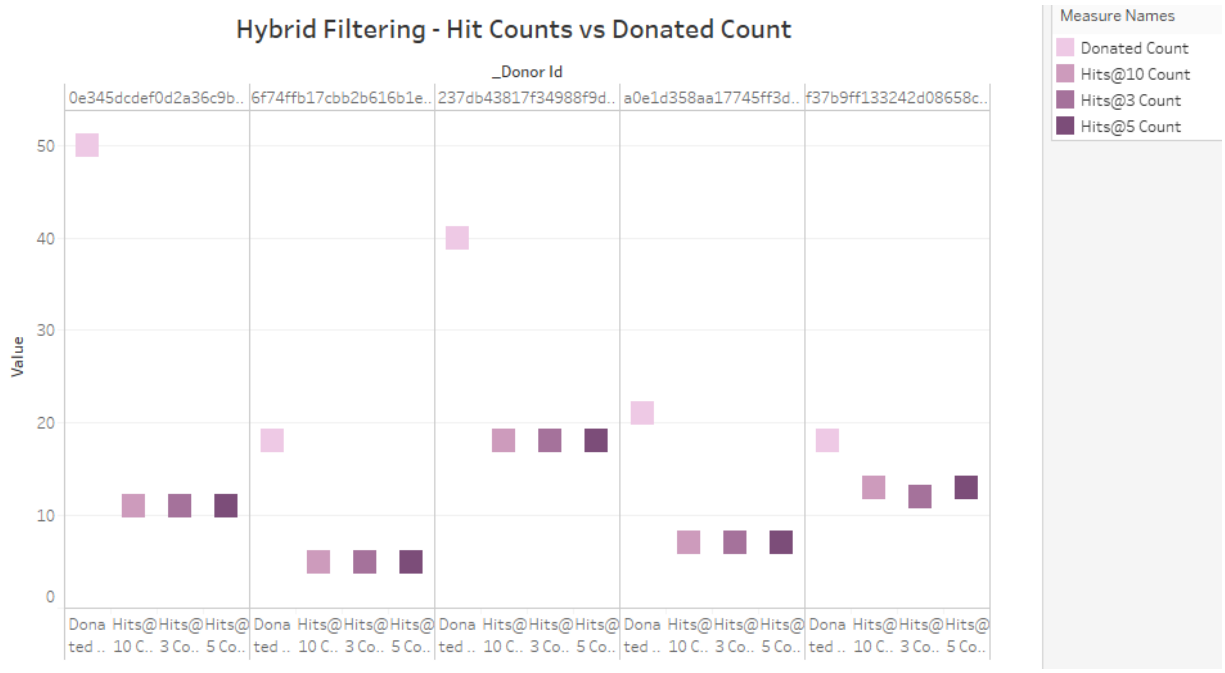
1. Content-Based Filtering Results



2. Collaborative Filtering Results



3. Hybrid Filtering Results



The following images show the global recall@3, recall@5 and recall@10 for all 3 algorithms. These have been obtained by considering all the donors in our test set.

The tabular representation of the output is as follows:

modelName	recall@10	recall@3	recall@5
Collaborative Filtering	0.42	0.39	0.42
Content-Based	0.85	0.85	0.85
Hybrid	0.84	0.84	0.84

It can be noted that the recall for the Hybrid model lean more towards the Content-Based model as the recall for Collaborative model is very low based on our algorithm.

Conclusion

It can be inferred from the results that the content-based filtering model has a very high accuracy while collaborative filtering model has a low accuracy. This can be attributed to the reason that for collaborative filtering, the number of donors is very less (due to reduced dataset size) therefore, there are not enough donors to compare to in order to provide more relevant recommendations. Moreover, in our dataset most of the donors are first time donors which is why the system would not know which projects to recommend to such donors.

On the other hand, for content-based filtering the accuracy is very high because we completely rely on the project essay text to recommend projects. It does not consider the fact that the number of donors who have donated to that project in the test set are very less. Moreover, there are many projects in the test set which have only one donation so the recall values of donations to such projects is 100 percent.

Future Scope

We have identified the following tasks that can be performed in order to improve the accuracy of the recommendation engine.

1. If provided with high memory and computational power machines, we can perform the recommendation on the entire dataset since this is expected to balance out the recall values of content-based and collaborative filtering models.
2. Instead of limiting our recall values to be based on top 3, top 5 and top 10 only, we can also consider top 20, top 50 etc. in order to obtain more accurate recall values.
3. For performing natural language processing on the project essay, we should consider using technologies like Word2Vec and DeepNets in order to consider words with different meanings in different contexts and provide much more accurate recommendations in terms of content.
4. We would also like to utilize non-text based features like project category, project subcategory and resource category in addition to the text-based recommendation.

5. Currently, the hybrid recommendations have been obtained by multiplying the recommendation scores of content-based and collaborative models. We can look into some better ways of calculating hybrid recommendation scores.