Article

Not peer-reviewed version

Algorithm-based Data Generation (ADG) Engine for Data Analytics

Iman I. M. Abu Sulayman , Peter Voege , Abdelkader Ouda *

Posted Date: 19 June 2023

doi: 10.20944/preprints202306.1378.v1

Keywords: Data Generation; Anomaly Data; User Behavior Generation; Big Data

Article

Algorithm-Based Data Generation (ADG) Engine for Data Analytics

Iman I. M. Abu Sulayman [1], **Peter Voege** [2], **Abdelkader Ouda** [2*]

[1] Taif University, P.O. Box 11099, Taif21944, Saudi Arabia; email: iman@tu.edu.sa.
[2] Department of Electrical and Computer Engineering, Western University, London, Ontario, N6A 5B9, Canada. emails: pvoege2@uwo.ca, aouda@uwo.ca.
* Correspondence: aouda@uwo.ca

Abstract: The rising importance of Big Data in modern information analysis is supported by vast quantities of user data, but it is only possible to collect sufficient data for all tasks within certain data-gathering contexts. There are many cases where a domain is too novel, too niche, or too sparsely collected to adequately support Big Data tasks. To remedy this, we have created ADG Engine that allows for the generation of additional data that follows the trends and patterns of the data that's already been collected. Using a database structure that tracks users across different activity types, ADG Engine can use all available information to maximize the authenticity of the generated data. Our efforts are particularly geared towards data analytics by identifying abnormalities in the data and allowing the user to generate normal and abnormal data at custom ratios. In situations where it would be impractical or impossible to expand the available dataset by collecting more data, it can still be possible to move forward with algorithmically expanded data datasets.

Keywords: data generation; anomaly data; user behavior generation; big data

1. Introduction

When endeavoring to develop a data analytics system, one of the foremost and daunting hurdles is acquiring a suitable dataset for the task at hand. It is not only imperative that the dataset encompasses relevant data pertaining to your objective, but it must also be abundant in quantity. Additionally, in numerous cases, this data must be appropriately labeled. This is especially crucial for data analytics systems like anomaly detection, where the dataset needs to distinctly differentiate between normal and anomalous data.

In cases where obtaining real-world data of adequate size to fulfill the requirements of the desired anomaly-detection task proves challenging, and there are no feasible means to gather additional data, the only option left is to artificially generate the required data by accurately emulating the target conditions.

Due to the immense scale of these datasets, it is not feasible to create them using any other methods besides algorithmic generation. If successfully accomplished, the potential for executing a significantly larger number of anomaly detection tasks should considerably expand as well.

Consequently, the goal of this paper is to construct a diverse set of algorithms (ADG Engine) that have the ability to generate high-quality labeled data, which can be readily accessed and utilized by projects focused on anomaly detection.

The objective of ADG Engine is to achieve this goal by examining five prevalent data platforms employed in data analytics endeavors. For each data platform, ADG Engine will develop an algorithm with the capability to analyze the data and ascertain attributes such as the proportion of abnormal behavior. This will enable us to generate additional data that aligns with the existing dataset.

The rest of the paper will be structured as follows: Section 2 will center around the most pertinent research in the field of dataset generation, Section 3 will describe the importance of ADG Engine from several features, Section 4 will describe the method we use to generate data, Sections 5 & 6 will explore the details and configuration of the model, and in Section 7 we will test our model and discuss the results.

2. Related Work

There are several papers that use data generation engines or data expansion. In [1], Patki et al. introduced the Synthetic Data Vault (SDV), a system designed to generate synthetic data for relational databases. Their research focuses on developing generative models that can sample from the model to create synthetic data. The SDV algorithm computes statistics by considering the relationships between different database tables. It utilizes a state-of-the-art multivariate modeling technique to capture the underlying patterns in the data. By iterating through all possible relations, the SDV builds a comprehensive model of the entire database. Once the model is established, the SDV can synthesize data by sampling from any section of the database using the available relational information. This paper accomplishes most of the requirements needed to match ADG Engine. However, there are still two differences between this paper and ADG Engine, which are the necessity of an initial data distribution and anomaly features. ADG Engine creates the data from scratch and has anomaly features included at user-specified ratios.

Another research is presented by E. Lopez-Rojas and S. Axelsson in [2] which is a BankSim model BankSim is a software simulation tool that replicates bank payment transactions using combined and summarized data obtained from a bank in Spain. The primary goal of BankSim is to create artificial data that can be effectively employed in studies related to detecting fraudulent activities. To develop and fine-tune the simulation model, statistical analysis and Social Network Analysis (SNA) methods were applied to examine the connections between merchants and customers. The ultimate aim is for BankSim to accurately simulate different scenarios, encompassing both normal payment transactions and pre-defined fraudulent patterns. This paper is designed for Fraud detection which is a bit closer to ADG Engine. There are several differences between ADG Engine and this model. In terms of data size, this paper is limited to Data Size because it has a fixed amount of observations. The paper is not flexible enough to add or remove features based on user design. The research is only using one dataset for all users, which is incompatible with rational datasets. The paper only covers one application, which is the credit card application.

Zhao, et al. [3] developed a Data Generation Algorithm that utilizes Complex Event Processing (CEP) techniques. CEP involves processing real-time data streams and extracting valuable information from events as they occur. The primary objective of complex event processing is to identify significant data patterns in real-time scenarios and promptly respond to them. The authors introduce the concepts of selective event flow, sequential event flow, and causal event flow. Experimental findings demonstrate the effectiveness of this method. This paper has two differences from ADG Engine; anomaly features which are not included in this paper, and a variety of applications such as social media and credit cards are not provided in this paper.

Research paper [4] represents a model of uncertain data and corresponding uncertain data generation algorithms with different types of uncertain data. The analysis and experiments show that the algorithm proposed in their work has practicality as a tool. This research has only one feature of ADG Engine, which is unlimited data generation quantity, but it does not apply any other features related to anomaly factors or user behavior.

In their study [5], Kim et al. utilize a large-scale Location-Based Social Network (LBSN) simulation to establish a framework for simulating human behavior and generating synthetic, yet realistic, LBSN data based on typical human activity patterns. This data encompasses not only the geographical locations of users over time but also their interactions within social networks. To simulate patterns of life, the researchers assign agents (representing individuals) a range of "needs" that they strive to fulfill. For instance, agents return home when they are tired, visit restaurants when they are hungry, go to work to meet their financial obligations and visit recreational sites to socialize with friends and satisfy their social needs. This paper does not apply these features; anomaly columns, and rational datasets that include several datasets related to each other, the initial data is required to generate this data in this paper, and this data is focused on only one application.

In their research article [6], the authors introduce a synthetic dataset generator specifically designed for tabular data. This generator has the capability to identify and utilize nonlinear causal relationships among variables during the data generation process. Traditional approaches for discovering nonlinear causalities are often inefficient. To enhance efficiency, the authors limit the causal discovery process to features that appear in frequent patterns obtained through a pattern mining algorithm. To validate their approach, the authors develop a framework for generating synthetic datasets with known causal relationships. Extensive experiments conducted on various synthetic and real datasets with known causalities demonstrate the effectiveness of the proposed method. In this research, they have only two features that match ADG Engine. The remaining features that are related to abnormal observations or user behaviour are not included.

A. Kothare et al. in [7] used an open-source engine named Faker (v5.6.1) and Gaussian copula to create a platform that can generate datasets, based on user requirements as well as available resources. The user can also perform a variety of machine learning algorithms and differentiate their performance either over the generated dataset or a predefined dataset. This research uses a good tool to generate unlimited data observations with features that can be added or deleted as no initial data is required. However, the real data aspects for abnormal user behaviour are not included which makes this research and ADG Engine differ in five features.

In [8], the authors introduce the notion of a shadow database and present a framework for creating a shadow database that closely aligns with the distribution characteristics of a production database. Additionally, they develop and implement an integrated tool for generating synthetic data. This tool utilizes the data distribution profile, including histograms derived from the source data, as input to generate the corresponding shadow database. This research has several features like data size and related datasets but does not include the abnormal data design based on user behaviour for several applications.

In [9], the researchers conducted a study to explore the effectiveness of different synthetic data generation algorithms on various datasets. They examined the impact of SMOTE, Borderline-SMOTE, and Random data generation algorithms on 33 datasets. To achieve a comprehensive evaluation, each dataset was fully balanced through synthetic data generation. The datasets were then categorized into three groups based on their balance status: balanced, partially balanced-unbalanced, and unbalanced, according to the unbalanced ratio. This research is more of a study between dataset generators but the datasets are only applied to dataset size and real data aspects instead of abnormal features for datasets that are based on user behavior.

In their publication [10], the authors introduced a Generative Adversarial Network (GAN) combined with differential privacy mechanisms to generate a smart healthcare dataset that is both realistic and private. The proposed approach has the capability to generate synthetic data samples that closely resemble real data, while also ensuring privacy through differential privacy techniques. The approach accommodates different scenarios, such as learning from a noisy distribution or adding noise to the learned distribution. The research team validated and assessed the effectiveness of the proposed approach using a real-world Fitbit dataset. This research has real data aspects and a rational dataset structure with unlimited datasets. However, the other abnormal user aspects with several applications are not available in this research.

Table 1 shows a comparison study that shows the differences between ADG Engine and the existing Data Generation models. The first column is about generating a chosen number of observations in which you can enter any number you want. The second column is the flexibility of choosing a feature that is related to the research or generating more columns. The anomaly feature column indicates that the data has injected some anomaly features or observations. The rational Datasets feature is differentiated in whether the model is capable of generating multiple datasets related to one user or not. The real data aspect is a column that focuses on making conditions and relations between several features simulate real-world data. User Behavior is studying the model that generates all the observations based on the users and has several scenarios to describe the user behavior (such as working scenario, holiday scenario, and weekend scenario). Some models require initial data observations or an initial data distribution to generate more data that is not generated from scratch or at least using libraries. The last column is classifying research papers based on the use of multiple data applications like credit card applications, telecommunication applications, and Health care applications.

Table 1. Comparison Study of the Existing Data Generation Models.

Research Paper	Unlimited Data Size	Features Number Flexibility	Anomaly Features	Rational Datasets	Real data aspects	User Behavior	No initial data required	Number of applications Variety
[1]	✓	✓	x	✓	✓	✓	x	✓
[2]	x	x	✓	x	✓	✓	✓	x
[3]	✓	✓	x	✓	✓	✓	✓	x
[4]	✓	x	x	x	x	x	x	x
[5]	✓	✓	x	x	✓	✓	x	x
[6]	✓	x	x	x	✓	x	x	x
[7]	✓	✓	x	x	x	x	✓	x
[8]	✓	x	x	✓	✓	✓	x	x
[9]	✓	x	x	x	✓	✓	x	x
[10]	✓	x	x	✓	✓	✓	x	x
ADG Engine	✓	✓	✓	✓	✓	✓	✓	✓

3. Significance of the Work

The significance of the ADG Engine stems from its robust design, which incorporates a synthetic data generation engine based on user behavior encompassing both normal and abnormal instances. Notably, the Engine does not necessitate any initial dataset or data distribution. Instead, researchers can provide normal/abnormal distribution that aids in simulating real-world data aspects. Furthermore, researchers have the flexibility to select relevant features that reflect the interrelationships observed in their research problem. This engine enables the creation of rational datasets for various data applications, and researchers can choose from five different applications. The availability of these features and options sets this engine apart from existing studies, offering researchers the ability to detect anomalies more effectively in future research endeavors. [11]

Given the high likelihood of users utilizing multiple platforms, ADG Engine considers this factor by generating data from the same user across multiple datasets from different platforms. This approach enables ADG Engine to better capture real-world data collection scenarios, where individuals rely on various platforms to meet their everyday needs.

To accommodate diverse use cases, we aim to make key parameters of ADG Engine adjustable. For instance, the proportion of generated events that are considered anomalous can be tailored to specific requirements. Additionally, user attributes like marital status and employment status can impact specific dataset features, allowing us to control the generated data by adjusting these attributes. The combination of connecting multiple datasets through individual users and the ability to control the ratio of normal and abnormal behavior provides ADG Engine with remarkable flexibility and broad applicability.

Successful implementation of the engine's algorithms will enable the creation of effective anomaly detection systems, even when obtaining a large amount of training data is challenging. Furthermore, the principles embedded within ADG Engine's algorithms hold general applicability to platforms beyond those covered in this study, making them adaptable for generating data for various platforms.

4. ADG Engine Methodology

To implement ADG Engine, we created algorithms that generate events for each of the data platforms: credit card transaction data, bank accounts data, health record data, telecommunication data, and social media activity data. Instead of generating data for each one individually, isolated from the other platforms, ADG Engine re-uses the same users from one platform to another to accurately match the spread of real-life users across multiple services. From another perspective, ADG Engine creates one hundred different users and generates data instances for each user on all five data platforms. The data platforms ADG Engine is working with can be described by the following qualities: the number of features The Engine track, the number of keys in the data platform, and the number of anomaly features the Engine scan for. If there should be a need to introduce a sixth data platform, it will be entirely possible to describe that data platform using these same qualities. This relationship can be seen in the block diagram shown in Figure 1.

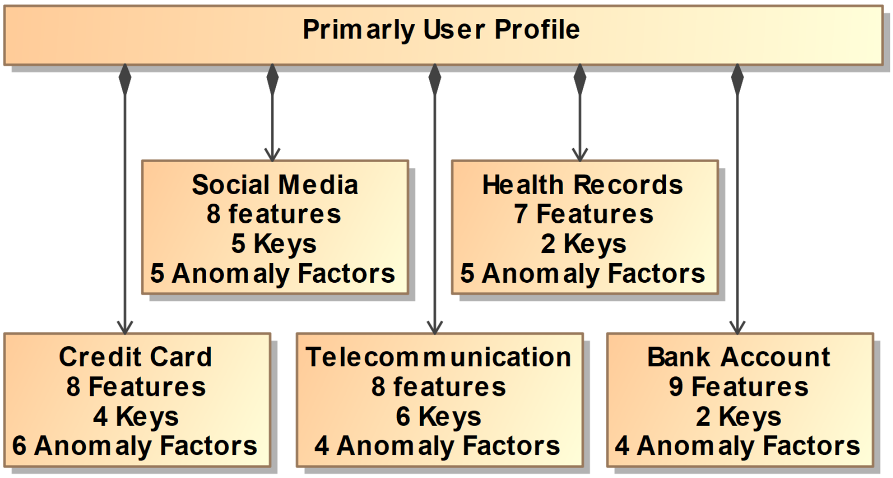

Figure 1. The Relationship Between User and Data Platforms.

To make the distribution of anomalies more accurate to real-life trends, we must first know the expected distribution of values. It is only in relation to the expected value that an anomalous value can be genuinely anomalous. If this distribution is constant for everyone, we simply need to specify the distribution ourselves before running ADG Engine. However, in cases where the expected outcomes vary from person to person, ADG Engine must have a way to dynamically calculate the expected distribution based on the user data. Figure 2 shows the process of data generation from the perspective of factors common to all five data platforms. For the user we are working with, ADG Engine begins by instantiating the platform profiles, one profile for each platform. Then, for each of these profiles, ADG Engine generates enough primary keys to match the requirements specified by the platform. These primary keys will be the seeds that ADG Engine will use to randomly generate the rest of the features. At the same time, ADG Engine identifies which features of the data platform are designated as anomaly features. Anomaly features are the features of the data platform that can have discernably anomalous data, and as such ADG Engine needs to determine what the expected distribution of data is. If the anomaly feature is static, ADG Engine uses a manually defined expected distribution defined ahead of time. However, if the anomaly feature is dynamic, ADG Engine generates a function that automatically calculates the expected distribution. Once the expected distribution has been determined for all anomaly features, ADG Engine has everything to create the final data platforms which will be explained in detail by the data Synthesis section.

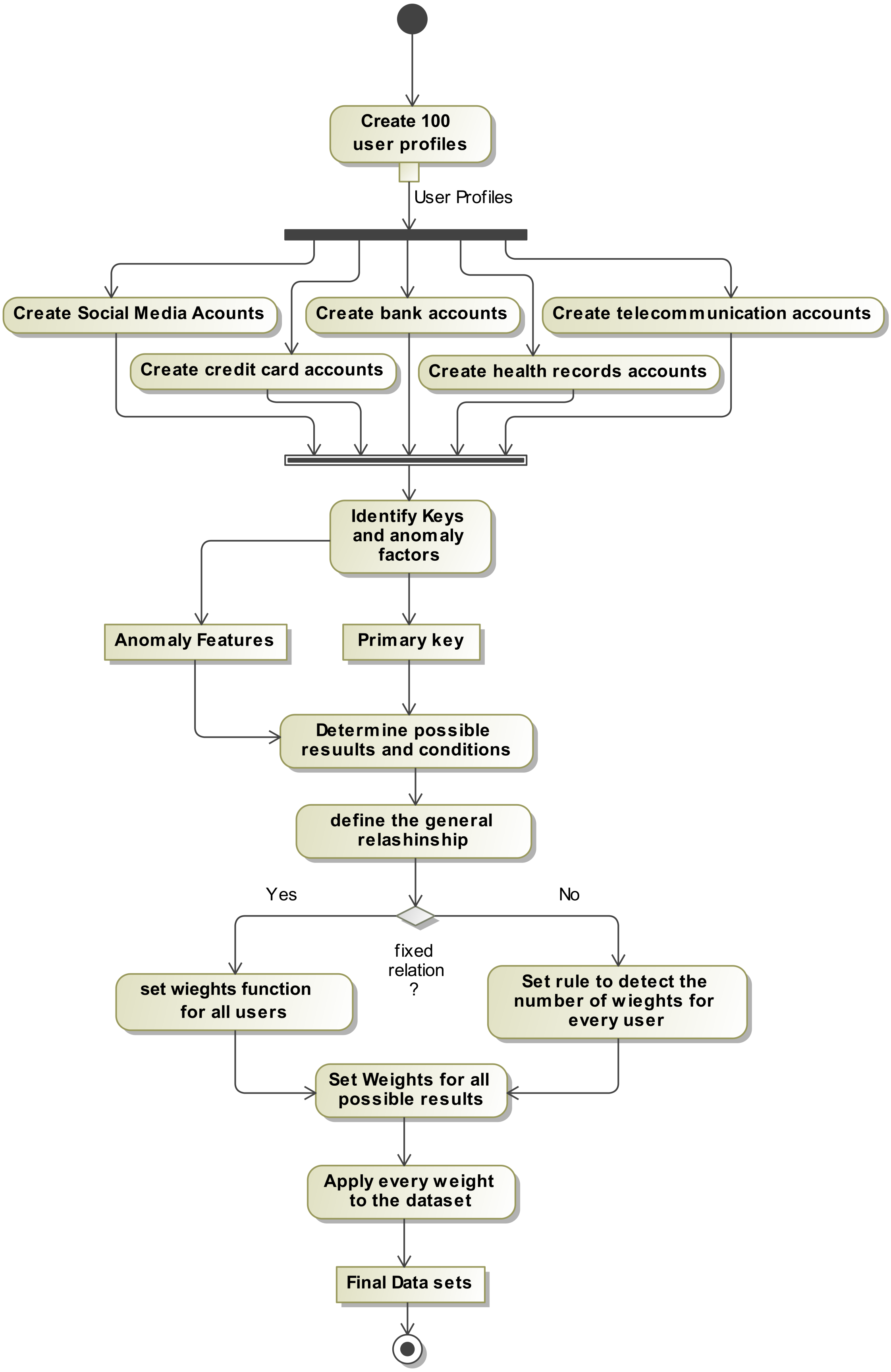
Create 100 user profiles
User Profiles
Create Social Media Acounts
Create bank accounts
Create telecommunication accounts
Create credit card accounts
Create health records accounts
Identify Keys and anomaly factors
Anomaly Features
Primary key
Determine possible resuults and conditions
define the general relashinship
Yes
No
fixed relation ?
set wieghts function for all users
Set rule to detect the number of wieghts for every user
Set Weights for all possible results
Apply every weight to the dataset
Final Data sets

Figure 2. Dataset Generation Process.

5. Data Synthesis

When generating a given feature, it is possible to specify a sequence of weights pertaining to the expected ratio of values. Between competing mutually exclusive options, the weights describe what outcome we consider likely. For example, if a time-based feature were divided into four segments: morning, afternoon, evening, and night, then a set of four weights could describe which of those sections the event is most likely to happen in. If the event is far more likely to happen in the afternoon than anytime else, a weighting of [10, 90, 10, 10] could accurately describe the ordinary baseline.

This weightings system is configurable by the user and can be set according to whatever parameters or input data are provided to it. All that must be true is that the sequence of weights specified provides a meaningful ratio between the various categories. It is also possible to specify the ratios cumulatively rather than relatively. The sequence of relative weights [10, 90, 10, 10] can be equivalently described as [10, 100, 110, 120]. If no sequence is provided at all, the events are assumed to be of equal probability.

5.1. Choosing Features Relevant to Anomaly Detection (Anomaly Factors)

From the data platforms, ADG Engine needs to figure out which features are going to be useful for the task of anomaly detection. This decision is made after considering various specific factors. For instance, ADG Engine studies the data outcomes associated with the feature, and looks at how many possible results it has, and if those results correlate with user behavior in some way. These possible results could be one option from a set of possibilities or a subsection of a numeric range. ADG Engine checks if the feature has a normal routine, and if it has some kind of ratio between normal and abnormal results. If it does, if one of the possible results is more common than another, ADG Engine can label the common result as 'normal behavior' and the uncommon result as 'abnormal behavior'.

The user's personal information can be highly relevant to this endeavor, as the user's qualities can impact the distribution of data in their observations and what trends are more normal than others. An employed person is more likely to travel on weekdays and shop outside of work hours, while a married person might see significantly different spending habits than an unmarried person. For instance, an unmarried man is unlikely to buy family-sized orders or toys for children, so transactions such as that would become abnormal.

5.2. Time-Based Anomaly Generation

Every data platform needs a time feature, but in ADG Engine, the time feature is also one of the main anomaly factors, as the timing of events is a common way for observations to prove unusual. To make use of this, ADG Engine begins, as shown in Figure 3, by selecting the start and end dates of the data platform, generating timestamps within that range. The default precision of the timestamps is to the minute, but it can be set to the second or the hour as needed. ADG Engine then splits up the timestamp range into two categories: weekday and weekend, and then again into four more: early morning, morning, afternoon, and evening. These divisions will allow us to correlate the time of events with a user's other qualities such as employment status. A person who is at work during weekdays will be more likely to do their shopping and spend over the weekend while relying more on transportation at specific times during weekdays.

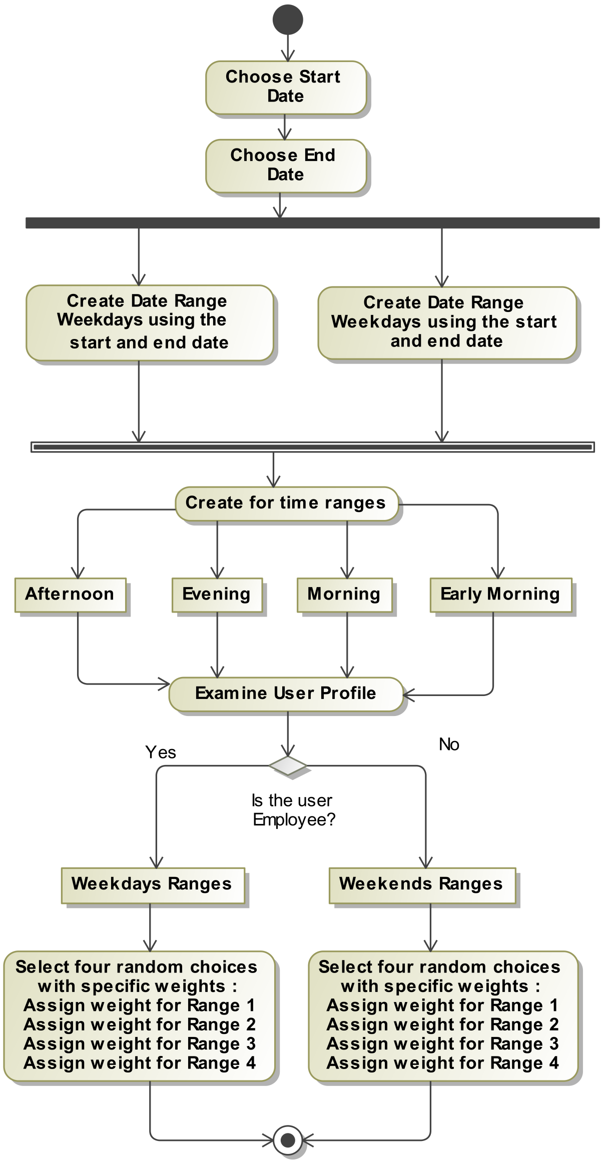
Choose Start Date
Choose End Date
Create Date Range Weekdays using the start and end date
Create Date Range Weekdays using the start and end date
Create for time ranges
Afternoon
Evening
Morning
Early Morning
Examine User Profile
Yes
No
Is the user Employee?
Weekdays Ranges
Weekends Ranges
Select four random choices with specific weights : Assign weight for Range 1 Assign weight for Range 2 Assign weight for Range 3 Assign weight for Range 4
Select four random choices with specific weights : Assign weight for Range 1 Assign weight for Range 2 Assign weight for Range 3 Assign weight for Range 4

Figure 3. Time-Based Anomaly Generation Process.

5.3. Numeric Based Anomaly Generation

Numeric features are used in many of ADG Engine data platforms for a variety of different purposes, including the size of a transaction, a social media post number, visit count, appointment duration, or travel time. The process of generating data for a numeric feature starts by defining the upper and lower bounds of the feature's range and then subdividing it into smaller ranges based on the nature of the numeric feature. As an example, if we look at the features denoting the employment and marital status of the user, we can create ranges of values subdivided from the main range of permitted values, and then assign the ranges a different set of weights depending on whether the user is an employee or not and whether they're married or not. This process can be shown in Figure 4 and is applicable broadly across our data platform. Generally speaking, it is possible to create weighted ranges of numeric values affected by any number of other features of the dataset, allowing great flexibility in numeric feature data generation.

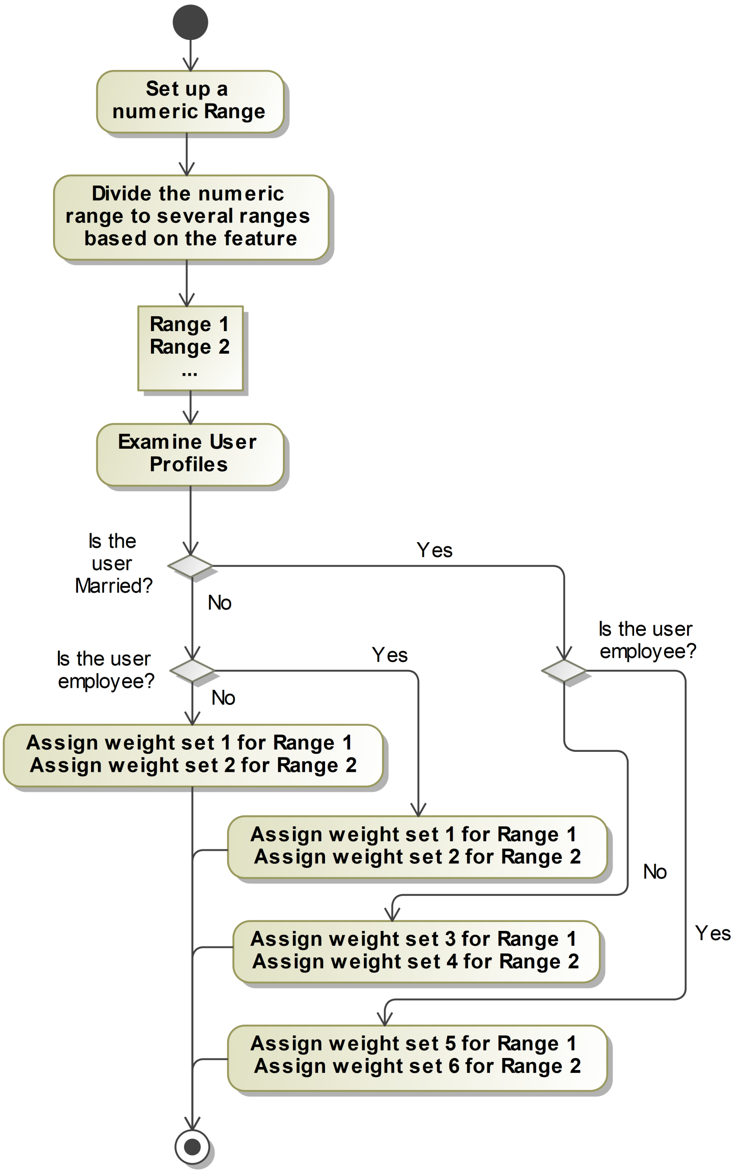
Set up a numeric Range
Divide the numeric range to several ranges based on the feature
Range 1
Range 2
...
Examine User Profiles
Is the user Married?
Yes
No
Is the user employee?
Yes
No
Is the user employee?
Assign weight set 1 for Range 1
Assign weight set 2 for Range 2
Assign weight set 1 for Range 1
Assign weight set 2 for Range 2
No
Yes
Assign weight set 3 for Range 1
Assign weight set 4 for Range 2
Assign weight set 5 for Range 1
Assign weight set 6 for Range 2

Figure 4. Numeric Based Anomaly Generation Process.

5.4. Variable Wights and Fixed Weights

Each set of weights has a size determined by the nature of the weighted feature. For example, the time-based feature described earlier has four possible outcomes and thus four weights. However, it is possible for the number of possible outcomes to be dynamic, changing from user to user. One user may have three credit cards associated with them while another user may have four credit cards. The programmer can specify reasonable limits on the possible number of cards, such as a minimum of 2 and a maximum of 5, and with that, it becomes possible to create a set of weights for each of the possible outcomes. ADG Engine will then ensure it generates a number of credit cards that falls within the predefined boundaries, using the appropriate set of weights to generate the desired data.

6. Experiments Setup

The primary data platform contains information about each user ADG Engine track. Each row of the data platform represents an individual user. It includes the following features: Name, User ID Function, Phone Number Function, Marital Status Function, Employment Status Function, Job, Company, Social Security Number (SSN), Residence, Current Location, Blood Type, Website, Username, Sex, Address, Email, and Birth Date. Details for this data platform are shown in Figure 5.

This data platform holds information that is common to all other data platforms in ADG Engine. For all other tasks in the Engine, it is possible to refer to the qualities of the users listed here. Features like age or marital status will be quite valuable for generating accurate information about this user. There are no anomaly features in this data platform as there are no user actions listed here, but instead, this data platform includes the information necessary to calibrate the anomaly features of the other data platforms.

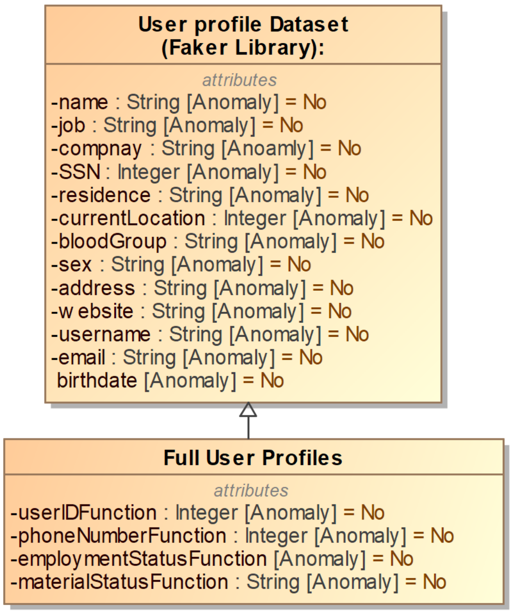
User profile Dataset
(Faker Library):
attributes
-name : String [Anomaly] = No
-job : String [Anomaly] = No
-compnay : String [Anoamly] = No
-SSN : Integer [Anomaly] = No
-residence : String [Anomaly] = No
-currentLocation : Integer [Anomaly] = No
-bloodGroup : String [Anomaly] = No
-sex : String [Anomaly] = No
-address : String [Anomaly] = No
-w ebsite : String [Anomaly] = No
-username : String [Anomaly] = No
-email : String [Anomaly] = No
birthdate [Anomaly] = No
Full User Profiles
attributes
-userIDFunction : Integer [Anomaly] = No
-phoneNumberFunction : Integer [Anomaly] = No
-employmentStatusFunction [Anomaly] = No
-materialStatusFunction : String [Anomaly] = No

Figure 5. General Data Platform Description.

The first of our five Data platforms is credit card activity. Each row of the data Platform represents a single credit card transaction. This data platform includes the following features: name (string), credit card number (integer), transaction amount (float), merchant address (string), merchant name (string), transaction type (string), and time (date). The anomaly features in the credit card activity data platform are credit card number, transaction amount, merchant address, merchant name, transaction category, transaction type, and time. The relationship between this platform and the rest of the platforms can be seen in Figure 6.

The second of our five platforms is bank account activity. Each row of the data platform represents a single bank account transaction. This data platform includes the following features: Name (string), transaction amount (float), time (date), country code (integer), account number (integer), IBAN (integer), SWIFT code (integer), account type (string), and transaction type (string). The anomaly features in the bank account activity data platform are account number, transaction amount, transaction type, and time. The relationship between this platform and the rest of the platforms can be seen in Figure 6.

The third of our five platforms is health records. Each row of the data platform represents a single appointment. This platform includes the following features: Name (string), appointment duration (integer), procedure duration (integer), appointment date and time (date), lateness history (integer), visit type (string), and visit count (integer). The anomaly features in the health records platform are appointment duration, procedure duration, date and time, lateness history, and visit type. The relationship between this platform and the rest of the platforms can be seen in Figure 6.

The fourth activity of our five platforms is telecommunication activity. Each row of the platform represents a single communication. This platform includes the following features: Name (string), starting tower owner (integer), registered home company (string), user routes (integer), starting time (date), travel time (float), starting location (integer), and destination location (integer). The anomaly features in the telecommunications platform are user routes, starting tower owner, starting time, traveling duration, starting location, and destination location. The relationship between this platform and the rest of the platforms can be seen in Figure 6.

The last of our five platforms is social media activity. Each row of the platform represents a single post on social media. This platform includes the following features: name (string), post ID (integer), topic of post (string), time of post (date), comment ID (integer), feedback (string), and time of comment (date). The anomaly features in the social media activity platform are: post topic, post time, and comment time. The relationship between this platform and the rest of the platforms can be seen in Figure 6.

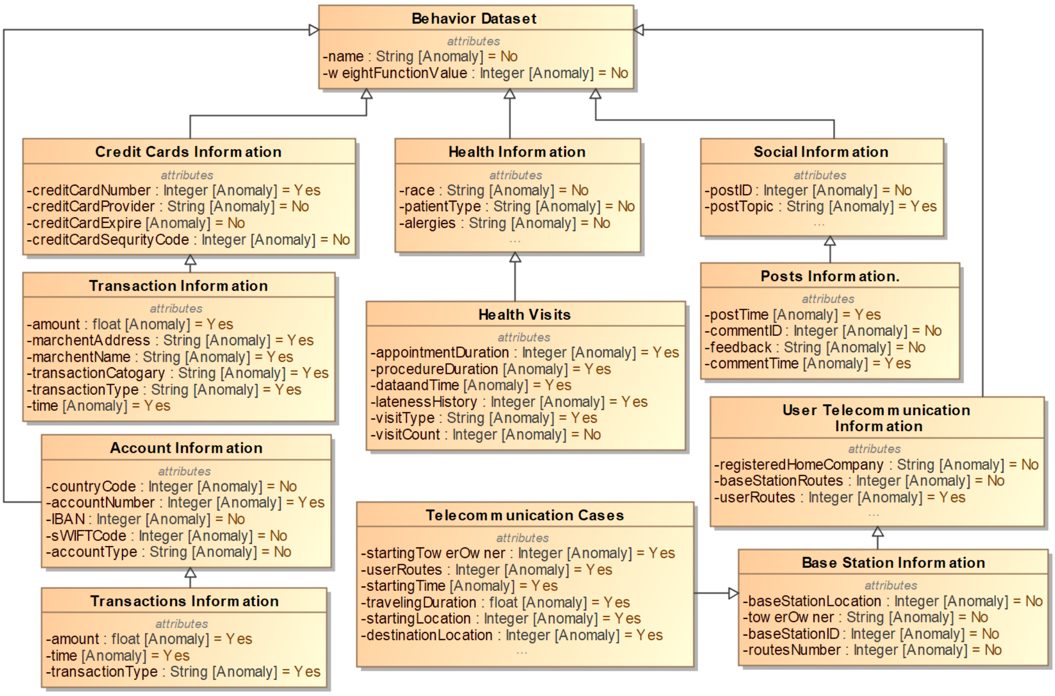
Behavior Dataset
attributes
-name : String [Anomaly] = No
-w eightFunctionValue : Integer [Anomaly] = No
Credit Cards Inform ation
attributes
-creditCardNumber : Integer [Anomaly] = Yes
-creditCardProvider : String [Anomaly] = No
-creditCardExpire [Anomaly] = No
-creditCardSequrityCode : Integer [Anomaly] = No
Health Inform ation
attributes
-race : String [Anomaly] = No
-patientType : String [Anomaly] = No
-alergies : String [Anomaly] = No
...
Social Inform ation
attributes
-postID : Integer [Anomaly] = No
-postTopic : String [Anomaly] = Yes
...
Transaction Inform ation
attributes
-amount : float [Anomaly] = Yes
-marchentAddress : String [Anomaly] = Yes
-marchentName : String [Anomaly] = Yes
-transactionCatogary : String [Anomaly] = Yes
-transactionType : String [Anomaly] = Yes
-time [Anomaly] = Yes
Health Visits
attributes
-appointmentDuration : Integer [Anomaly] = Yes
-procedureDuration [Anomaly] = Yes
-dataandTime [Anomaly] = Yes
-latenessHistory : Integer [Anomaly] = Yes
-visitType : String [Anomaly] = Yes
-visitCount : Integer [Anomaly] = No
Posts Inform ation.
attributes
-postTime [Anomaly] = Yes
-commentID : Integer [Anomaly] = No
-feedback : String [Anomaly] = No
-commentTime [Anomaly] = Yes
User Telecom m unication Inform ation
attributes
-registeredHomeCompany : String [Anomaly] = No
-baseStationRoutes : Integer [Anomaly] = No
-userRoutes : Integer [Anomaly] = Yes
...
Account Inform ation
attributes
-countryCode : Integer [Anomaly] = No
-accountNumber : Integer [Anomaly] = Yes
-IBAN : Integer [Anomaly] = No
-sWIFTCode : Integer [Anomaly] = No
-accountType : String [Anomaly] = No
Telecom m unication Cases
attributes
-startingTow erOw ner : Integer [Anomaly] = Yes
-userRoutes : Integer [Anomaly] = Yes
-startingTime [Anomaly] = Yes
-travelingDuration : float [Anomaly] = Yes
-startingLocation : Integer [Anomaly] = Yes
-destinationLocation : Integer [Anomaly] = Yes
...
Base Station Inform ation
attributes
-baseStationLocation : Integer [Anomaly] = No
-tow erOw ner : String [Anomaly] = No
-baseStationID : Integer [Anomaly] = No
-routesNumber : Integer [Anomaly] = No
Transactions Inform ation
attributes
-amount : float [Anomaly] = Yes
-time [Anomaly] = Yes
-transactionType : String [Anomaly] = Yes

Figure 6. The five data platforms; columns, tables, types, and anomaly factors.

6.1. The Experimental and the Expected Results Comparison

Table 2 describes the total size of ADG Engine platforms after running ADG Engine data generation algorithms. Each data platform has generated a different quantity of entries, ranging from tens of thousands of entries to over a million entries. The quantity of entries here is not a fixed number; running the code for a second time will produce a different observation number for each platform. In other words, the data size is chosen randomly by the algorithm in a specific range which will produce a random number every time the algorithm is executed. For example, the first time the algorithm is run the code might produce 1 thousand observations, but the second run might instead produce 3400 observations, the quantity randomly generated every run within the range designated by the algorithm. During this test, ADG Engine only instructed the minimum observation numbers necessary to guarantee a large number of observations. For example, if ADG Engine assert that the minimum number is 1000, then every time the data is generated the number of observation number will not be less than 1000. Table 2 also shows what proportion of the generated entries are normal behavior and what proportion is abnormal behavior.

Table 2. Data Platforms Comparison and Abnormal Factors.

Dataset Name	Number of Observations	Abnormal/Normal Ration for Columns
Credit Card	82715	6/10
Bank Account	1464810	4/9
Health Records	1428995	5/7
Telecommunication	517810	4/8
Social Media	671946	5/8

6.2. Examples of Several Users in One Sample of Data Platform

Table 3 shows a list of multiple users from the data platform, and the number of observations ADG Engine have from them in the credit card platform. For each user, ADG Engine also generated a sample 'abnormal ratio' by looking at the 'amount' feature of their observations. Both the number of observations and the abnormal ratio for the amount feature is unique for each user in the data platform.

Table 3. Credit Card Data Platform.

Username	Observations Number	Abnormal Ratio for Amount Feature
Sonya Parsons	487	17.86%
Mariah Phillips	1703	15.9%
Ryan Ferguson	364	13.46%
Daniel Williamson	493	13.39%
Joshua York	1120	13.66%

6.3. A Sample of One User Data (Sonya Parsons)

Table 4 looks at one sample user by the name of Sonya Parsons. Sonya has recorded activity in each of ADG Engine five data platforms, with a different number of observations in each platform. This is the input data that we will use to generate normal and abnormal data.

Table 4. One User Observation Number in Every Dataset.

Dataset Name	Observations number
Credit Card	487
Bank Account	15665
Health Records	16096
Telecommunication	114
Social Media	13021

Table 5 shows the creation of an anomaly ratio from input data. In Sonya Parson's recorded data in the credit card platform, ADG Engine looks at the 'amount' feature describing the monetary value of each recorded activity. For simplicity, ADG Engine only divides the feature into two ranges, one for low monetary values and one for high monetary values. Since activity with a low monetary value is much more common than activity with a high monetary value, ADG Engine designates the low monetary value range as the normal section and the high monetary value range as the abnormal section. Table 5 then shows the normal and abnormal ratios derived from these ranges on this platform.

Table 5. Credit Card Platform, 'Sonya Parsons' – Amount Feature.

Amount Ranges	Observations Number
Range1 (0-500) – Normal Behavior	400
Range 2 (500-1000) – Abnormal Behavior	87
Total	487
Abnormal Ratio	(87/487) 17.86%
Normal Ratio	(400/487) 82.14%

6.4. User Information Data

Table 6 shows more information about Sonya Parsons from the user profile platform, which we use in each of ADG Engine five Data platforms to help generate abnormal activities. There is no user behavior in this platform, but this data helps us define what is and is not abnormal for Sonya Parsons.

Table 6. Credit Card Platform, 'Sonya Parsons' – Amount Feature.

Feature Name	Value
name	Sonya Parsons
userID	140736587507472
phoneNumber	1 (488) 882-2393
maritalStatus	S
employmentStatus	non_employee
job	Contractor
company	Boone, Gallagher and Scott
SSN	743 765 026
residence	9074 Brittany Cove Suite 000 South Glendaches...
currentLocation	(2.798248, -56.054469)
bloodGroup	B-
website	[https://www.monroe-hawkins.com/, http://paul....
username	christopher81
sex	F
address	7070 Warner Ridges Suite 228 North Kaylee, ON...
mail	reyesmelody@hotmail.com

7. Results

In this section, ADG Engine examines in detail the sample data from one user across all of ADG Engine five data platforms. The sample user we will be examining is "Sonya Parsons", whose user profile information has already been examined in the previous section. In each platform, ADG Engine will show five sample transactions, and ADG Engine will examine each feature of the platform individually.

7.1. Credit Card platform Sample for One User

In the credit card transaction platform, the first feature as shown in Table 7 is the user's name, which ADG Engine already knows to be Sonya Parsons as ADG Engine are selecting her credit card transactions. The second feature is the credit card number, which is an anomaly feature because it is possible for a user to have multiple cards, one which they normally use and one which they rarely use. The amount feature is an anomaly feature for reasons explained previously, with certain ranges of the numeric value being more common than others. The merchant's name and address are highly correlated features and also together serve as an anomaly feature since it is possible for Sonya Parsons to visit some merchants more often than others. The transaction category feature is an anomaly feature because it is possible for Sonya Parsons to purchase certain types of items more often than others. For example, we see in the five sample entries that "Electronic and Technology Services" and "Toys and Sports" each appear twice, but "hotel services" only appear once. If we extrapolated only from these five samples, we might say that "hotel services" is the abnormal transaction category, but this is only a small sample of the full set of credit card transactions by Sonya Parsons. The last feature is the time of the transaction, which is an anomaly feature where, after sorting the time into one of four times of day, some times of day will be more common than others. For example, four of the five sample transactions take place in the afternoon, but only one of the five takes place in the morning, and none take place at night.

Table 7. Credit Card Platform Sample for "Sonya Parsons".

Name	Credit Card Number	Amount	Merchant Address	Merchant Name	Transaction Category	Transaction Type	Time Prob
Sonya Parsons	36917086006072	[66.91]	9365 Christian Keys Suite 532	Freeman, Davis and Jimenez	['Electronic and Technology Services']	Purchase	2016-02-26 11:36:15
Sonya Parsons	36917086006072	[245.59]	1281 John Pike	Davis-Houston	['Electonic and Technology Services']	Purchase	2016-02-28 13:50:34
Sonya Parsons	36917086006072	[481.06]	USNS Durham FPO AP 47489	Richardson and Sons	['Toys and Sports']	Return	2016-02-22 12:53:45
Sonya Parsons	36917086006072	[219.92]	667 Chelsea Mountains Apt. 243 Morenoberg, UT...	Thompson-Guzman	['hotel services']	Return	2016-02-15 12:43:44
Sonya Parsons	36917086006072	[216.87]	82929 Annette Shoals Thompsonshire, WY 88227	Wright PLC	['Toys and Sports']	Payment	2016-02-16 14:03:15

7.2. Bank Account Platform Sample for One User

Table 8 shows the bank account transaction platform. The first feature is again the user's name, which we know to be Sonya Parsons for the purposes of this sample. The first feature is the account number, which is an anomaly feature because a user may have several bank accounts in their name and use one more than another. This is not shown in this sample because all five sample data entries from Sonya Parsons used the same account number. In the same manner, as before, the amount feature and time feature are both anomaly features. The last anomaly feature of this platform is the transaction type, which similarly to the previous platform is an anomaly feature because it is possible for one transaction type to be more common than others.

Table 8. Bank Account Platform Sample for "SONYA PARSONS".

name	accountNumber	amount	time	transactionType
Sonya Parsons	PWIC38335764390619	118.47	1998-02-16 18:51:41	Deposit
Sonya Parsons	PWIC38335764390619	56.64	2003-12-08 21:46:40	eDeposit
Sonya Parsons	PWIC38335764390619	78.13	2012-06-11 17:08:29	eTransfer
Sonya Parsons	PWIC38335764390619	236.94	1995-07-18 16:29:35	Bill_payment
Sonya Parsons	PWIC38335764390619	518.52	1999-04-21 04:46:51	Withdraw

7.3. Health Records Platform Sample for One User

In the health records platform, as described in Table 9, the user registers their health information and then visits the clinic for their appointment. The second feature in the platform is 'appointment duration', and it is an anomaly feature because, much like the 'amount' features of previous Platforms, ADG Engine can divide the data into different ranges, some of which are more common than others. The duration is measured in minutes, so if appointments typically last two or three hours then a very short appointment would be abnormal. The procedure duration feature is separated into three or four types and is not considered to be a meaningful anomaly feature. The 'appointment date and time' feature is an anomaly feature in the same manner as previous time-based anomaly features. 'Lateness history' tracks in minutes how late the user is and is an anomaly feature because if a user is not late most of the time, then it is unusual for them to be late. 'Visit type' is an anomaly feature because it is possible for Sonya Parsons to visit the clinic for some reasons more often than other reasons, such as "Dermatologist" which appears in two of our five sample entries. The last feature of the platform, 'Visit Count', simply tracks which visit number to the clinic the current visit is, and is not considered an anomaly feature.

Table 9. Health Records Platform for "Sonya Parsons".

name	appointment duration	procedure duration	time	lateness history	visit type	visit count
Sonya Parsons	118	Brief	2003-02-28 13:09:45	absent	Eye Doctor	226
Sonya Parsons	156	Extended	2007-04-11 04:25:45	absent	Registered Nurses	679
Sonya Parsons	177	Intermediate	2019-10-05 07:52:27	10	Dermatologist	197
Sonya Parsons	188	Brief	2015-03-04 02:57:55	0	Mental Health Professionals	786
Sonya Parsons	1	Extended	2020-05-04 23:56:16	0	Dermatologist	740

7.4. Telecommunication Platform Sample for One User

In the telecommunication activity platform, the first feature describes which base station the telecommunication begins at and will always be the same if the user starts the telecommunication at the same geographical point, but because the user can start telecommunications from a variety of places this is an anomaly feature where some starting base stations can be more common than others. The telecommunication company the user is registered to is not considered a meaningful anomaly feature, as the user remains registered to the same company for all their telecommunications with a change in registered company being rare and infrequent. The majority of the unusual work for this platform pertains to the 'user routes' feature, which we modeled after base stations in London, Ontario. ADG Engien assigned every base station in London, Ontario with an ID number increments up from zero. Once every base station had an ID, ADG Engine created a variety of routes that connect various base stations together in sequence and then assigned each user three to five random routes. One of these routes is the user's routine route and the other routes are uncommon routes. For example, in Sonya Parson's sample platform, the common user route is [0, 17, 9, 7, 3, 26, 5] and the anomalous user route is [0, 27, 15, 22, 24, 8, 3, 2]. As before, the 'starting time' feature is an anomaly feature because some times of the day can be more common than other times of the day. The 'traveling time' feature is derived from the 'user route' feature by calculating how many base stations need to be passed through and is primarily a matter of simplicity. The starting location and destination location features are both anomaly features that refer to the initial and final base stations in the routes. Both features can substantially vary and have one starting point or destination occur more commonly than others.

Table 10. Telecommunication Platform for "Sonya Parsons".

name	starting tower owner	registered home company	user routes	starting time	traveling time	starting loc	destination loc
Sonya Parsons	SaskTel	Shaw Communications	[0, 17, 9, 7, 3, 26, 5]	2016-03-04 04:45:51	[26.93]	[10]	5
Sonya Parsons	SaskTel	Shaw Communications	[0, 27, 15, 22, 24, 8, 3, 2]	2016-03-05 01:06:15	[88.83]	[10]	2
Sonya Parsons	SaskTel	Shaw Communications	[0, 17, 9, 7, 3, 26, 5]	2016-02-24 13:20:22	[82.38]	[10]	5
Sonya Parsons	SaskTel	Shaw Communications	[0, 17, 9, 7, 3, 26, 5]	2016-02-28 04:41:17	[138.65]	[10]	5
Sonya Parsons	SaskTel	Shaw Communications	[0, 17, 9, 7, 3, 26, 5]	2016-02-19 20:46:41	[82.5]	[10]	5

7.5. Example of User Rout through London Ontario base stations

Figure 7 shows an example of a user route in London, Ontario. The blue line depicts the course of the user route, beginning at the user's starting point and ending at the user's destination. The user route also passes through three base stations along the way, extending the route to what is shown in the picture. The GPS coordinates of these base stations are as follows: (-81.314165, 43.019989), (-81.299122, 42.909328), (-81.616511, 42.955678).

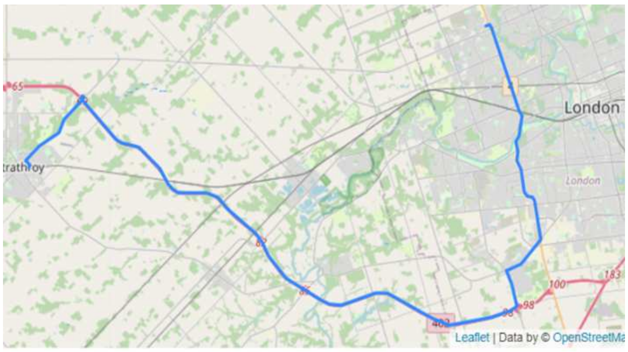

Figure 7. London Ontario User Route Example.

7.6. Social Media Platform Sample for One User

In the social media activity platform, ADG Engine tracks elements such as post information, comments, and post feedback. The Post ID is identical in all five of Sonya Parson's sample events because that post received multiple likes and dislikes, each of which is treated as a separate event. The post topic is an anomaly factor, because a user may have one topic they talk about more often than other topics. Like before, time-based features such as 'post time' and 'comment time' are anomaly features for this platform. The feedback feature is considered an anomaly feature because different users have a different ratios of likes and dislikes, rendering one more normal than the other.

Table 11. Social Media Platform for "Sonya Parsons".

name	postID	postTopic	postTime	commentID	feedback	commentTime
Sonya Parsons	140736587507472	religion	2016-03-04 16:18:01	140736587507472	['dislike']	2016-02-22 17:29:18
Sonya Parsons	140736587507472	religion	2016-03-04 16:18:01	140736587507504	['like']	2016-02-10 17:38:33
Sonya Parsons	140736587507472	religion	2016-03-04 16:18:01	140736587507536	['like']	2016-03-03 10:12:11
Sonya Parsons	140736587507472	religion	2016-03-04 16:18:01	140736587507568	['dislike']	2016-02-09 13:49:23
Sonya Parsons	140736587507472	religion	2016-03-04 16:18:01	140736587507600	['like']	2016-03-01 16:10:41

8. Conclusions

In this paper, we created an Algorithm-based Data Generation Engine (ADG Engine) capable of efficiently generating large quantities of data to match the trends and qualities of any data that are already present. We took five platforms containing different user behaviors and linked them together into a rational platform with a user profile to reflect the fact that individual users will not use only one major service in their daily life. We used this rational platform to determine the trends of the data within, using the user profile platform in common with all five activity platforms to help in the analysis, and we used these trends to generate new data for the platforms in a nondeterministic fashion, expanding it to whatever size we deem fit. Our results show that the model matched all the criteria explained in Table 1. This shows that the generated data has many aspects that simulate real-life data and can substitute it for our research purposes. Future development of this work could include further interconnectivity of features in the platform, such that the normal/abnormal status of each feature can affect whether other features are generated as normal or abnormal. This would allow for the expression of higher-level trends while retaining the value of the current anomaly ratios produced by this project.

doi:10.20944/preprints202306.1378.v1

References

1. Patki, N.; Wedge, R.; Veeramachaneni, K. The synthetic data vault. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2016, pp. 399–410.
2. Lopez-Rojas, E.A.; Axelsson, S. Banksim: A bank payments simulator for fraud detection research. In Proceedings of the 26th European Modeling and Simulation Symposium, EMSS, 2014, pp. 144–152.
3. Zhao, H.; Yang, Y. A data generation algorithm for internet of things based on complex event processing. In Proceedings of the 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity). IEEE, 2015, pp. 827–831.
4. Hu, M.; Wang, H.; Tang, D.; Li, F. Research on uncertain data generation algorithm. In Proceedings of the 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA). IEEE, 2017, pp. 120–123.
5. Kim, J.S.; Jin, H.; Kavak, H.; Rouly, O.C.; Crooks, A.; Pfoser, D.; Wenk, C.; Züfle, A. Location-based social network data generation based on patterns of life. In Proceedings of the 2020 21st IEEE International Conference on Mobile Data Management (MDM). IEEE, 2020, pp. 158–167.
6. Cinquini, M.; Giannotti, F.; Guidotti, R. Boosting Synthetic Data Generation with Effective Nonlinear Causal Discovery. In Proceedings of the 2021 IEEE Third International Conference on Cognitive Machine Intelligence (CogMI). IEEE, 2021, pp. 54–63.
7. Kothare, A.; Chaube, S.; Moharir, Y.; Bajodia, G.; Dongre, S. SynGen: Synthetic Data Generation. In Proceedings of the 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA). IEEE, 2021, pp. 1–4.
8. Hu, J.W.; Bowman, I.T.; Nica, A.; Goel, A. Distribution-driven, embedded synthetic data generation system and tool for RDBMS. In Proceedings of the 2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW). IEEE, 2019, pp. 113–115.
9. Topal, A.; Amasyali, M.F. When does Synthetic Data Generation Work? In Proceedings of the 2021 29th Signal Processing and Communications Applications Conference (SIU). IEEE, 2021, pp. 1–4.
10. Imtiaz, S.; Arsalan, M.; Vlassov, V.; Sadre, R. Synthetic and private smart health care data generation using GANs. In Proceedings of the 2021 International Conference on Computer Communications and Networks (ICCCN). IEEE, 2021, pp. 1–7.
11. Ouda, A. A framework for next generation user authentication. In Proceedings of the 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC). IEEE, 2016, pp. 1–4.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.