Visual Analytics Framework for High Dimensional Data Streams

(Proposal for)
Master Thesis
Software Systems Engineering

Khan Hafizur Rahman
Matriculation number 360142

2018-07-06

Supervisors:
Prof. Dr. Matthias Jarke
Prof. Dr. Christoph Quix

Advisors:
Arnab Chakrabarti

Eidesstattliche Versicherung

Name, Vorname

Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum

Unterschrift

Acknowledgements

You should add some acknowledgements: thank your advisors, supervisors, parents, family, colleagues, students, etc.

Abstract

You can add an abstract if you like.

Contents

Chapter 1

Introduction

With the development of new technology and technology oriented devices now the data produces continuosly in a heterogeneous structure in every single second. According to statistics of youtube, in every minute hours of video uploaded to YouTube is 300 hours equivalent to watch [?]. Recently, sensors and Internet of Things devices produce data in a continuous fashion with each and every single second. As a consequece, we are now having a wide variety of large, complex, high-dimensional dataset. Data scientist and researchers are doing research on finding optimum way of collecting data, storing data, analysing data, proposed new algorithm for data mining, finding training methods to train the data, visualizing data and many more. Due to the various characteristics of the data in respect of size, structure, format there has been always challenging issue arises in the existing field and also new research domain being created.

We are living in the world called "Big Data". These Big Data have diversified sources such as images, videos, different social sites activities like post status, comments etc. Moreover now-a-days a lot of data is generated through sensors applied in different fields, GPS signals and many more [?]. The term "Big Data" is first mentioned on the IEEE 8th conference on visualisation in year 1997 [?]. However, since 2011 the interest in big data area had been increased exponentially [?]. The word "Big" means significance, complexity, challenge, quantification.

There are many definition for the Big Data but unfortunately there is no exact established definition throughout the whole world wide. The most used one is from Gartner report [?]. According to that, data those can be termed the "three Vs": volume,variety and velocity are known as big data. Volume means the significant amount of data produces from the source. Data are produced in a structured, semi-structured or sometimes non-structured way which defined by the term variety. Velocity is the rate at which data is being produced. Later, two new addidational "V"'s are added known as Veracity and Value. Veracity is for the trustworthiness of the data and value determines the business value add in the dataset.

Data can be modelled based on some characteristics. Traditionally, data is modelled as persistent relationships and store into database management system. From the system, data can be accessed and analysed based on the requirements. But recently a new class of data intensive application evolved where application does not follow the traditional system rather considered as transient data streams. Examples of such application fields include financial applications, web page visits, network monitoring, security, telecommunication data management, sensor network and many more [?]. In data streams, data produce continously in multiple structured with the flow of the time. Datas are unbounded and unpredictable as well. Due to having this unique characteristics; a new approach is required to capture, storage, transformation data.

In many application field real time analysis is very much important. For example, in medical domain there are many diagnostic data has been generated from different machine where doctors need to analyse the data and take decision in real time. In stock market, price are continuously changing as time changes and the traders need to analyse those data and take decision of doing transaction in real time. The same type of real-time analysis is also needed with the sensor data attached with the internet of devices or the meteorological data and space data.

In present world, each data set has a huge number of features or dimensions. In general, the features are those which help to describe the data. These are called as "High-Dimensional Data" and in almost every fields of study starting from meterology to economics field now data are high dimensional. The computational complexity and storage complexity increases with the increament of the dimensions. Moreover, if a data is high dimensional then it is also difficult for the human being to reflect their intuitions to percept the knowledge from the data. Although each feature has the important information of the data set but still we can compare among the features based on the important information and skip the less feature one and this process is known as dimensionality reduction.

The concept of using pictures to understand data has been around for the centuries and the process to present data in forms of pictures or graphs is known as data visualization. With the help of visualisation it is easy to get the inner meaning of the data. Though earlier visulaization is only used for the communication but now the visualization is also helpful to achieve the inner meanings of the data. It helps us to analyse the data and present in form of patterns, trends, gaps and outliers. It also helps us to compare, make correlations among the data. One of the most important benefit of visualisation is that it encompasses various data set quickly, effectively and efficiently. Scientifically the effectiveness of data visualization is to maintain a proper balance between perception and cognition through visualisation.

In last few decades, huge number of efforts have been introduced for data visualization. Many tools have already been published for the effective visualization but still it is one of the open research field for visualization researcher to build an effective tool for high dimensional data visualization. Some of the most conventional visualization tools are Histogram, x-y plots, line plots, scatter plots, Venn diagram, pie charts etc. These visualization method are not suitable enough for high Dimensional Data [?]. Though some new methods like Tree map, paralel coordinates, Heat Map are emerged as extensions to the convolutional method but still these techniques are far from the target and also suffer because of high dimensions. Due to the lack of proper high dimensional data visualizaiton tool one of the prominent way is to reduce the dimensions efficiently to preseve information as much as possible and then visulaize the result with any of the extending visualization tool.

Though High Dimensional static data have been studied out for the last few decade but still there is no far research has been carried out for the dynamic or streaming data. There are many algorithms for high dimensional data which works fine for the static data set but the performance deteriotes for the streaming one. Moreover, the dimensionality reduction method is not evaluated from visualization perspective in streaming area. In this thesis, this problem is identified and will try to provide a solution for dimensionality reduction in streaming data and also evaluate through visualisation technique.

1.1 Motivation

Due to the growth of streaming data it is now demand of the some application domain to analyse the data and present data to the end user in quick response. For example, in stock market data produce continuously about telling the price, volume of the stock etc; if a system can analyse the pattern of the data, find correlation among the features and present them intuitively to the end users then users will be able to take a quick decision to buy the perfect stock at that time and do profit more.

The necessity of visulaization in this kind of scenario opens up the challenge for the researcher to present the user friendly, intuitive, interactive, easily understandable visulisation to the end user. There are three main transforamtion steps to achieve such a effective visualization [?]. These are:

1. Data transformation
2. Visual mapping
3. View transformation

Data transformation is the process of transforming source data to suitable vesrion of the data for carrying further process. There are many ways to transform data and Dimensionality reduction is one of them. Moreover, dimensionality reduction is not only the effective for the visualization but also helpful to build many effective data model. It is also helpful for any learning algorithm.

There are two different ways to reduce dimensions from the original dataset. The reduction can be done either by selecting significant features and form a subset of the original set. The other way to reduce it by transforming dimension in order to get a new one or reduced set of dimensions. The first one is called the Feature Selection which is greedy in nature where the later one is known as Feature Extraction. Due to the nature of Feature Selection it is always a challenge an optimal solution or the subset based on some defined criteria. Orthogonal Centroid Algorithm is a Feature Selection algorithm.

Feature Extraction algorithm aim to extract features by projecting high-dimensional space into lower dimensional space using algebraic expression. The feature extraction algorithm can be further divided into linear projection and non-linear projection. Linear Discriminant Analysis (LDA), Maximum Margin Classification (MMC) & Principle Component Analysis (PCA) are linear feature extraction algorithm [?] where non-linear algorithm are kernel PCA, graph-kernel PCA etc.

The conventional dimensionality reduction algorithm use Gaussian Maximum Likelihood estimation which involves different matrices like covariant matrix, scatter matrix which have time & space complexity of *O(n^2)* or *O(n^3).* The complexity can be tolerable if the data size is small but if the data sample is more than 20000 then this method does not perform well [?]. Due to the high mathematical computation it is not suitable to use those methods on streaming data specially in real time analysis field. The time increased rapidly for streaming data because everytime new data comes it starts from the scratch.

One of the solution is to propose incremental version of feature extraction algorithm. The term "incremental" means learning from new data without forgetting the prior knowledge. In this mechanism, a system must acquire knowledge from the past data without keeping the original data and also ready to handle the new data [?].Incremental Principle Component Analysis(IPCA) [?], Incremental Linear Discriminant Analysis [?], Candid Covariance-free incremental principle component analysis [?], Incremental Dimension Reduction via QR decomposition [?] etc. The major drawback is still the involvement of numerical transformation which performs very poor where there is a lot of data & dimensions. The details of each algorithm will be covered on the related work section.

In all of the incremental algorithm researchers use static data sets. They divide the data set into two portions where first portion they used to calculate all the mathematical computation and remember values irrespective of the dataset and later they used the remaining portion to update those values incrementally and see the final output of the algorithm. In our thesis, we will use time based streaming data having a fixed time window which is unpredictable in nature and also data will not available after the time pass. The existing use of data set is controlled by the user and also continuous update of visualization is interrupted. Moreover, the performance of any kind of incremental dimensionality reduction is not evaluated through visualization perspective.

Though dimensionality reduction has many advantages but it has some demerits too. The main drawback is the possibility of information loss. It is undoubtedly true that when we left of one dimension that means we have the trade off of loosing some data but in some cases we have lost the important information. The essential idea of dimensionality reduction is to preserve the instrinsic meaning of the data by keeping similar data points close and maintaining a distance among dissimilarity data. Researchers is now also looking for different approaches to get the same or better visualisation as ouput without doing information loss. One of them is reordering the columns based on similarities. Djuric et.all proposed an algorithm based on the reordering approaches in their paper [?].The details will be covered in the "Entropy Minimisation ordering" of the related work section.

1.2 Goals of the thesis

The aim of this thesis is to develop an effective visualization framework for Stream data. In this framework, we can select the high dimensional datasets from different domains and reduce the dimensions through dimensionality reduction algorithm. For dimensionality reduction we will use the incremental version of traditional dimensionality reduction algorithm known as Linear Discriminant Analysis. The another prime concern will be regarding the information loss due to the dimensionality reduction. Our framework will be evaluated regarding the information loss and also some more evaluation criteria. The visulaisation will be presented to the end user through HeatMap; one of the most used prominent tool for high dimensional data visualization technique.

1.2.1 Selection of Dataset

In this thesis, we are looking for the high dimensional data set.There is no fixed threshold number of the dimension what can be termed as high but while we look for the data set we considered from visualisation perspective and ensure data set are high dimensional. If required we can convert the batch data to streaming data using any kind of state on art tool.

1.2.2 Define Data Model

The main difference between static data and streaming data is in static data we can save data into the disk and access that whenever we need that for any purpose but in streaming data the situation is not the same. In streaming data hence velocity of the data is huge so we can not save the data rather we need to do operations with the flow of data. Sliding window structure means we will divide the data in a block and only have access for once; there is no scope of viewing the data for the second time. The block can be done based on the time interval.

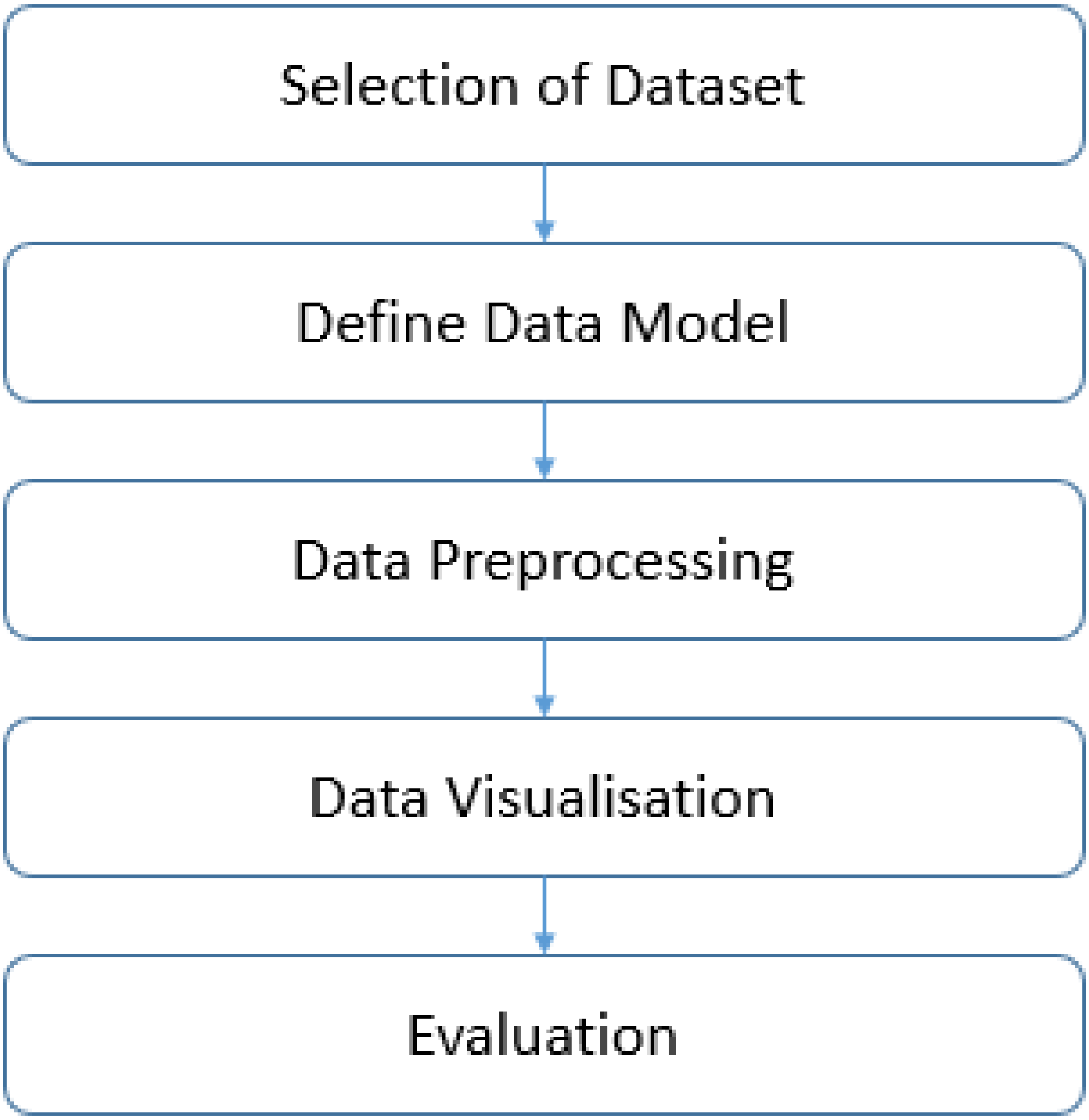


Figure 1.1: Thesis Goals

1.2.3 Data Preprocessing

Data preprocessing includes a lot of steps to be done before using the data. We will only focus on dimensionality reduction in this thesis and this is the core part of the thesis. There are many algorithms has been developed by researchers regarding Linear Discriminant Analysis and all of them either use the scatter matrix or QR decomposition under the big picture. Recently, one algorithm use Cholesky-decomposition to implement Linear Discriminant Analysis which is comparatively faster than others in theory but there is no open source of that code. Moreover, the algorithm is used for batch data set in an incremental way but we will directly imply the algorithm on stream data.

1.2.4 Visualization

Visualization is the beginning of the output section of the thesis.There are many state-on-art visualization techniques for visualize high dimensional data. We will visualize the output of the framework through one of the common procedure known as Heatmap.

1.2.5 Evaluation

There are basically two ways to evaluate the system. One is known as the quantitative evaluation and the other is qualitative evaluation. In Quantitative evaluation we define the metrics for example calculating the entropy or the matrix reordering quantitatively. Qualitative evaluation is the evaluation for example after showing the visualization by deciding which one is more informative. This can be done by doing user studies or selecting streaming data from different fields.

In our thesis we will follow the qualitative evaluation part by calculating different kind of matrices after getting the reduced dimensional data.

The remaining part of this paper is organized in the following manner: Section 2 will decribe the related work about dimensionality reduction, it will be followed by the solution of the thesis. Section 4 is for the evaluation and last section is the timeplan of the thesis.

Chapter 2

Related Work

There are three different types of complexities to handle big data systems [?]. These are:

- Data complexity
- Computational complexity
- System Complexity

Data complexity arises due to the multiple formats & unstructured of the data. Moreover data have huge number of dimensions and also there is a complexity between inter-dimensional and intra-dimensional relationships. The data complexity also increase the computational complexity in big data systems. Moreover,The extensive computational requirements of big data systems increase the system level complexity.

In the following two subsections we will describe the details of the past work of dimensionality reduction algorithm and Entropy Minimisation based reordering algorithm respectively.

2.0.1 Related work with Dimensionality Reduction

Efficient storage and retrieval of high-dimensional data certainly one of the major issue in database and data mining research [?]. In the past, many attempts took to design multi dimensional indexing structure for example R-trees, R^*-trees, X-trees, SR-tree, etc. in order to speed up the query procedure. But these procedures fail to do so when the number of dimensions increases. The more is the number of dimension the more performance deteriorates. To overcome this issue, one way is to transform from high dimensional data to low dimensional data with the trade-off of limited information loss.

Though Dimensionality reduction is a very old problem but still this problem exists. Principle Component Analysis(PCA) and Linear Discriminant Analysis(LDA) are two mostly used one to reduce the number of dimensions. Both PCA and LDA seek directions of the component but the main difference among them is PCA seeks direction for representation where LDA seek directions for efficient discrimination. In both cases it assumes that training data set are available in advance but in reality a complete training set might not given beforehand.

Principle Component Analysis

PCA is a multivariate technique to analyze the data table. Here the observations are described by several inter-correlated quantitative dependent variables [?]. The goal is to find a subspace whose basis vectors correspond to the direction with maximal variances [?]. Mathematically, it depends on eigen decomposition by computing eigenvectors and eigenvalues and upon singular value decomposition (SVD) of rectangular matrices.

Lets denote $C = \frac{1}{n}\sum_{i=1,2,\ldots,n}(x_i - m)(x_i - m)^T$ as the covariance matrix of the sample data. We define the objective function as *J(W)* $=$ *trace*W^TCW. PCA aims to maximise the objective function *J(W)* in a solution space $H^{d*p} = W \in R^{d*P}, W^TW = I$.

Linear Discriminant Analysis

The goal of the Linear Discriminant Analysis (LDA) is used to find a lower dimensional space that best discriminants the samples from different classes [?]. In Linear Discriminant Analysis (LDA), we need to compute a linear transformation by maximising the ratio of the between-class distance to the within-class distance in target of achieving maximal discrimination [?]. After that we need to find eigen value decomposition of both matrix to select new feature subspace. Mathematically the aim is to maximize the Fisher criterion an objective function:

$$J(W) = \frac{W^TS_bW}{W^TS_wW} \tag{2.1}$$

where $S_b = \sum_{i=1}^{c} p_i(m_i - m)(m_i - m)^T$ and $S_w = \sum_{i=1}^{c} p_i \underset{x\in c_i}{\mathrm{E}}(x - m_i)(x - m_i)^T$ are called Interclass scatter matrix and Intraclass scatter matrix respectively. Here E denotes the expectation and $p_i = \frac{n_i}{n}$ is the prior probability of class i. We can get W by solving $W^* =$ arg max J(W) in the solution space $H^{d*p} = W \in R^{d*p}, W^TW = I$. This is done by solving the generalised eigenvalue decomposition problem: $S_bw = \lambda S_ww$

Limitation of PCA & LDA

By considering the availability of the data before applying algorithm is the main challenge in stream data application. In streaming data, data comes in a lot of numbers and continuous speed as a result it is not possible to store the data before applying algorithm. Moreover the traditional LDA and PCA perform in batch mode which is computationally expensive when dealing with large scale problems. In streaming data if there is a new data then both the LDA and PCA algorithm starts from the scratch to learn it from beginning which increase the computational complexity and large memory [?].

Therefore, researchers look for the solution and one of the way to get rid of this to collect data whenever new data are presented and apply batch learning approach for the collected data so far. But the drawback of this approach is that it requires a large memory to store the data and high computational expenses are required. Moreover the system also forget about the knowledge that it acquires in past batch mode. Researchers work on this issue and some of their work are presented in the following section.

The another main challenge in streaming data is data may come into chunk from and also may be a single data in allowed form. That means the rate of data passing is unpredictable in nature therefore the dimensionality reduction algorithm should be adaptable with this issue.

Principle Component Analysis for Streaming data

Traditionally PCA efficiently represent high dimensional vectors with a small number of orthogonal basis vectors but this method is usually perform in batch-mode which is computationally expensive for large scale problems. To address this issue, researchers developed several incremental algorithms in their previous studies [?].

Artac Jogan et. al [?] describes the way of remembering data from the sample and then delete the data to optimize the storage complexity. They used image as input where the representation of the image consists only of the corresponding coefficients stored as per image then the image is discarded. Here the performance is almost similar with the batch method but the learning method helps us to relearn data. Hall et.all [?], Chandrasekaran et.all [?], DeGroat et.all also propose based on gaining information from the past and learn from the past and delete the data after acquiring the knowledge.

Li Xu et al. proposed an algorthm by removing the outliers. The estimation is done using the likelihood function. All incremental PCA algorithm proposed so far is how to effectively handle with the covariance matrix but all of their effectiveness and computational complexity is more or less similar. Weng et all. [?] proposed a candid covariance free incremental analysis by using a well-known statistical concept efficient estimate like some well-known distribution for example Gaussian distribution. This algorithm also compute the principal component of samples incrementally without estimating the covariance matrix by keeping the scale of observations and computes the mean of observations incrementally. The main problem of this technique is to run into convergence problems in high dimensions.

Linear Discriminant Analysis for Streaming data

Similar to IPCA researchers also modify the LDA algorithm to run an incremental fashion to accomodate the new data. Here also one of the major concern is not forgetting prior knowledge.

Pang et. all [?] propose the algorithm by adopting the system ready for any new data arrival in basis of single or chunk basis and termed separately sequential incremental LDA and chunk incremental LDA. In this paper they handle the eigenvectors by providing ranking and select the top most eigen vectors. This proposed algorithm will confront difficulty as the dimension if the data is very high. Specially it will require large memory hence it needs to solve a high dimensional generalised eigen value problem [?]. Ye et.al [?] proposed an algorithm called Streaming LDA consisting three steps: first it compute the centroid matrix then update within-class scatter matrix. Finally the between-class scatter matrix is updated. After this steps, it also solve the eigenvalue problem.

kim et.all [?] propose a new concept to update between-class and within-class scatter matrix. They used sufficient spanning set to do so. In every step both matrices are kept and updated and minor components are removed in every step.

The main difficulty in all of the propose solution is the presence of the eigenvalue problem of scatter matrices which makes it difficult to maintain it incrementally.

QR decomposition

LDA algorithm use Singular Value Decomposition(SVD). It is difficult to design an incremental solution for the eigenvalue problem on the product of scatter matrices.

To solve this, Ye Li et.al [?] propose an LDA based incremental dimension reduction algorithm called IDR/QR which applies QR decomposition rather than Singular Value Decomposition. The reason for using this technique is it does not require the whole data sets in memory before implementation. The algorithm is also computational cost efficient when new data item is inserted. The classification accuracy of this algorithm is very close in compare with the other best described LDA algorithm but it has much less cost when new items are inserted compare to others. Moreover hence it is computed some approximate matrices there is a chance of accumulating the approximate error as new data are appended sequentially. The larger the error the more the opportunity of information loss [?].

Fast Online incremental learning on mixture streaming data

In [?] they proposed an algorithm for streaming data known as Fast Batch LDA algorithm known as FLDA/QR learning algorithm. In the algorithm they use the cluster centres to solve a lower triangular system which is optimized by the Cholesky-factorization. They also develop this algorithm for an exact incremental algorithm called IFLDA/QR. For reorthogonalization they use the Gram-Schmidt process which saves the space and time expense compared with the rank-one QR-updating of most existing methods. In there paper they mentioned their contributions are twofold:

- They use the advantage of the QR-decomposition on a lower triangular matrix and propose a new fast batch method called FLDA/QR. In this algorithm, they take the centroid of each cluster to constitute the matrix decomposition. Because of this, they require a smaller storage and less computation. They also use the Cholesky-factorization which surpass in performance specially the computation load of the FLDA/QR method.
- Next they develop an exact incremental version of the FLDA/QR known as the IFLDA/QR. It is mathematically possible to update the Gram-Schmidt reorthogonalization process. According to them, this process is faster than the rank-one updating in many other ILDA algorithms based on QR-decomposition.

The algorithm described in a paper is presented in the following figure 2.1 as a flow-chart. The key characteristic of their algorithm which separates it from the other is the space and time complexity. It is space and time efficient compare to others while updating the algorithm because of the arrival of the data. This algorithm is 2 to 10 times faster than the state-of-the-art algorithm where the classification performance is same like the others.

Fast Batch Linear Discriminant Analysis: After having the data matrix as an input first the algorithm try to compute the global centroid matrix C consisting of the K centers. In this algorithm, lower triangular linear system is used for the calculation. Since the input training data is the centroid matrix therefore the scatter matrix within-class scatter is 0. They used the Cholesky-factorization to generate the matrix of R for the QR decomposiiton. The algorithm complexity is *O(dn)*

Incremental Linear Discriminant Analysis: After development of the FLDA algorithm they extend the algorithm for the incremental development of the new incoming streaming data.The streaming data may come in three different forms:

1. new labeled samples to the existing classes
2. samples from an entirely new (novel) class
3. a chunk of samples mixed with those as 1) and 2)

In cases 1 & 2, the samples from the existing classes and from an entirely new classes the updated calculation of the new centroid matrix is same. The r,α,q is also calculated in the same way for both case. The difference lies within the update G$_c$, Q$_c$ and R$_c$. The update version of any *variable* is represented as $\widehat{variable}$. If the the new labeled samples from the existing classes then the Q_c & R_c are updated using equation 2.2

$$\hat{C} = Q_c * R_c \tag{2.2}$$

For new novel classes data are updated by the equation 2.3

$$\hat{G_c} = \left[G_c - qc_{new}{}^T G_c/\alpha \quad q/\alpha\right] \tag{2.3}$$

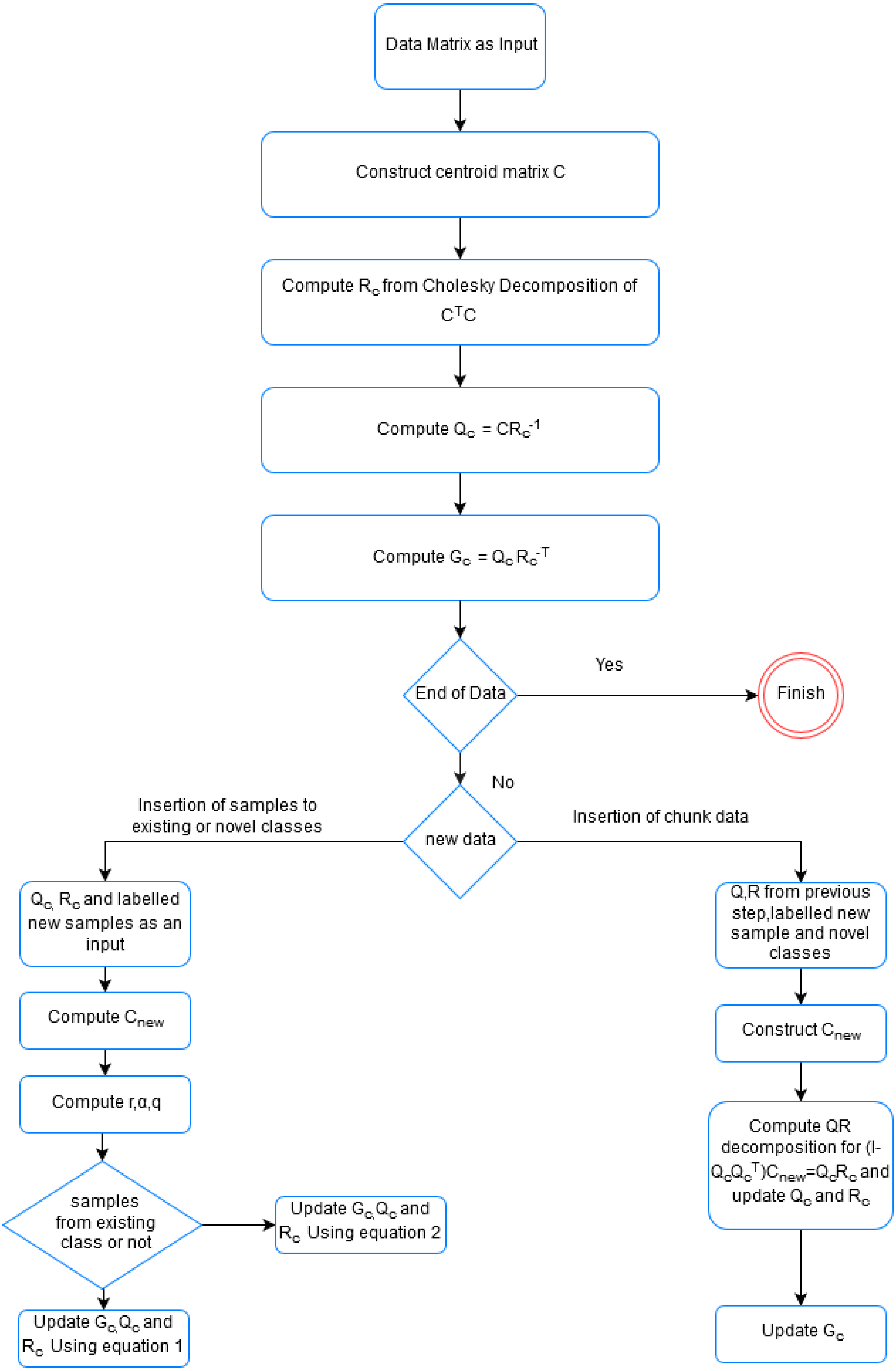

Data Matrix as Input
Construct centroid matrix C
Compute R_c from Cholesky Decomposition of C^TC
Compute $Q_c = CR_c^{-1}$
Compute $G_c = Q_cR_c^{-T}$
End of Data
Yes
Finish
No
new data
Insertion of samples to existing or novel classes
Insertion of chunk data
Q_c, R_c and labelled new samples as an input
Compute C_{new}
Compute r,α,q
samples from existing class or not
Update G_c, Q_c and R_c Using equation 2
Update G_c, Q_c and R_c Using equation 1
Q,R from previous step,labelled new sample and novel classes
Construct C_{new}
Compute QR decomposition for $(I-Q_cQ_c^T)C_{new}=Q_cR_c$ and update Q_c and R_c
Update G_c

Figure 2.1: IFLDA/QR algorithm

There may be a chunk of new data which contain samples from the existing classes and novel classes. The challenge in this case is to extract the information from these mixed data and also they need to preserve the previous learned ones. The current algorithms fail to perfom in this scenario. After constructing C_{new} they compute the QR decomposition of $(I - Q_c{Q_c}^T)C_{new} = \hat{Q}_c\hat{R}_c$ and update $\hat{Q}_c$ and $\hat{R}_c$. Update the $\hat{G}_c$ using equation 2.4

$$\hat{G}_c = \Big[G_c - \hat{Q}_c({\hat{R}_c}^{-T}({C_{new}}^T)G_c) \quad 0\Big] + \hat{Q}_c({\hat{R}_c}^{-T}Z) \tag{2.4}$$

2.0.2 Related work with Entropy Minimisation Algorithm:

Petrie introduced the model of how to order a data matrix in 1899. Later this is named as data reordering or seriation. This methodology is used in many different application disciplines for example archaeology, anthropology etc. Data ordering has huge impact on some cases for instance gene expression data analysis in bio-informatics, geographical data analysis, bandwidth minimization or data compression [?].

The main assumption of seriation in data visualization is based on the assumption of permuting either rows or columns of the dataset without loss of information. Therefore, data reordering is done in such a way that similar examples or features are close to each other. Closeness of data helps to improve the quality of visualization without loss of any information as data dimension reduction methods do.

The seriation of the dataset can be formalised as if there is a dataset having n objects $O_1,\ldots, O_j$ one can construct an n*n symmetric dissimilarity matrix $D=(d_{i,j})$ where $d_{i,j}$ for $1 \leq i, j \leq n$ represents the dissimilarity between objects O_i and O_j, and $d_{i,i} = 0$ for all i. The major challenge in seriation problem is to find a permutation function.

Researchers proposed different permutation function for reordering data. Hahsler et al. [?] reviewed a larger number of permutation function such as column gradient measure, anti-robinson effects, Hamilton path length, inertia criterion, least squares criterion, linear seriation criterion, measure of effectiveness and stress measure.

To find the most loss/merit functions in discrete optimization problem is a complex problem. An exhaustive serach is infeasible because the number of possible permutations for n objects is n!. Researchers proposed different heuristics and seriation methods that are briefly covered in the following subsections.

Entropy Minimization

There are some loss functions used for data seriation which are related with entropy. For instance, in [?] author defined the stress measure as a sum of local entropy of each data item. To minimize the sum, Wilkinson proposed a heuristic approach [?] and Niermann proposed a genetic evolutionary algorithm [?]. The another way to encode the dataset using

Differential Predictive Coding (DPC). In this method, each item is encoded as a difference between current and previous items. Djuric et al. proposed an efficient algorithm to minimize the entropy of the encoded dataset in [?].

Travelling salesman problem solver

Data reordering using the length of a Hamiltonian path as a loss function is equal to solving a Travelling Salesman Problem (TSP), which is a well known and extensively researched combinatorial optimization problem. The aim of an Travelling Salesman Problem Solver (TSP-solver) is to find the shortest tour that, starting from a specific city, visits each city exactly once and then returns to the starting point. As the general seriation problem, solving the TSP is also complex. In case of seriation with n + 1 cities, n! tours have to be checked. In order to avoid exhaustive search, different heuristics were proposed, from simple nearest neighbour methods to complex approaches like the Lin-Kernighan (LK) algorithm [?]. Recently, Djuric and Vucetic [?] introduced a fast *O(nlog2n)* TSP-solver, called the TSP-means.

Hierarchical clustering

Hierarchical clustering and its extension named Hierarchical clustering with optimal leaf ordering(HC-olo) are most commonly used methods in bioinformatics [?]. The output of HC is a series of nested clustering which are stored in a tree. The order of leaf nodes in a tree is used to produce the linear order of the example. The problem is to find the optimal leaf ordering because a binary tree having n leafs and fixed tree structures have 2^{n-1} different linear orderings. To solve this issue, Bar-joseph et al. [?] proposed an optimization algorithm called HC-olo. The time complexity of $O(n^3)$ is not at all suitable for big data.

Low-dimensional projection algorithm

Researchers also target to use the state-on-art dimensionality reduction algorithm for example, PCA, LLE, LDA to project the original dataset into one-dimensional subspace. The linear ordering of the examples is that new subspace can be used for reordering. The problem is the intention of any dimensional reduction method is not the reordering rather it is a byproduct of manifold search so the quality of produced ordering is not at all satisfactory.

Sugiyama's algorithm

To draw a bipartile graph with as few edge crossing as possible is a well-known NP-complete problem [?]. Sugiyama's algorithm is based on average heuristics approach for drawing problem. The backbone of the algorithm is to order nodes according to the average of their adjacent nodes in the opposite node set. To apply Sugiyama's algorithm for data reordering first Makinen et.al proposed first to make data into a binary format from the original dataset. Then we consider rows and columns of the matrix as two separate nodes sets, where binary values represent edges between nodes. In figure 2.2 shows applying the average heuristic to a simple bipartite graph.

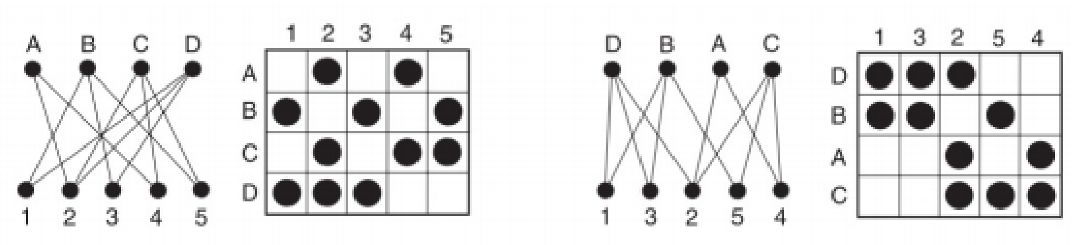


Figure 2.2: IFLDA/QR algorithm

Entropy-Minimization-based data reordering

In this section we will cover the algorithm that we will implement from data reordering representative. The algorithm will be presented shortly in this section. The approach that has been proposed in [**?**] we will implement in the algorithm and modify it to adopt for data streaming. According to Djuric et.al [**?**]:

- Ordering can be done by doing permutation of rows or columns that ensures maximally compressible data set. The maximally can be determined by the entropy of the residuals of predictive coding.

- The problem is determined by an Expectation-Maximization algorithm which alternatively solves a TSP and assign reweights features based on the quality of the resulting tour.

- The proposed TSP solver known as TSP-means find the path comparable to those by LK algorithm. By applying K-means (k=2) recursively the algorithm construct a Binary tree. The runtime of TSP solver is *O(nlog(n))*.

In next following paragraph we will describe the differential predictive coding, entropy minimisation reordering and TSP-means algorithm respectively.

Differential Predictive Coding: We assume our dataset D is stored in a form of n*m data table.

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2j} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{im} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nj} & \dots & x_{nm} \end{bmatrix}$$

where ith row vector represent an example with m features.
Differential predictive coding replaces each example with its difference from the previous one, $\varepsilon_i = x_i - x_{i-1}$, where ε_i is called DPC residual. As a result, the initial data table D is transformed into D$_{\mathrm{DPC}}$ without loss of information since the original dataset can be retreived from the encoding.

$$\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1m} \\ \varepsilon_{21} & \varepsilon_{22} & \dots & x_{2j} & \dots & \varepsilon_{2m} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \varepsilon_{i1} & \varepsilon_{i2} & \dots & \varepsilon_{ij} & \dots & \varepsilon_{im} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \varepsilon_{n1} & \varepsilon_{n2} & \dots & \varepsilon_{nj} & \dots & \varepsilon_{nm} \end{bmatrix}$$

Entropy Minimization Reordering: In data compression theory entropy is used as a measure of randomness of the dataset, where the larger value of entropy denotes large randomness and small compression ratio. Small entropy denoted by $H_{DPC(\varepsilon)}$ means DPC residuals are small which implies D is a well ordered dataset. $H_{DPC(\varepsilon)}$ can be estimated as

$$H_{DPC}(\varepsilon) = -\frac{1}{n-1}\sum_{i=2}^{n} logP_{\varepsilon}(x_i - x_i - 1) \tag{2.5}$$

where $p_{\varepsilon}(\varepsilon)$ is the probability density of vectors ε.
The permutation of examples π^* whose has minimize entropy of DPC residuals is the optimal one.

$$\pi^* = \underset{\pi}{\operatorname{argmin}}(H^{\pi}_{DPC}) \tag{2.6}$$

Djuric et al. [?] proposed a model P_{ε} as Gaussian or Laplacian distribution that results in introduction of $\sigma = [\sigma_1\sigma_2...\sigma m]$- vector of m parameters. According to his paper equation 2 can be restated as:

$$(\pi^*, \bar{\sigma}^*) = \underset{\pi^*,\bar{\sigma}^*}{\operatorname{argmin}}(H^{\pi}_{DPC}) \tag{2.7}$$

For data reordering the reason for using expectation-maximization-like algorithm is used to find a permutation of examples π which gives the minimize DPC residuals.

In M-step we need to find a permutation of examples π,which assumes σ_j are known and find the entropy. This way is equivalent of solving TSP where features are downscaled using σ_j and then TSP-means algorithm can be applied.

When the current ordering π is found, the goal of E-step is to calculate new values of $\bar{\sigma}$ which in return minimizes H^{π}_{DPC}. We can derive the new parameters using following formula:

$$\sigma_j^2 = \frac{1}{n-1}\sum_{i=2}^{n}(x_{\pi(i),j} - x_{pi(i-1),j})^2 \tag{2.8}$$

TSP-means: To start TSP-means we need to recursively apply k-means clustering (k=2) to initial dataset to create binary tree T which follows a conversation from binary tree to 2^l-ary tree T^l by keeping only nodes at every l^{th} tree level. After the conversation the algorithm traverse the tree in a breadth first way from left to right. The goal is to reorder internal and leaf nodes so that similar examples (leafs) and clusters (internal nodes) are closer to each other. To achieve this, we are going to use a TSP-solver for example: LK algorithm to reorder the children of the node together with their neighbours and replace the node by its reordered children.

Chapter 3

Solution

At present due to the increment of mobile devices huge number of streaming data are being generated. Unfortunately, these datasets are not in an organised as it should be for the analysis. There are many null values in the dataset and the data are being high-dimensional as well. To analyse this kind of data we need to implement fast pre-processing steps to analyse those data sets. One of the important preprocessing technique is dimensionality reduction.

An extensible framework to reduce the dimensions of streaming data with a concept of modularity of each component. To achieve this goal, framework is designed in such a way that each component act as a stand alone component but can be interacted with each other in a very convenient way. Moreover, the components are plug-in and play designed architecture. Each component can be easily replaced or added more functionality through out the whole implementation phase when it is necessary.

In a broadshell, there are three modules of the whole framework which are listed below:

- Data Set Load and Processing Module
- Dimensionality Reduction Module
- Visualization Module

Each module has its own responsibilty of the total framework. The task is divided in such a way so that the separation of concern principle exists. Each module is considered as a distinct section so that each section addresses a totally separate task. However, there are some interdependency of executing the order of component which will be addressed into the implementaiton part.

The architecture of the framework is the multitier architecture. From software engineering perspective in multitier architecture[1] client-server are involved where presentation, application processing and data management functions are physically separated.

1. Persistence Layer

[1] https://en.wikipedia.org/wiki/Multitier_architecture

2. Business Layer
3. Access Layer
4. Presentation Layer

According to the defintion from software perspective, operation with the all data

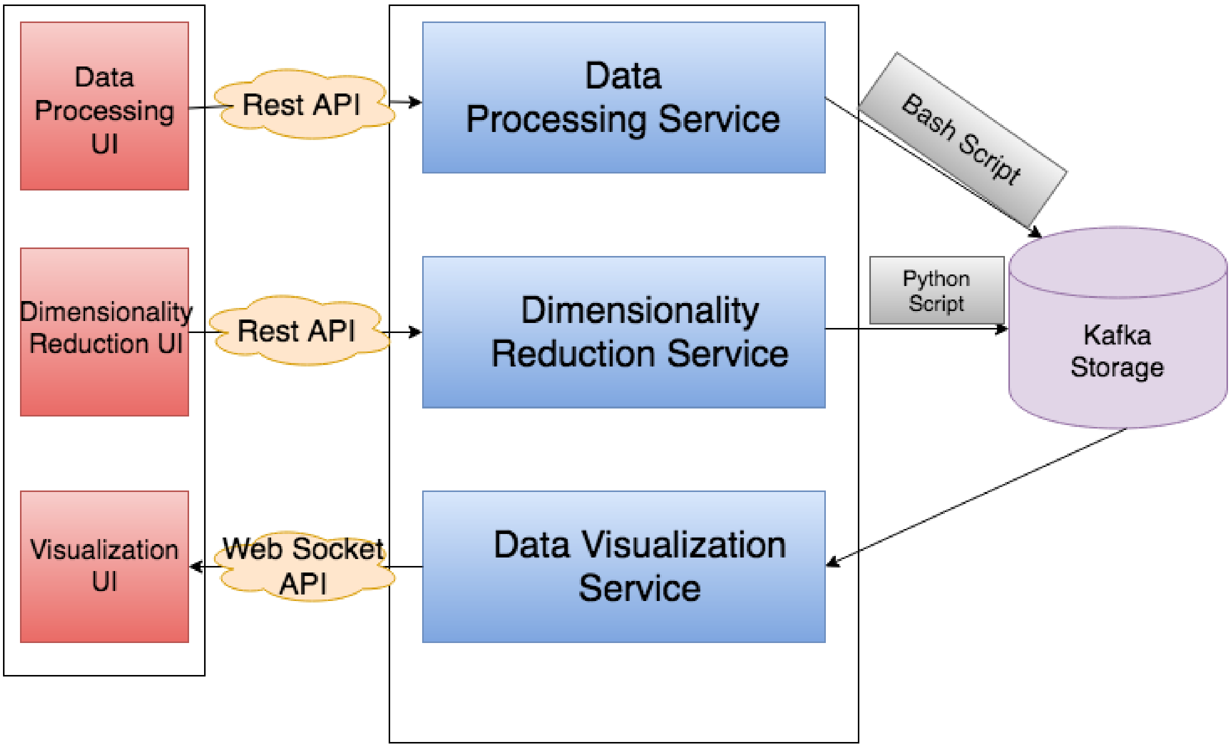


Figure 3.1: Architecture/ Design for the Extensible Framework

In this section, I will describe the solution approach of reducing the high dimensional streaming data to low dimensional streaming data.

The following figure 3.2 contains the general process model of our Framework.

3.0.1 Selection of Data Sets

The framework will be flexible for both the online available streaming API to consider as a data source and also have the ability to upload the static data. Hence we are considering the streaming data it will be good if the chosen data domain have some impacts on the change of the data with the passage of the time.

If the chosen data set is not the streaming data then the framework will be capable enough to convert the static data to the streaming data using any available free state-on-art tools.

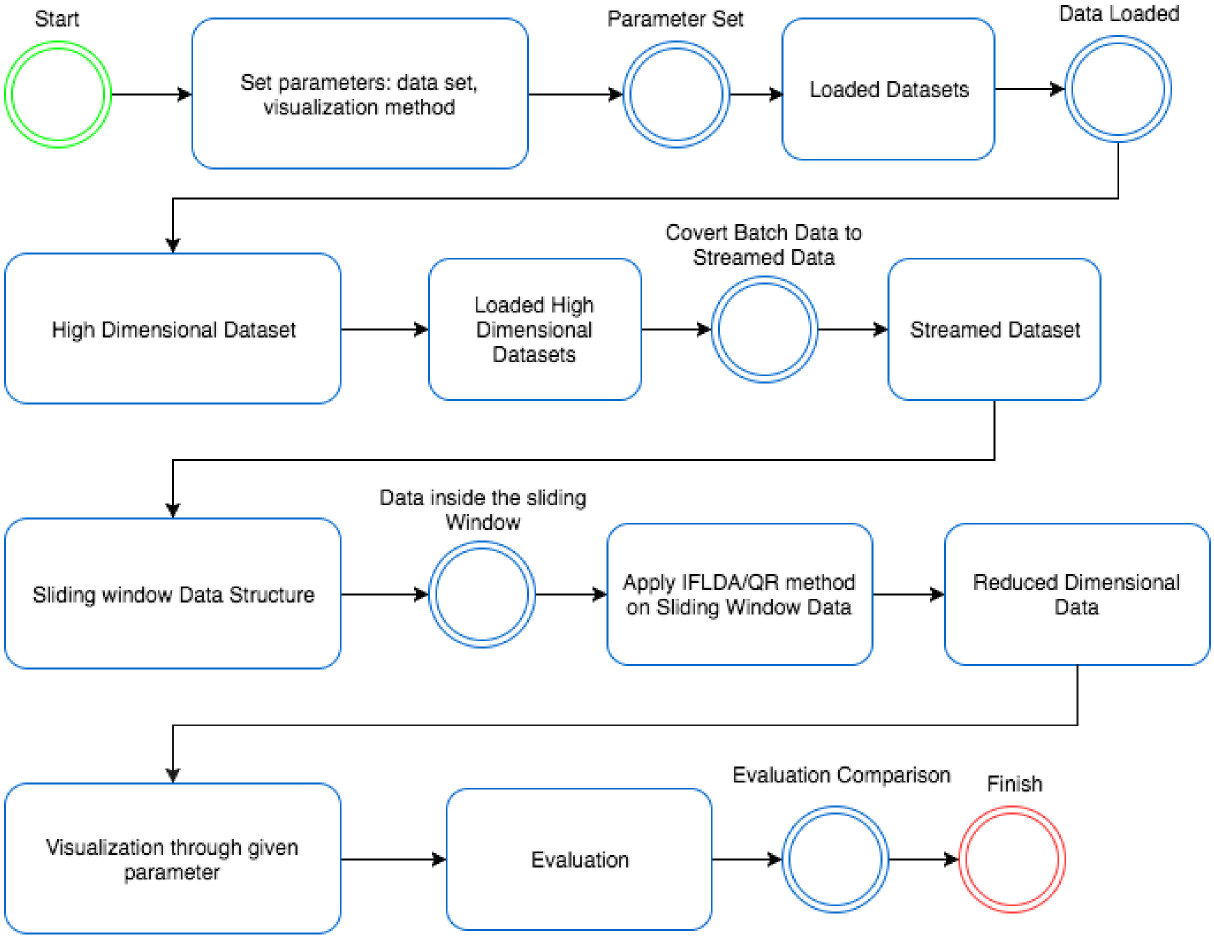

Start
Set parameters: data set, visualization method
Parameter Set
Loaded Datasets
Data Loaded
High Dimensional Dataset
Loaded High Dimensional Datasets
Covert Batch Data to Streamed Data
Streamed Dataset
Sliding window Data Structure
Data inside the sliding Window
Apply IFLDA/QR method on Sliding Window Data
Reduced Dimensional Data
Visualization through given parameter
Evaluation
Evaluation Comparison
Finish

Figure 3.2: Dimensionality Reduction Framework of Streamed Data

3.0.2 Define Data Model

The main characteristics of the streaming data that the data comes continuously with an unbounded structure and pattern as time progresses. For this reason, to handle the streaming data there should be some mechanism to hold the data for a minimum period of time and within that time all necessary work related with the data should be done then the data is thrown off. One of the common mechanism to handle those data is known as sliding window approach. Here the window is defined based on the time like in particular interval for example in each 500 ms the data comes are considered one window. The window will be moved after each fixed period of time and that's the reason it is known as sliding window approach. Our framework will be capable of considering the streaming data in a sliding window basis.

3.0.3 IFLDA/QR algorithm

As previously mentioned we will implement the IFLDA/QR algorithm. At the first window, we will implement the FLDA algorithm to calculate the centroid matrix and implement the Cholesky Decomposition of the centroid matrix. The output of this mechanism is the optimal transformation matrix.

From second window and so on we will implement the algorithm of the insertion of the chunk data. Here as an input of the process we will consider labeled new samples and also the novel class. Here the centroid matrix for an existing class will be updated as well as the new centroid of the new cluster will be calculated.In following figure 3.3 the whole process is shown step by step wise.

3.0.4 Data Visualization

The final outcome of the framework will be the visualization. In visualization, we will see the inner relationship of the data. In this case, visualization will not be only used for the communication to the end user but also it will reveal the inner meanings of the data for example, the pattern of the data, the relationship among the features and many more. The visulaisation tool we used here will be the Heatmap, one of the most used visualization tool for high dimensional data.

3.0.5 Design Framework

In this section, we will describe the design of the framework in details. The framework should have ability to extend for further implementation if needed. Moreover, the component should be reusable. The framework will also have the separation of concerns by allocating tasks to different layers.

The whole design Framework has four layers:

1. Persistence Layer
2. Business Layer
3. Access Layer
4. Presentation Layer

Each layer will be associated individual category of tasks. In Persistence layer the data will be saved and used for the tasks involved with data. The Business layer is responsible for doing all back end task for the framework. Through Access layer, Business layer and Presentation layer will communicate with each other where Presentation layer is responsible to present the data and take input from the user.

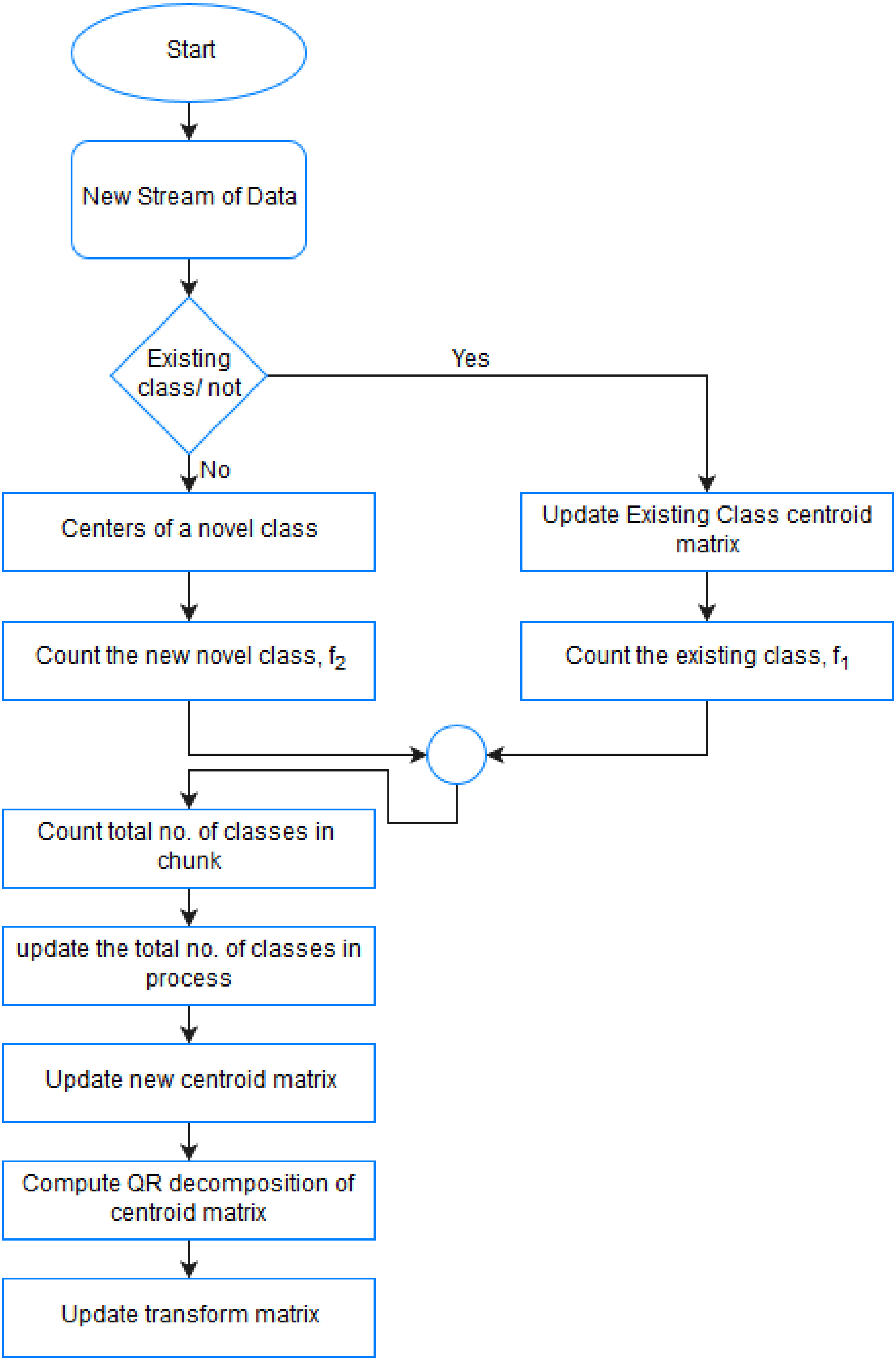

Start
New Stream of Data
Existing class/ not
Yes
No
Centers of a novel class
Update Existing Class centroid matrix
Count the new novel class, f_2
Count the existing class, f_1
Count total no. of classes in chunk
update the total no. of classes in process
Update new centroid matrix
Compute QR decomposition of centroid matrix
Update transform matrix

Figure 3.3: Process of Handling Stream Data

This whole section will be further divided into three subsections. We will describe the requirements list at the beginning then the overall design of the framework and last but not the least the sequence diagram of the framework.

List of Requirements

The whole requirements for the framework will be described in this section. The framework should be adaptable enough to response based on the user input. Till now, the framework should meet the following requirements.

1. User can select data type from stream data or batch Data
2. User can choose data set from available dataset
3. User can add new dataset to the specified place by uploading to the framework
4. User will get the reduced dimensional data for given input data
5. User can select the time interval for observing the visualization
6. User can select the required evaluation criteria from the evaluation list and show the evaluation information.

Architecture Diagram

In this section we will present the architecture diagram of the framework. As mentioned, the architecture is divided into four sections and each section has separate components for the different concerns. In fig 3.4, each section is defined with the individual components. Each component is responsible for fixed set of requirements. Each layer has a bi-directional communication where one layer is communicated with others. Based on the requirements a particular set of component is assigned for a fixed task. In Persistence layer, all information of the data is stored and ability to perform both read and write operation. There are in total five components in Business layer for performing all the requirements. The Access layer is all used for the communication using Application Programming Interface. There are five different ways for the user to communicate with the framework. In Presentation layer, the user will send the parameter data to the Business layer & receive data after the process done. In Business layer for implementing business logic it will communicate with the Persistence layer and send data to the front end. In addition, the business layer will also do the mathematical calculation for presenting evaluation information to the presentation layer.

In Presentation layer, user can give three inputs for showing the intended output: Select Data type, Dataset, Time interval selection, Select evaluation parameter. For adding new dataset user can upload to the framework. The details requirement list for the Presentaiton layer is listed below:

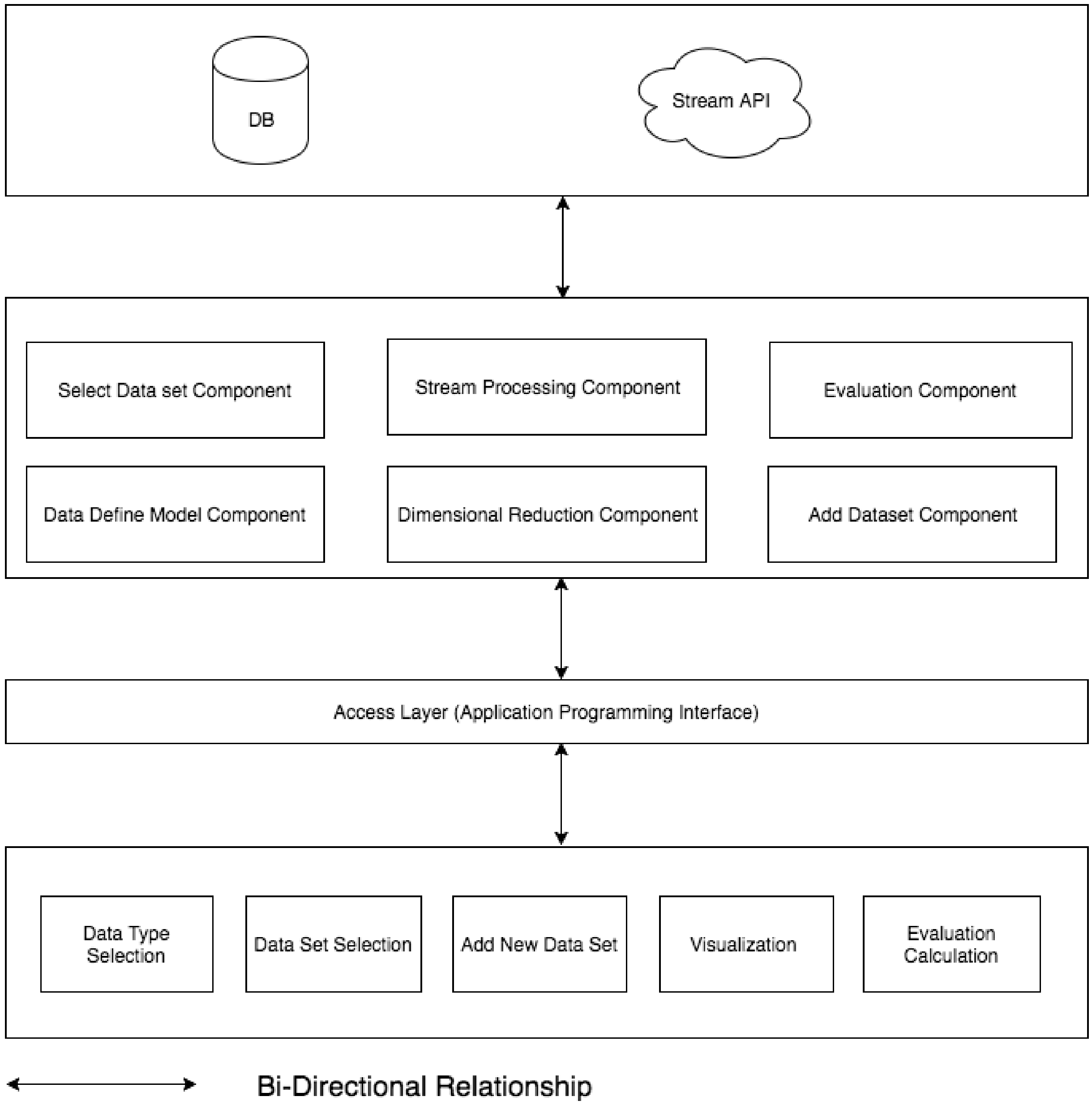

DB
Stream API
Select Data set Component
Stream Processing Component
Evaluation Component
Data Define Model Component
Dimensional Reduction Component
Add Dataset Component
Access Layer (Application Programming Interface)
Data Type Selection
Data Set Selection
Add New Data Set
Visualization
Evaluation Calculation
Bi-Directional Relationship

Figure 3.4: Framework Architecture

- Select Data type: User can select either Stream Data or Batch Data
- Select Dataset: User can choose data set from Available Dataset
- Add new Dataset: User can add new dataset to the specified place by uploading
- Show Visualization based on given time interval: Here, user will select at what interval of time user wants to visualize and be able to watch visualization
- Evaluation Information: User can show the evaluation information for example information loss will be shown in continuous fashion

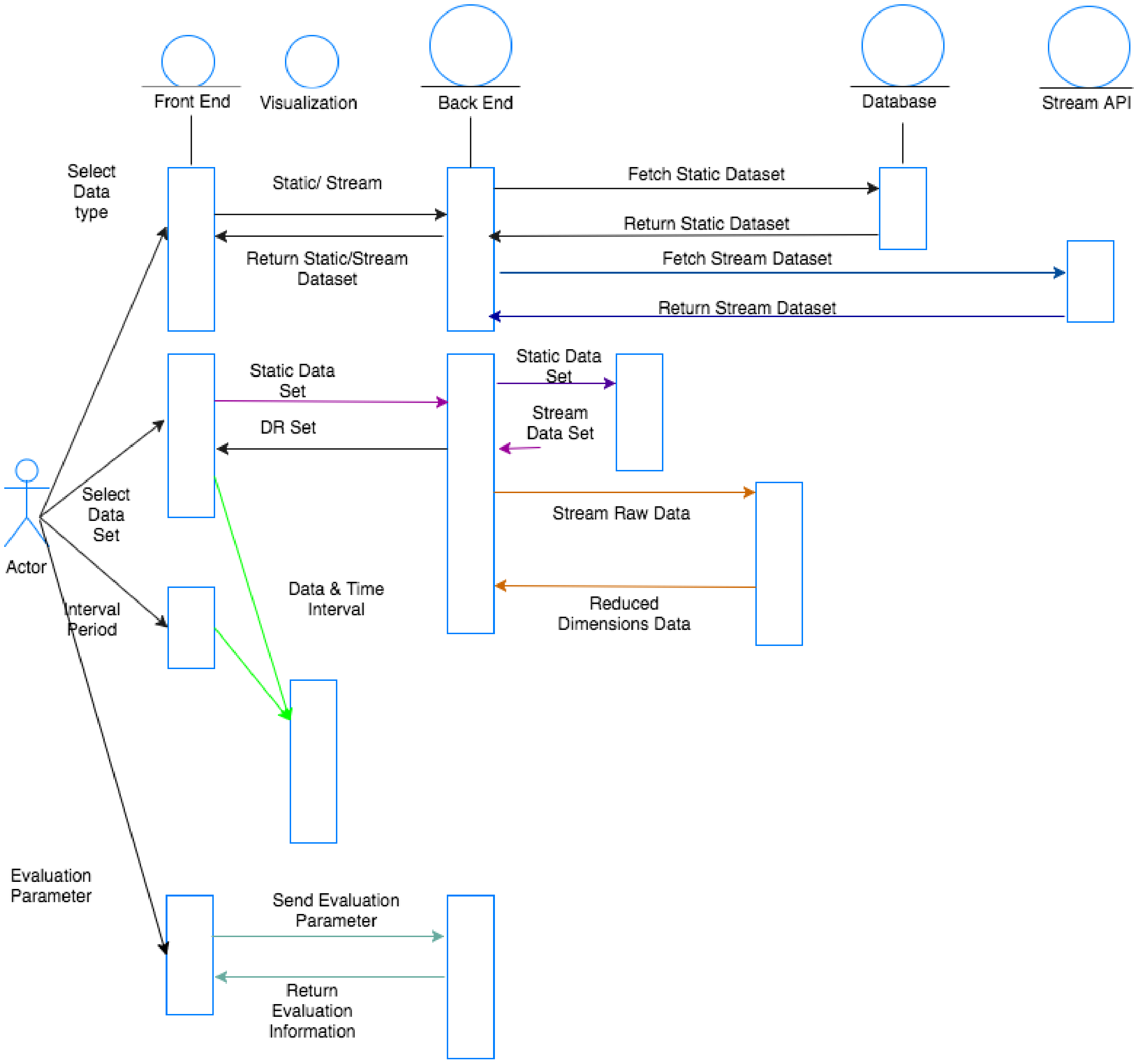
From Presentation layer the framework will access the Business layer through Access layer. The framework will have few Application Programming Interface associated with each task. In Business layer, the framework will implement the business logic of the framework. The details requirement list for the Business layer is listed below:

- Save Datasets to Database: Framework should save the dataset that user gives to the Presentation layer and give feedback in both accept state or rejection state. The dataset should be available from that point on wards in the framework if accepted.
- List of available Datasets: If user selects the Batch data then framework should send all the available datasets to the user for selection.
- Stream Processing: The framework should be able to detect whether selected type is batch or stream. If Selected type is stream than this component should not do any further tasks otherwise the framework will convert the static data to the stream data
- Data Define Model: After selection of the data the framework should capture the data for the fixed time window and apply the dimensionality reduction on that window.
- Dimensionality Reduction: Framework should be able to reduce the dimensions of the input dataset.

Sequence Diagram

In this section, the corresponding sequence of each requirement is described in details like how the request/ response is communicated from each layer through out the framework and also start or finishing time of any requirement. In fig 3.5 the whole process is shown in details. The total sequence diagram is explained in step wise one after another.

Selection of Data Type: User will select the data type and this value will be send to the backend. The value will be saved and send to "Select Data set" component . From backend, it will communicate to the database and receive the available data set. If the type is stream then it will directly communicate with the cloud and collect the available datasets from that stream. The dataset will be either a set of the static data or a set of streaming Application Program Interface.



DR = Dimensionality Reduction

Figure 3.5: Sequence Diagram

Show Visualization based on time interval: User will select the time to define the frame of each window. This input will be required to the Business layer to "Data Define Model" component. User will also select time interval to see the visualization after a certain interval. For example, if the given interval value is 5 then user will see the visualisation after 5 window respectively.

Selection of Data Set: User will select the data set from the available data set from the previous step . If the selected data set is "Batch" data set then from persistence layer it will go the "Stream Processing" component. The main task of this component is to convert static to stream data. If the type is "Stream" then this component has nothing to do other than sending void. This stream data will be used for further process.

Now the stream data along with the window frame size will go to the "Dimensional Reduction" component. This component is the heart of the framework. This component will be responsible to reduce the dimensions by implementing the algorithm and send back to Persistence layer. Based on the given interval value, the persistence layer will show the visualization in segment "Visualization" of the Persistence layer.

Evaluation Information: Here the list of all possible evaluation criteria will be given in "Evaluation" of the presentation layer. From the given value it will communicate with the "Evaluation" component of the persistence layer and show the value to the user.

Add new Dataset: User will also be able to add new data set or a cloud Stream API. From Presentation layer with the given data set it will go the Business layer. From there Business layer will communicate with the Persistence layer and save the value. The datset or API should be available from that time on if required.

Chapter 4

Implementation

- Describe in detail how the solution was implemented.
- Give a detailed system architecture and detailed descriptions of the individual components.
- Explain algorithms, data structures (e.g., database schemas) in more detail.
- The right level of detail is UML class diagrams. Do not list any code here, unless it is a really important part of an algorithm and there are no better means to illustrate the algorithm.
- Important: describe the *process* of getting to the final solution, do not describe only the final solution. All design decisions are important (I preferred X, because Y performs badly).

Chapter 5

Evaluation

We will evaluate the visualization of framework through different ways. The evaluation will help us to measure the quality of our framework. The rate of data arrival is very high in stream data. Therefore it's a big challenge for the framework to adopt itself with the flow of data. Secondly, the presence of data will be for a limited time and there is no chance of data availability for the second time so it's also a challenge for the framework to visualize the correct result just by using once. Last but not the least, data will be unpredictable and unbounded in structure therefore framework need to be stable from that perspective as well.

As previously mentioned, evaluation can be done in two different ways: Quantitative evaluation, Qualitative evaluation. Here, we will focus more on the Qualitative evaluation through comparative methodology. The first comparison will be regarding the studying of effect of data reduction through visualization. Our framework will be able to give the visualization without applying the data reduction framework and after applying data reduction. Hence, information loss is must while applying data reduction; we will calculate the Entropy matrix for calculating the amount of information loss.

Our second evaluation will be done regarding the comparison of performance between using incremental data and also the streaming data. Our framework will be flexible to use data in both ways. Through this, we can show the effect of interrupting continuous flow of data to the visualization. To calculate this we will use the Earth-Mover distance matrix for that.

Our framework should be scalable enough to adopt as the number of dimensions increase. The framework will be tested with the excel of dimensions.

For evaluation, the framework will be tested with the expert user from domain field. We will observe how the expert user is gaining the knowledge or communicated through out the visualization and how they understand the inner meaning of the visualization. The framework will also be tested by giving a new data set from the user and see the performance of the framework.

The following list contains the evaluation criteria of our framework.

1. Studying the effect of data reduction on the visualization quality.
2. Comparison of performance of the algorithm between using incremental data and streaming data
3. Scalability of the Framework with the increase of dimensions.
4. Calculate the evaluation matrix: Entropy calculation, Earth-Mover Distance.

Chapter 6

Conclusion

- Briefly summarize the contents of your thesis.
- Discuss the results: what are the important conclusions which can be drawn from your work?
- Give a brief outlook for future work. List only a few points (2-4), but avoid trivial issues such as “The user interface has to be improved” or “There needs to be more testing”.