

A deep fast learning framework towards exploring Imbalanced data and Multi-class Drift in Evolving Data Streams

K.Amrita Priya¹, Dr.R.Priya²

¹Research Scholar Department of Computer Science, Sree Narayana Guru College, Bharathiar University Coimbatore

²Associate Professor & Head Department of Computer Science, Sree Narayana Guru College, Bharathiar University Coimbatore

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 20 April 2021

Abstract : Data stream classification poses great challenges in the text based data mining community towards handling evolving data stream. Identification of feature evolution and imbalanced data on the class generated is an important research area for data stream classification on employing of traditional machine learning classifiers. Class evolution and drift is the phenomenon of class emergence and disappearance. Due to class evolutions, performance of the learning model degrades drastically over time. Class evolution problem has been handled on analysis of feature drift and multi class drift. Multi-class drift occur according to probability and time, is categorized as sudden, gradual and recurring drift. Multi class drift has been captured by proposing a new framework in this paper which is named as “Deep Fast Learning Framework”. Initially feature has been extracted using ensemble of technique such as Incremental Kernel Principle Component Analysis, Incremental linear Discriminant analysis and Incremental Linear Principle Component Analysis. These techniques for feature extraction have treated as online feature extraction process. Extracted feature has been processed in the deep fast learning classifier framework which is composed of hybrid ensemble classifiers which follows chunk based ensemble and online ensemble classifiers in parallel on basis of gradual class evolution on block of data on the data streams in form of features. Base learner or classifier has been established using deep neural network to generate the fast learning model on deep analysis of the features obtained and its relationship with existing classes on continuously updating the learner by replacing the older model with newly trained model. Further Base learner will remove the emerging classes which is least utilized and detect the recurring classes on basis of the feature obtained easily. This model is effective in determining the novel classes and recurring class to features which has the possibility of multi class drift. Finally class imbalance problem has been handled on employing under sampling method for base learning model. Experimental results has proved the superiority of the proposed framework on benchmark dataset against state of art approaches on the performance measures such as precision , recall and f measure.

Keywords: Data Stream Classification, Class Evolution, Ensemble Approach, Deep Learning, Class Imbalance data, Feature Extraction

1. Introduction

Classification is an important application in the field of data analysis. In data analysis, classification of streaming data has been widely analysed on various aspects. Generally, text data stream classification is considered as classification task that are employed on a of rapidly incoming data records to distributed servers [Baena-Garca et.al ,2006]. In a situation, text data classification becomes cumbersome as data extracted from distributed system may change dynamically and its text data distribution also change accordingly to data is considered as concept drift. Meanwhile it also termed as temporal class changes of data distributions. Over the last few decades, concept drift has been largely studied on employing the machine learning classifier through ensemble mechanism [Baena-Garcia.M, et.al, 2006]. The large number of the existing studies considered on the concept drift which has been occurred due to change in class-conditional probability distribution [Barddal.J.P et.al, 2015]. While class evolution on the incoming data streams is another important consideration which induces concept drift, it has been attracted relatively less attention.

Class evolution is concerned with certain kind of data change in the prior probability distribution of available classes and it corresponds to the evolving of a novel class and the disappearance of an outdated class on data distributions. For instance, twitter application propagates with new topics and outdated topics will be disappeared for short span of time. Besides, old topic such related to festivals becomes popular among the twitter users[Ditzler.G et.al ,2010]. This phenomenon has to be observed from data streaming applications since interest of user may change over time. Therefore Class evolution has considered as class-incremental learning problem. Class evolutions denotes the set of classes of feature instance occurred from data distributions whose prior probability is positive at various time stamp may change when class evolution occurs through data streams[Ditzler.G et.al,2013]. Hence the model needs to be adapted to classify the distributed emerging data

distribution to available existing classes in addition to determining the novel classes. At the same time, the distribution of data not happening to classes which has been considered as disappeared classes as to be removed from the model.

Class evolution can be effectively handled in this work on proposal of Deep Fast Learning Framework. It is termed as ensemble of technique to analyse the feature drift and multi class drift. Ensemble technique consists of feature extraction and classification on employing Incremental Kernel Principle Component Analysis, Incremental linear Discriminant analysis and Incremental Linear Principle Component Analysis for online feature extraction. Towards classification of extracted features, hybrid ensemble classifiers which follow chunk based ensemble and online ensemble classifiers has been employed in parallel. Base learner or classifier has been established using deep neural network to generate the fast learning model on deep analysis of the features and under sampling method has been used to eliminate the data imbalance issues. Finally learner effectively replaces the older model with newly trained model to detect novel class and recurring classes for feature with multiclass drift.

The rest of the article is sectionized as follows, Section 2 describes the related work on the classification of evolving data streams with concept drift and Section 3 represents the proposed approach to detect the novel classes and recurring classes with multiclass drift on evolving data streams. Empirical results on proposed and existing approaches have been reported with performance a measure on benchmark dataset in section 4. Finally section 5 concludes the paper.

2. Related work

In this section, brief review of technique dealing with novel class detection and recurring class detection in the multiclass drift on terms of concept drift and feature drift has been examined with examples and impacts.

2.1. Adaptive Classification of feature-evolving data streams

In this model, four major challenges have been addressed on evolving data streams such as high length, feature-evolution, concept-drift and concept-evolution using ensemble technique [Domingos.P,2000]. Concept-drift is a familiar phenomenon in evolving distributed data streams which happens as a result on changes in the underlying concepts of the evolving data streams. Concept evolution is also considered as another phenomenon in data streams which occur as an result of new arriving classes for feature instances in evolving in the stream. Finally Feature-evolution is a frequently occurring phenomenon process in many data streams especially in text form with occurrence of new features (i.e., words or phrases) appears as the stream progresses. Ensemble classification approach has been employed on those phenomena analysed to detect novel class, as it is more adaptive to the emerging data stream to enable it to identify more novel class at a particular time stamp.

2.2. Classification of Evolving Data Streams with Markov Boundary Learning

In this model, markov boundary has been projected towards classifying the evolving data streams without time delay on label prediction[Gaber.M.M et.al , 2005]. Traditional process of the label prediction leads to high error rate to detect changes in data distributions. Markov boundary on unsupervised classification produces class labels with dynamic learning structure which substantially different. These structures generate the micro cluster on distance based strategies by updating the position. Drift detection methods explicitly computes the concept drift and update the classifier model accordingly to the sliding window strategies. The model selection method for classifier and class based structure are designed for the important characteristics of class evolution for available data streams.

3. Proposed Model

In this section, a new framework named as deep fast learning framework which is to detect the adaptation of the class evolution using deep learning technique. The approach details each component of the framework to detect the novel class and recurring class of evolving data streams with multiclass drift

3.1. Definitions

Class evolutions on evolving data streams denote the set of classes which is composed of the following form of classes will be defined as follows

- **Data streams**

Data stream is a conditionally unbounded and ordered sequence of data instances that occur over various time. Let $\{(x_1,y_1), (x_2,y_2), \dots, (x_t,y_t), \dots\}$ denote the data stream. In this x_t and y_t are data instance received at t time stamps which is further mapped to class based corresponding class labels [Gama.J et.al, 2014].

- **Class Emergence or Novel Class**

It is a class which represents unknown class at the current time. Class generated at current time is represented as C which is

$$C \in C_1 \cup C_2 \cup C_3 \dots C_{t-1} \text{ Where } C \in C_t$$

- **Class disappearance**

It is a class which represents existing class which would not be represented in the next time stamps or due to evolving of data streams. Class disappeared at next time stamp is represented as follows

$$c \in C_{t-1}$$

- **Class Reappearance**

It is a class which represents reoccurrence class; it represents the class which disappeared as possibility of reoccurrence after some point of time or due to instance of streaming data. Class C is a recurring class is represented as

$$C \in C_1 \cup \dots \cup C_{d-1}, \text{ Where } C \in C_t$$

3.2. Ensemble Feature Extraction

Ensemble Feature extraction is that combining the outputs of several single feature extraction models will obtain better results than using a single feature extraction approach. Ensemble Feature extraction is carried out using wrappers filters and embedded models. Efficiency on ensemble approach is to obtain the feature subset with diversity [Japkowicz.N,2001]. Feature extraction in related to knowledge discovery, data dimensionality reduction and generalization concepts. Thus, several feature extraction processes are carried out either using different training sets, and their results are aggregated to obtain a final subset of features. The idea is that a more appropriate (stable) feature subset is obtained by combining the multiple feature subsets of the ensemble, as the aggregated result tends to obtain more accurate and stable results, reducing the risk of choosing an unstable subset. Embedded Method of feature extraction will balance complexity, diversity and stability of the process

3.2.1. Incremental Linear Principle Component Analysis

Incremental linear Principle Component analysis is a process in which some part of data streams is transformed into a feature space. This method is processed in such way that the data set obtained is represented by a reduced number of effective features and yet retain most of the intrinsic information contained in the data [Minku.L. et.al 2014]. It maximizes the rate of decrease of variance. Feature Transformed is represented in Vector form in the assumption that it has zero mean . The assumption of the data with mean is given as

$$E[x]=0$$

Where E is the statistical expectation operator.

Let q denote a unit vector, also of dimension m , onto which the vector X is to be projected with data instances. This projection is defined by the inner product of the vectors X and q , the transformation from projection is represented as :

$$A=X^Tq=q^TX$$

$$\text{constraint: } \|q\|=(q^Tq)^{1/2}=1$$

The projection A is a random variable with a mean and variance related to the statistics of vector X with feature instance. Assuming that X has zero mean which can be calculated as the mean value of the projection A is

$$E[A]=q^TE[X]=0$$

The variance of projection A is therefore same as its mean-square value for the vector on the variance of its feature instance. Projection A generates the principle component of feature instance on collected feature instance. It is given by

$$\sigma^2=E[A^2]=E[(q^TX)(X^Tq)]$$

$$\sigma^2 =q^TE[XX^T]q=q^TR q$$

Eigenvector and eigen value determines the principle vector and principle feature instance on the computation of covariance measures on the projected instances. With m eigenvectors, the projections of \mathbf{x} into the eigenvectors are given by:

$$\alpha_j = \mathbf{q}_j^T \mathbf{x} = \mathbf{x}^T \mathbf{q}_j, j=1,2,\dots,m$$

Vector represented the resultant feature space of the IPCA as numbers α_j . The numbers are called the principal components. In particular, particular technique may reduce the number of features needed for effective data representation by discarding those linear combinations in the previous formula that have small variances and retain only these terms that have large variances.

3.2.2. Incremental Linear discriminant analysis

Incremental linear discriminant analysis is to obtain linear combination of feature on the remaining part of evolving data streams to produce the maximum separation between the obtained instances. It is also computed using Eigen vector and scatter matrix[Minku.L, et.al, 2010]. The extraction of the feature will be obtained using linear mapping of the projection data to generate the feature instance on the data streams into feature set.

Given a set of data points of p variables $\{x_1, x_2, \dots, x_n\}$

The optimum mapping on objective criterion is given as $y=f(x)$

Vector is given by

The low-dimensional vector is represented as

$$\mathbf{x} = \sum x_i v_i = x_1 v_1 + x_2 v_2 + \dots + x_N v_N$$

The above representation of the vector containing data set generates the feature vector on particular objective function to produce the maximum discriminatory information on the obtained feature space on feature categories. The discriminatory information obtained on objective function undergoes analysis on scatter feature space which is given as follows

The scattered feature space produces the resultant vector containing the Eigen vector on eliminating the redundancy of the information in the data space. Finally obtained Eigen vector will processed towards classification

$$S_w = \sum_{i=1}^C \sum_{j=1}^{M_i} (x_{ij} - \mu_i)(x_{ij} - \mu_i)^T$$

3.2.3. Component Analysis

Incremental kernel Principle Component analysis is employed on final part of the dataset from the data streams in the particular time stamp. These dataset is processed to obtain the feature space on employing kernel function[Ramamurthy.S & Bhatnagar.R, 2007]. The kernel function takes the non linear input spaces and transforms it to kernel matrix. Kernel Matrix is given as

$$K = \{K(x_i, x_j)\},$$

$$\text{Where } K(x_i, x_j) = \varphi^T(x_i) \varphi(x_j)$$

Normalization of the kernel matrix is given by $Ka = \lambda a$

Where λ is an eigenvalue of the kernel matrix K and a is the associated eigenvector

Non Zero Eigen Vector for normalized kernel matrix is obtained as follows

$$a_k^T a_k = 1 / \lambda_k$$

Where λ_p is the smallest nonzero eigenvalue of the matrix K on generate the feature space.

Further extraction of the principal components containing a eigen values on the eigen vector to generate feature space using kernel based principle component analysis is obtained using complete projection as

$$a_k = \tilde{q}_k^T \vec{\phi}(\vec{x}) = \sum_{j=1}^N a_{k,j} K(\vec{x}_j, \vec{x}), \quad k = 1,2,\dots,p$$

Finally the feature space obtained on basis of the kernel computation has provided with the principle component as feature subset on the considered data set on the data streams collected at respective time intervals.

3.2.4. Aggregation method – Sliding window method

Sliding window method is employed as aggregation method for the feature space or feature subset obtained using the ensemble of feature extraction methods such as incremental linear principle component analysis, Incremental linear discriminant analysis and incremental kernel principle component analysis. Figure 1 represents the architecture of ensemble based feature extraction model for streaming data.

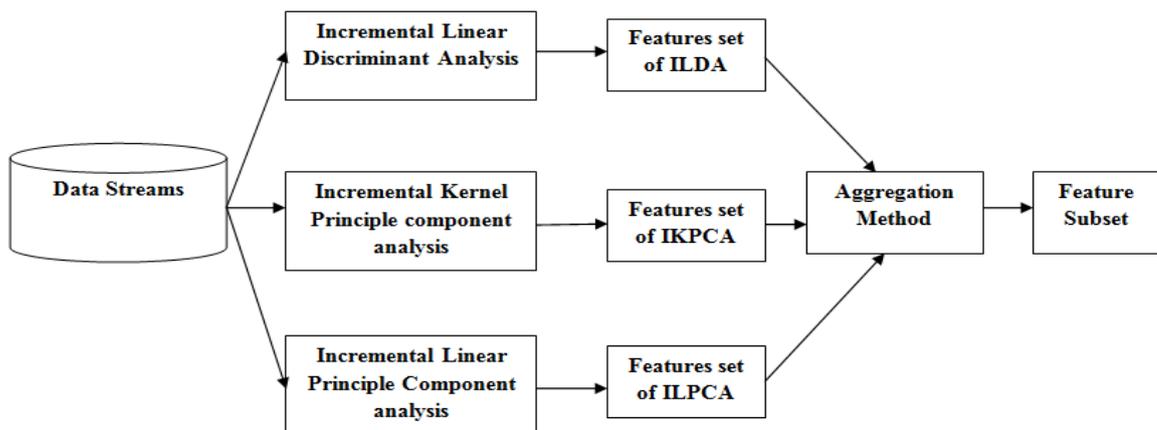


Figure 1: Ensemble Based Feature Extraction Technique

Sliding window method discards the old instance and constructs with optimal feature subsets. It is employed either by cutting-off oldest instances or weighting them dynamically according to their relevance. However, the size of the window has a crucial impact on its performance. A small window will be able to adjust to small and rapid changes, but may lose the general context of the analyzed problem and be prone to overfitting. A large window can efficiently store more information, but may instances originating from different concepts will be analysed against feature drift and concept drift.

- **Analysis of Feature Drift**

Feature drift is detection of changes in the distribution of feature on different time instances. Analysis of feature drift is carried out different time instance for the discriminative feature which is represented as $F_i \in F$. It is considered as feature set on the particular time point. On any two points i and j , $f_i \neq f_j$ it is considered as feature drift

$$F_i = \operatorname{argmax}_{f_i \in F} f^D(f_i, t)$$

- **Analysis of Concept Drift**

Concept drift may be defined as detection of changes in distributions of data streams. Presence of drift can affect the underlying properties of classes that the learning system aims to discover, thus reducing the relevance of used classifier as the change progresses. Drift in the concept of the feature on different time period is as follows

$$C_i = \operatorname{argmax}_{C_i \in C} C^D(f_i, t)$$

The change in data distribution of the feature set occurs whenever a subset of features becomes on the feature subset generation.

3.3. Deep learning based Ensemble Classifier

In this part, deep learning based ensemble classifiers has been utilized to detect the novel class and recurrent class to the multiclass drift occurrence on the feature subset on data distribution of the stream. It is composed of hybrid ensemble classifiers which are a set of algorithm to obtain the better classification results[Wankhade.K, Rane.D & Thool.R, 2017]. The hybrid ensemble classifier are employed to learn extracted features set during training period and then combined together to classify the unknown data. The single classifier tends to cause the bias in terms of a fixed set of parameters, and reduction of such bias can be obtained through the ensemble learning. The performance of ensemble learning depends on the precision of the constituent classifiers, which usually has stronger generalization ability than those base classifiers.

3.3.1. Chuck based Ensemble Classifier

Due to their compound structure they can easily accommodate changes in the stream, offering gains in both flexibility and predictive power. A new classifier is being trained on recently arrived data (usually collected in a form of chunk) and added to the ensemble.

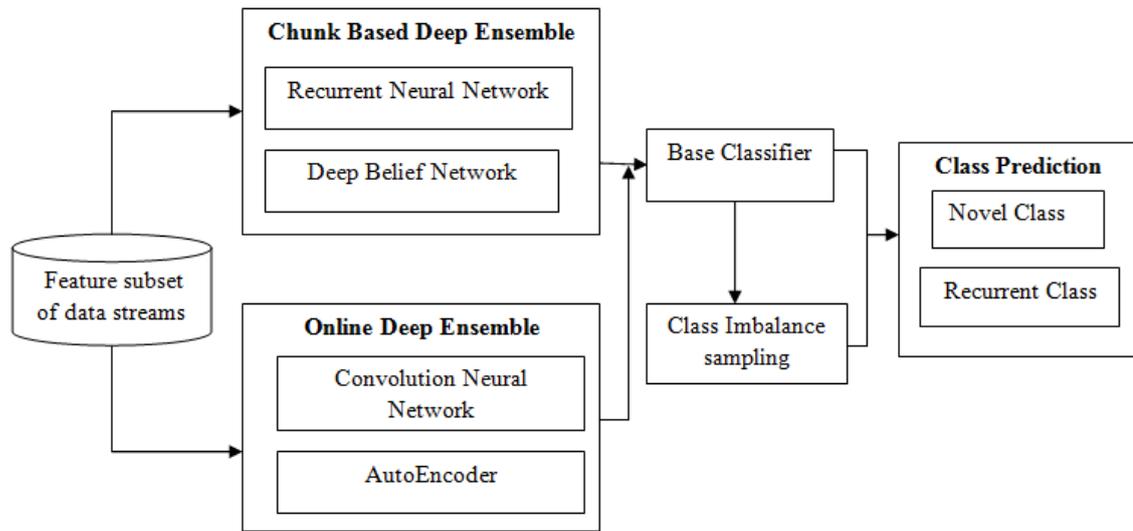


Figure 2: Deep Ensemble Classification of proposed framework

Figure 2 represents the architecture of the proposed framework for data classification of the feature vector obtained by ensemble feature extraction model. Pruning is used to control the size of the committee and remove irrelevant or oldest models. A weighting scheme allows assigning highest importance to newest ensemble components, although more sophisticated solutions allow increasing weights of classifiers that are recently best-performing. In this Deep neural network has been used in the work. Especially Recurrent Neural Network has been employed for classification.

- **Recurrent Neural Network**

Feature instances with temporal or sequential structures and varying length of inputs and outputs has been processed with different layers of RNN. It allows a memory of the previous inputs to persist in the model’s internal state and influence the outcome on basis of non linear dynamics. It works on reusing of weights .Figure 3 represents the ensemble layer of recurrent neural network towards classification of the instance of data streams.

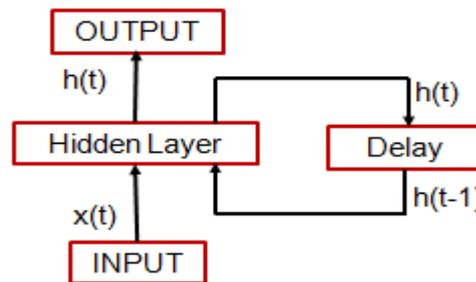


Figure 3: Ensemble Classification layers of Recurrent Neural Network

The ensemble classification on basis of hidden layer has been characterized to classification of the feature set to determine the novel class and recurrent class on generation of hyper parameter. Hyper parameter is modelled on basis of growing and pruning condition. Hidden layer generates the novel class and recurrent class for feature subset on sigmoid activation function as follows

Novel class is given as $\delta_k \equiv (t_k - a_k) a_k (1 - a_k)$
 Recurrent class is given as $\delta_j \equiv \sum_k (\delta_k w_{kj}) a_j (1 - a_j)$

In order to maintain sequential learning, this drift issue needs to be resolved without sacrificing the already acquired knowledge over the observed data distribution.

3.3.2. Online Ensemble Classifier

Classifiers are updated instance by instance, thus accommodating changes in stream as soon as they occur. Each object must be processed only once in the course of training, computational complexity of handling each instance must be as small as possible, and its accuracy should not be lower than that of a classifier trained on batch data collected up to the given time. Figure 4 represents the architecture of the CNN for classification of feature vector.

- **Convolution Neural Network**

Convolution Neural Network has been employed to classify the feature set on basis of Convolution layer which uses input layer and max pooling layer, activation layer on ReLu and Class Prediction layer. Convolution layer contain the feature set, further feature set are represented as max pooling of the feature. This feature in form of feature vector is processed using ReLu activation. ReLu activation layer balances the data to produce the classes on feature maps. Figure 4 represents the data stream classification on basis of convolution neural network.

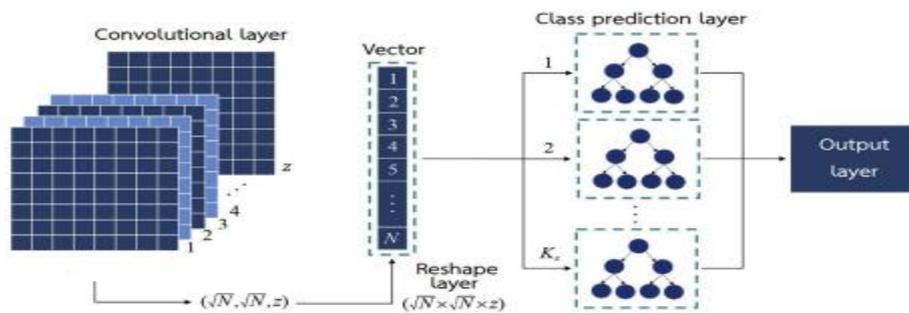


Figure 4: Ensemble classification based on Convolution Neural Network

Class prediction layer predict the novel class, recurrent classes and disappear class to disappear in the processing on basis of hyper parameter of the activation layer. Output layer of the convolution neural network consist of novel class and recurrent classes. Convolution Neural Network is more adaptable to different types of drifts.

3.4. Deep Neural Network- Base Classifier

Base classifier on ensemble learning using deep neural network will be constructed using weighted aggregation of majority. In this base learner maintains the weight vector which combines linearly with local class prediction vector composed of the feature instance using RNN on chunk based ensemble approach and CNN on online based ensemble approach. Weighted majority scheme determines the base learner to particular type of instance classification[Yan.Y et.al 2014]. The Weighted majority scheme on learner it determined using particular sampling rate

$$\text{Sampling rate} = 2/N\epsilon \log 1/(s\delta)$$

Multiclass drift on sampling rate estimation provides the better results. Class conditional probability is also happen to change in a real-world emerging data streams with the previously built class model for a novel class or recurrent classes which could become invalid later. Convolution Neural Network model predicts the class accurately on all time instances. It is computed using weighted adaptive bound of classifier towards sequence classification

$$\epsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}$$

Where r is random variable, R is the range of r, n is independent observations and Mean of r is at least $r_{avg} - \epsilon$, with probability $1 - \delta$. The model is capable of predicting the class in parallel on time changing data streams. Convolution Neural Network model can be employed as base classifier for data stream classification. Class imbalance will be occurring due to employment of soft max constraint.

3.5. Class Imbalance Estimation using Sampling method

Class imbalance problem can be resolved using sampling method. Class Imbalance is occurs due to as imbalance of the positive and negative instances. In this special case, the prior data distribution may change over various time stamps, generates dynamic class-imbalanced problem. To address this problem, an under-sampling strategy has to be embedded in base learner of ensemble deep learning. The sampling probabilities for the positive instance on the class and negative classes on the class are different. Furthermore, the size of each novel or recurrent class dynamically changes due to the gradual class evolution and emerging conditions.

Sampling aims to select the negative instance of data which is further to be classified in the positive instance of the particular class. In this w_t^i as the prior probability of class c_i at time t , the probability of sampling the negative feature instance for c_i is calculated as

$$p_i = \min(w_t^i / (1 - w_t^i), 1)$$

In order to quickly and accurately estimate updates the current base classifier by stochastic gradient descent with model. Updated model can able to predict the probability of the data classification and learn the emerging data stream with linear time complexity.

Algorithm 1: Deep Fast Ensemble learning

Input: Data Instance in streams or chunks Data instance (x_t, y_t) at time t , Class Set C_t , Feature subset F_t

Output: Class formation – New or Existing class

Process

// Novel Class Constraints

On Feature Vector F_t ,

If $(F_t = f_{t-1})$

$$C_t = \operatorname{argmax}_{C_i \in C^D(f_t, t)}$$

Then $C_t \leftarrow C_{t-1}$

Els $y = U^T x$

$$\text{Generate New class } C_n \delta_k \equiv (t_k - a_k) a_k (1 - a_k)$$

// Recurrent Class Constraints

On Feature Vector F_t ,

If $(F_1 \cup F_2 \cup F_3 \dots F_{t-1})$

Then Compute Class Label on all time instances on Max polling layer

Update Class Label using sigmoid function or ReLu activation function

$$\delta_j \equiv \sum_k (\delta_k w_{kj}) a_j (1 - a_j)$$

Assign the Class Label C_E to the feature instance F_t

4. Experimental Results

In section, the experimental results of the deep fast learning framework against the existing approaches towards data stream classification has been illustrated using k fold cross validation. In this analysis, proposed mechanism outperforms the existing approaches in terms of reliability as Pairwise completeness and reduction ratio as scalability, Precision, Recall, F1 score and Computation Time.

4.1. Dataset Description

The extensive experiments towards data stream classification using deep ensemble learning architecture using on two familiar real datasets composed of data stream with heterogeneous data integration to identify the novel class and recurrent class is as follows

4.1.1. RKB Person dataset

Forest cover dataset considered as data streaming data contains 581,000 instances, 7 classes, and 54 numeric attributes which is extracted from the UCI repository towards ensemble classification using deep learning. It

randomly generates 10 different sequences of each dataset for novel class and report the average result on basis feature extraction models.

4.1.2. SWAT Dataset

Dataset is a collection composed of large number of email messages collected from Spam Assassin collection as data stream. On analysis of each mail, it is represented as 500 attributes on feature extraction using the bag of words approach. Dataset are arranged to contain novel and recurring classes according the feature subset generated.

4.2. Evaluation

The proposed deep fast learning Framework for data stream classification is evaluated against the following measures Pairwise Completeness (PC), Precision (P); Recall(R), F1-scores (F) and computation time.

4.2.1. Precision

It is a measure used to measure the Positive predictive value on the class is the ratio of relevant feature instances among the retrieved feature instances from the feature vector on resultant feature extraction process. Precision is the number of exactly retrieved feature instance divided by the number of all returned feature instance in the feature space through particular technique [Amrita priya.K & Priya. R(2020),].

$$\text{Precision} = \frac{\text{True positive}}{\text{True positive} + \text{False Positive}}$$

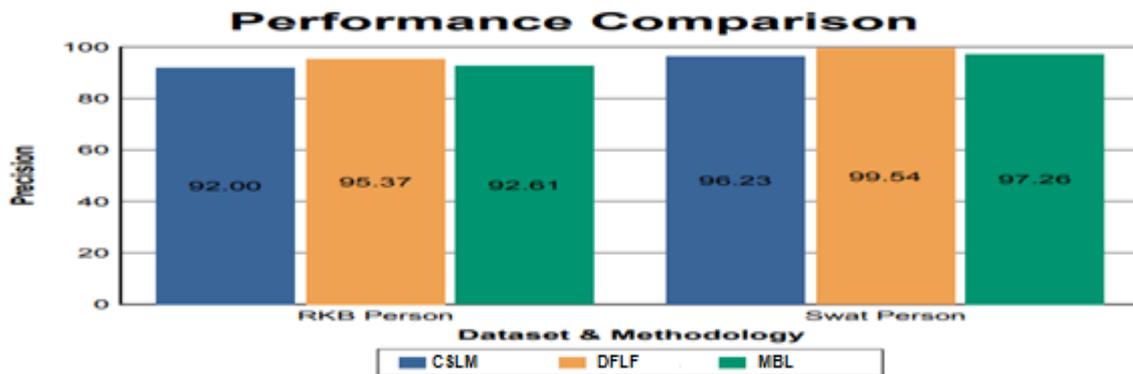


Figure 5: Performance Evaluation of the Methodologies on Precision against the different datasets

It is considered as True positive as it is a ratio of number of real positive feature instance in the feature vector and false negative is number of real negative feature instances in the feature vector. The precision is evaluated against different dataset is depicted in the figure 5

4.2.2. Recall

It is a measure of the fraction of relevant instances that have been retrieved from the feature vector over the total amount of relevant instances on the classes composing feature instances. The recall is termed as part of the relevant feature instance in the vector that are successfully classified into the exact classes as novel class or recurrent classes

$$\text{Recall} = \frac{\text{True positive}}{\text{True positive} + \text{False negative}}$$

It is a considered as True positive, as number of real positive instance in the feature vector categorized as class and false negative is number of real negative instance in the feature vector categorized as class. The recall measured is evaluated against different dataset is been represented in the figure 6

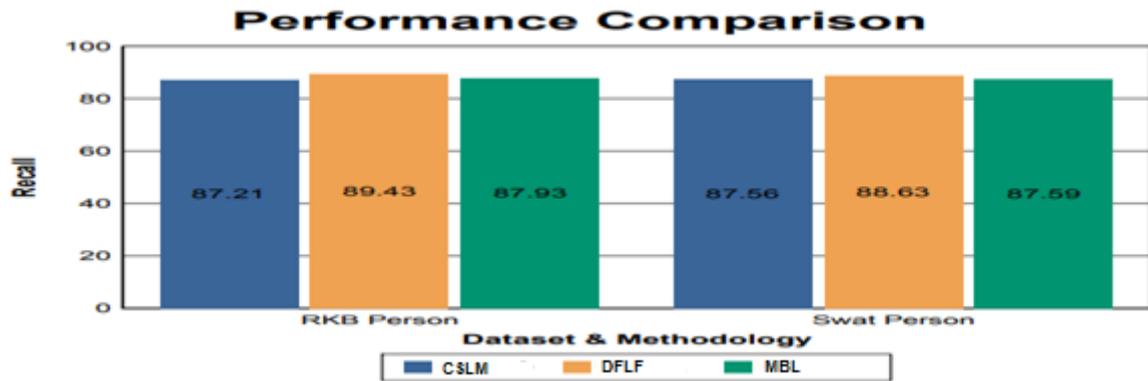


Figure 6: Performance Evaluation of the Methodologies on Recall against the different datasets

4.2.3. F Measure

It is a measure of a classification accuracy of the data streams and is defined as the weighted harmonic mean of the precision and recall of the data instance in the feature vector. The performance chart of the classification accuracy is described in figure 7 and its performance comparison value indicated in Table 1.

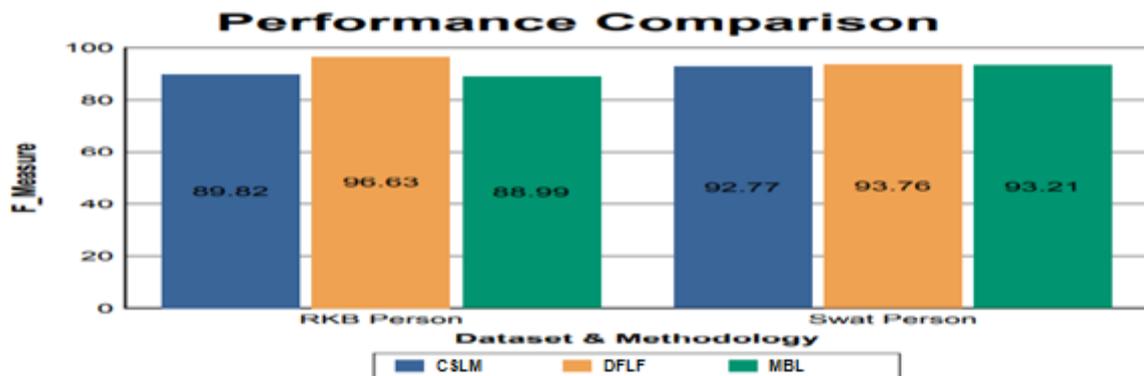


Figure 5: Performance Evaluation of the Methodologies on F Measure against the different datasets

4.2.4. Computation Time

It is defined as no of time taken to establish the classes towards detection of novel class and extracting recurrent classes which is disappeared in particular time for the evolving data streams on classifying the feature instance in the feature vector using deep learning architecture on the the different learning classifier between the two heterogeneous sources. The performance evaluation chart towards deep ensemble classification on data streams extracted from the various real dataset and its values is described in figure 7 and Table 1

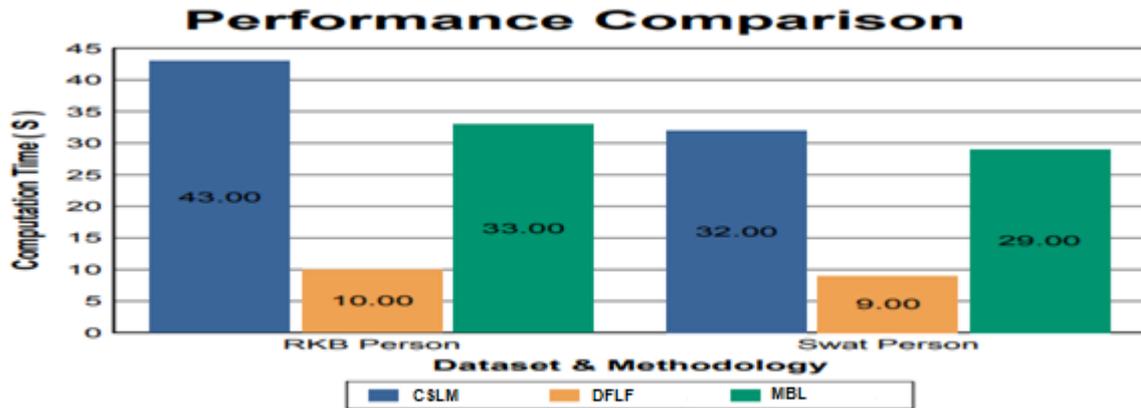


Figure 6: Performance Evaluation of the Methodologies on Computation Time against the different datasets

4.2.5. Data Reliability

It is the Measure of no of true positives is retained by a proposed deep ensemble classifier and it is the measure of the degree to which it reduces the number of pair wise comparisons needed respectively against data classified. The performance chart towards achieving the data classification is described in the figure 7 and its performance values of reliability on data classification are described in the Table 1.

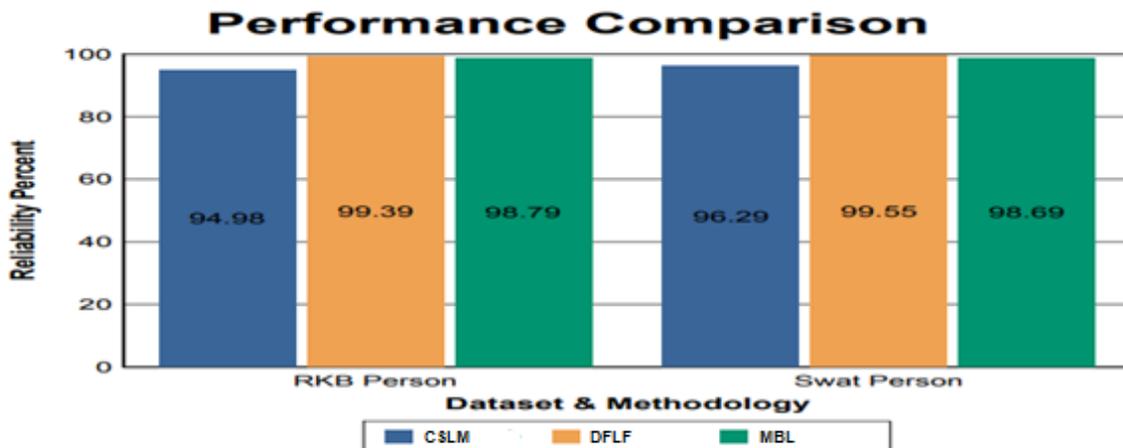


Figure 7: Performance Evaluation of the Methodologies on Reliability against the different datasets

As shown in Table 1, deep fast learning techniques enable the entire process to run 2-3 orders of magnitude faster than concept specific learning model and markov boundary learning model towards data classification on data streams.

Table 1: Performance Comparison of Methodology against measures for various dataset

Dataset	System	Precision in %	Recall in %	F measure in %	Computation Time (s)	Pairwise Completeness as reliability in %
RKB person	DFLF	95.37	89.	96.63	10	99.39
	CSL	92.00	43	89.82	43	94.98
	M	92.61	87.	88.99	33	98.79
	MBL		21			

			87. 93			
Swat Person	DFLF	99.54	88.	93.76	9	99.55
	CSL	96.23	63	92.77	32	96.29
	M	97.26	87.	93.21	29	98.69
	MBL		56			
				87. 59		

In order to examine the performance of the different ensemble learning of proposed algorithms greatly increased the runtime for both feature extraction on the data streams and the entire classification on terms of base classifier for classification of the feature vectors. Finally it is proved that it achieves better results in terms of the data reliability.

Conclusion

Deep fast learning framework for Deep ensemble classification on data streams has been designed and implemented in this work against various drift in the feature and classes. Initially ensemble feature extraction technique has been modelled using Incremental Kernel Principle Component Analysis, Incremental linear Discriminant analysis and Incremental Linear Principle Component Analysis to generate the feature vector composed of feature subset for deep ensemble classification. However feature subset extracted has been processed using hybrid ensemble classifier using deep neural network. Classifier using Recurrent Neural Network and deep belief network follows the chunk based ensemble classification. Classifier using convolution neural network and autoencoder follows online ensemble classification. Further base classifier has been obtained on weighted adaption using deep neural network to generate the novel class or recurrent class on basis of drift analysis. Finally class imbalance data has been aggregated using sampling methods to achieve data reliability. Experimental results has proved using cross validation that proposed model outperform state of approaches in terms of computation time and accuracy.

References

- A. Amrita priya.K & Priya. R(2020),A novel Concept Specific Learning model for classification of Dynamic Evolving Data Streams towards eliminating recurring classes,*Journal of Critical Review*,Vol.7, Issue .10, pp:6315-6326
- B. Baena-Garca.M,Campo-A´ vilaJ.D, Fidalgo.R, Bifet.A, Gavalda.R, & Morales-Bueno.R (2006), Early drift detection method, in *IWKDD Proceedings of the 4th ECML PKDD International Workshop on Knowledge Discovery From Data Streams*, pp. 77–86.
- C. Baena-Garcia.M, Del Campo-Avila.J, Fidalgo.F, & Bifet.A(2006), Early drift detection method, *Proceeding in European Conference in Machine Learning Principles*, pp. 77–86
- D. Barddal.J.P, Gomes.H.M &Enembreck.F, (2015). SNCStream: A social network-based data stream clustering algorithm,” in *ACMProceedings of the 30th Annual ACM Symposium on Applied Computing (SAC)*.
- E. Ditzler.G,Muhlbaier.M & Polikar.R (2010), Incremental learning of new classes in unbalanced datasets: Learn++.UDNC , *Journal of Multiple Classifier Systems. Springer*, vol. 5997, pp. 33–42.
- F. Ditzler.G, Rosen.G & Polikar.R(2013) “Incremental learning of new classes from unbalanced data,” *The 2013 International Joint Conference in Neural Network*, pp. 1–8.
- G. Domingos.P & Hulten.G (2000) Mining high-speed data streams, *ACM Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 71–80.
- H. Gaber.M.M , Zaslavsky.A & Krishnaswamy.S (2005). Mining data streams: A review, *SIGMOD Rec.*, vol. 34, no. 2, pp. 18–26.
- I. Gama.J, liobaite.Z.A, Bifet.A,Pechenizkiy.M & Bouchachia.A(2014) A survey on concept drift adaptation *ACM Computer. Survey.*, vol. 46, no. 4, pp. 44:1–44:37.
- J. Japkowicz.N (2001),Concept-learning in the presence of between-class and within-class imbalances,”in *Advances in Artificial Intelligence. Springer Berlin Heidelberg*, vol. 2056, pp. 67–77.
- K. Minku.L.L & Xin.Y (2012) , DDD: /A new ensemble approach for dealing with concept drift, *IEEE Transaction on Knowledge and Data Engineering.*, vol. 24, no. 4, pp. 619–633,.

- L. Minku.L, White.A & Yao.X (2010). The impact of diversity on online ensemble learning in the presence of concept drift, *IEEE Transaction on Knowledge and Data Engineering*, vol. 22, no. 5, pp. 730– 742,
- M. Ramamurthy.S & Bhatnagar.R. (2007) “Tracking recurrent concept drift in streaming data using ensemble classifiers,” in *ICMLA 6th International conference on Machine Learning and Applications.*, pp. 404–409.
- N. Wankhade.K, Rane.D & Thool.R (2013), A new feature selection algorithm for stream data classification,” *International Conference on Advances in Computing, Communications and Informatics* , pp. 1843–1848.
- O. Yan.Y, Liu.Y, Shyu.M.L & Chen.M(2014), Utilizing concept correlations for effective imbalanced data classification, 15th IEEE Conference Proceeding in Information Reuse Integration pp. 561–568.