New Distributed Data Fusion Using Pregel For Large Text Dataset

Nimesh Kumar, Sufal Das

Abstract: The Data fusion method plays a vital role in analysis of large set of data. It can be used in IoT (Internet of Things) applications as well as big data analysis. Semantic data fusion plays a vital role to handle big data where data is very large in size, dynamic in nature and heterogeneous. Multi sensor data fusion is considered to combine the different data from different sensors and to give a more accurate and significant result. It is also consider as an integration of information and knowledge from different sources. Big data can be considered as large, dynamic and heterogeneous dataset. Data fusion techniques play a vital role for providing integration of knowledge from different heterogeneous data sources. Purpose of this paper is to provide an overview of data fusion techniques with big data perspective. In this paper, we have presented a complete comparison analysis of different data fusion methods proposed by different researchers with different methodologies. This paper is concluded with some open is-sues in big data fusion.

Index Terms: Big Data Analysis, Big Data Fusion, Data Aggregation, Multi-sensor Data Fusion.

1. INTRODUCTION

Big Data involves the creation of large amounts of complex data, its storage, its retrieval and finally its analysis. The term Big Data was coined by Roger Mougalas back in 2005. However, the application of big data and the quest to understand the available data is something that has been in existence for a long time. As a matter of fact, some of the earliest records of the application of data to analyses and control business activities date as far back as 7000 years. The World Wide Web contains billions of documents and is growing every year. The size of these documents may range from a few pages to over a thousand pages. The data/document that is available in the web is mostly unstructured or semi structured. Processing of this unstructured or semi structured data and retrieving new information from this web data is a challenge [6]. Fusion is a term which means to integrate. Now days, due to lot of development of technologies, many things are becoming online and the information are stored in form of data at several sites. If the organization is being spread all over the world then it is not feasible to store all the data in a single machine. For that, organization distributes in the form of its sub branch wise across the country. At each site, certain data are being stored. The techniques which are used to combine several data from different data sources and relate information from associated datasets are called Data Fusion. Using this technique one can achieve improved accuracies and more inferences that could be achieved by the use of single sensor alone. Humans and animals are the best examples of data fusion as they have evolved the capability to use multiple senses to improve their ability to survive. For example, the evaluation of the quality of edible substance may not be possible based on solely only on the vision or touch but can be achieved using combination of sight, touch, smell, and taste. Thus, multisensory data fusion is artlessly accomplished by animals or humans to get a more accurate evaluation of the surrounding environment and identification of threats, thereby increasing their chance of survival [14]. Research hers are facing problem with managing large streams of data which are generated from different sources due to rapid growth of internet technology. Thus, big data becomes very important for processing these huge, heterogeneous, stream datasets. Traditional machine analysis algorithms cannot perform well for handling these heterogeneous data. Data fusion is a method to provide integrated information from different heterogeneous information networks. This method is a very new and capable research tool to enhance the big data processing research. This method considers multi-typed data, interconnected datasets including the structured, unstructured and semi-structured dataset which are mostly available in heterogeneous information networks. Semantic analysis can be suitable for the semi-structured heterogeneous information network model with the rich semantics of linked nodes (sources) and relationships (links) of the network. It can be applied to discover the rich knowledge representation from the network [3].

- *Nimesh Kumar is an Assistant Professor in William Carey University, India, E-mail: nimeshkmr5@gmail.com*
- *Dr. Sufal Das is an Assistant Professor in North-Eastern Hill University, India. E-mail: sufal.das@gmail.com*

1.2 Data Fusion

In order to fuse data from multiple sources, different challenges are likely to occur namely imperfection, correlation, inconsistency and disparity. Imperfection means not sufficient. Further this defect can be in categorized into three types i.e. uncertainty a situation which involves imperfect or unknown information; secondly, imprecision -lack of accuracy, this impression is also further may be classified into three ways i.e. vagueness which deals with lack of preciseness, ambiguity which deals with inexact-ness and incompleteness deals with not complete in all respect and thirdly, granularity-deals with level of data. Granularity is inversely proportional to the level of data and vice versa. Correlation means relation between two or more things; same data can be linked with different sites. Inconsistency means the value of information produced by a particular data is not remain same throughout. Further, this defect also can be categorized into three types i.e. conflict- the information of some data will produce mismatching; secondly, outlier- on merging the data, some unwanted (also regarded as noise) data may involve and thirdly, disorder-ness may produce in such. Disparity means on merging the data from different sources, results in the huge difference.

1.3 Challenging Issues of Distributed Data Fusion

Data fusion is a challenging task as it has number of issues and majority of these issues are arises from the data to be fused, imperfection, and diversity of the sensor technologies. Some of the issues are as follows:

• Data Imperfection: Sensor data is always affected by unreliability as well as inexactness in the measurements. So the data fusion algorithms should be able to convey such imperfections effectively, and to exploit the data redundancy to diminish their effects.

• Conflicting Data: Data fusion method rule of combination rule of Dempster cannot handle conflicting data where more than one representation of data is available. Thus data fusion algorithms should provide special care for conflicting data by avoiding of producing counter-intuitive results.

• Data Modality: Data modality is very much important when data are considered for sensor networks which may generate data qualitatively. Different data such as audio, tabular and textual etc. can be considered together as there are different heterogeneous sensors for the networks. Thus, a data fusion scheme can be designed accordingly.

• Data Association: Data association problem is one of the most difficulties in data fusion system. Multi-objectives decision problems introduce a high complexity to data fusion system compared to the single-objective decision problem. In real dataset, dependencies among different attributes are occurred. Data fusion system may be improved by considering data association property.

• Operational Timing: Operational timing is most important issue which should be considered in sensor network environment. As sensors are overspread in environment which may reaches a vast environment group of different features. Also, operational frequency of the sensors may be different in heterogeneous sensors network. A well-designed data fusion method should incorporate multiple time scales to deal with such timing variations in data.

• Data Dimensionality: As each sensor is generating huge data, data fusion methods for sensor network face challenges due to high dimensionality of dataset. Data compression methods can be applied but compression loss is occurred while these data is pre-processed locally at each of the sensor nodes.

• Ambiguity: The ambiguities and inconsistencies present in the environment also cause uncertainty in sensor. It is not necessary always that the uncertainty occurs due to impreciseness and noise in measurements [7]. Data fusion algorithms should be able to exploit the redundant data to alleviate such effects.

With respect to merge the data simultaneously from different sources, certain issues are likely to occur.

1. What architecture should be used (i.e., where in the processing flow should data be fused (viz. at the data, feature, or decision levels) such that no any further discrepancy will likely to occur?
2. Which algorithms or techniques are appropriate and optimal for particular data applications?
3. How much accuracy can realistically be attained by a data fusion process?
4. How should individual sensor data be processed to extract the at most amount of useful information?
5. Which form of the data collection environment (i.e., signal propagation, target characteristics etc.) affect the processing?
6. For what kind of environment multi-sensor data fusions improve system operation?

2 BACKGROUND STUDY

Every information which may exist in the form of text, image, video etc. is present in form of data. Growing of data is rapidly increasing by which stored at different sites in division manner. In general, the integration of required data from multiple sources are attaining in hierarchical manner. Prior to fuse data, some pre-processing is attained. While fusion, fusion point need to merge the data and update periodically which symbolizes event processing. Event processing is a method of tracking and processing data stream to deriving a conclusion from them. Complex Event Processing or CEP is event processing which combines data from multiple sources to infer events or patterns, which aims to derive meaningful events and respond to them as early as possible. Data can also be merged with the concept of Drools fusion technique which deals with streams of events instead of static data. CEP engines are designed to process a large volume of events at extremely high speeds. Throughput requirements are often well over 100,000 events per second, while processing latency demands can be as low as one millisecond or less. Drools help in processing streams to perform tasks such as threat detection, real-time event monitoring, tracking user behavior and many more. A sensor designed to detect and respond to the presence of a flame or fire, allowing flame detection is known as a flame detector. On merging the data in a hierarchical manner, the data is fused in several layers. Primarily, data at different sources does not have any idea about the global nature, firstly, communicate locally then merged data deals globally which is to combine from multiple sources. On fusion, merged data bears some lack of information which takes place by considering them as outlier, results to process faster and minimizes the cost of computation. Majorly, existing data fusion approaches uses the concept of buffer, which works with the concept of stampede. Such buffer does not handle the data if arrives late as per designated stampede and treat them as noise which result in the lack of accuracy on the fused data. Parallel fusion plays a vital role to decrease the time consuming factor upon fusing data from multiple sources.

3 LITERATURE SURVEY

In this section, several related research works are discussed and finally some open issues in this area are mentioned.

3.1 Multi-sensor Data fusion Using Neural Networks [2]: In this work, a linear single layer neural network measurement fusion for multi sensor network is presented based on Hebbian learning. In this work, an algorithm is introduced which is unsupervised learning algorithm. Authors have performed comparison with traditional fusion methods based on Kalman filtering such as State Vector fusion etc.

• Description:

Neural Network Based Multi-sensor Data Fusion: - A common application of data fusion techniques is the estimation of target position or kinematic information from multiple measurements from a single or multiple sensors. Two essential processes are involved in the derivation of position or kinematic information: data association and state estimation. State estimation refers to the optimal estimation of those values, e.g., the position, velocity, acceleration, angular position etc. of the target, from

observation data. The Kalman filter is the best known and most widely applied state estimator algorithm. The Kalman filter gives a linear, unbiased, and minimum error variance recursive algorithm to optimally estimate the unknown state of a linear dynamic system from Gaussian distributed noisy observations. The Kalman filtering process can be considered as a prediction-update formulation. The algorithm uses a predefined (linear) model of the sys-tem to predict the state at the next time step. Added to this is a component to update for errors in the model using the actual observations of the system? The prediction and update are combined using the Kalman gain which is calculated to minimize the mean-square error of the state estimate.

Neural network is a good tool to construct relationship between inputs and output. It is a method designed to mimic a theory of how biological nervous system work. Here, in this paper neural network methodology is considered to solve the fusion problem by relaxing the limitation of traditional methods. To represent sensor data neurons can be trained and through associate recall, the combination of neurons can be activated in response to different sensory stimuli. The learning algorithms for neural networks are classified as supervised and unsupervised (self-organized) learning. The purpose of an algorithm for self-organized learning is to discover significant patterns or features in the input data, and to do the discovery without a teacher.

• Method Analysis: The algorithm is implemented through traditional and artificial neural networks methods. For traditional method (Kalman), the radar data in spherical coordinates (Range, Azimuthal and Elevation) is filtered using the Kalman filter. Then the data is converted to the X, Y and Z coordinates before applying any fusion method on the obtained data. The measurement noise co-variance matrix (R) and system noise covariance matrix (Q) are considered to be Gaussian. For training the neural network an unsupervised Hebbian adaptive rule is used and the learning parameter selected is 0.1.The performance comparison of both the methods in terms of mean square error.

• Limitations of Neural Network Based Data Fusion Model: The limitation of this proposed neural network based method is its pre -processing requirement wherein the inputs are to be normalized. The proposed method given in this paper does not work if the noise level is too high.

3.2 Multi-Sensor Data Fusion Algorithms for Target Tracking using Multiple Measurements [3]:

Multi-Sensor Data Fusion (MSDF) is very rapidly growing as an independent discipline to be considered with and finds applications in many areas. Surplus and complementary sensor data can be fused using multi-sensor fusion techniques to enhance system competence and consistency. The objective of this work is to evaluate multi sensor data fusion algorithms for target tracking. Target tracking using observations from several sensors can achieve improved estimation performance than a single sensor. In this work, three data fusion algorithms based on Kalman filter namely State Vector Fusion (SVF), Measurement Fusion (MF) and Gain fusion (GF) are implemented in a tracking system. Using MATLAB, these three methods are compared and performance metrics are computed for the evaluation of algorithms. The results show State Vector Fusion estimates the states well when compared to Measurement Fusion and Gain Fusion.

• Description: Multi-sensor data fusion (MSDF) is defined as the process of combining information from multiple sources to produce the most precise and complete unified data about an entity, activity or event [3]. MSDF is the combination and application of many conventional disciplines and new areas of engineering. The measurement value taken from a single sensor is not accurate and not re liable. It consumes more time. The spatial coverage of a single sensor is also low. Compared to the single sensor measurement, the MSDF gives more accurate, reliable and timely data. It covers a larger geometrical area and the results obtained are fault tolerant. In this paper three different fusion algorithms are considered and employed in a tracking system [8]. The performance metrics are computed and the results obtained are reported descriptively. The goal of this work is to perform multi-sensor data fusion algorithms for target tracking. Deploying different sensors, target tracking can be achieved to improve estimation performance. In this work, three data fusion algorithms are applied based on Kalman filter (SVF), Measurement Fusion (MF) and Gain fusion (GF) which are implemented in the tracking system. This paper shows how State Vector Fusion method estimates the states as well as provides comparison with Measurement Fusion and Gain Fusion. Authors have established their work with implementation using RADAR data and performance analysis based on Percentage Fit Error (PFE), Mean Square Error (MSE) and Mean Absolute Error (MAE).

3.2.1 State Vector Fusion

State Vector Fusion (SVF) is a Kalman Filter (KF) based data fusion. The KF [1] is given for each set of observations, i.e., the algorithm is applied independently for each sensor (data) and generates state estimates. Each sensor uses an estimator that obtains an estimate of the state vector and its associated covariance matrices (of the tracked target) from the data of that associated sensor. Then at the fusion Centre, track-to-track correlation is carried out and the fused state vector is obtained [9].

3.2.2 Measurement Fusion

Measurement fusion (MF) algorithm fuses the sensor observations directly via a measurement model and uses one KF [1, 2] to estimate the fused state vector.

3.2.3 Gain Fusion

In the Gain fusion (GF) algorithm [4, 5], a global processor receives the information in the form of a Kalman gain from local systems and formulates the global estimate.

• Method Analysis: Here, tracking system is implemented using Kalman Filter based State Vector Fusion (SVF), Measurement Fusion (MF) and Gain Fusion (GF) algorithms. From the performance measure it is found that the Kalman Filter based State Vector Fusion algorithm has comparatively better efficiency than others.

3.3 Distributed Data Fusion for the Internet of Things [4]

Wireless communication enabled connecting previously disconnected embedded de-vices into the global network, facilitating device discovery, querying and interaction. Examples of such wireless networking technologies, actively used in the context of complex distributed IoT systems, include LPWAN, Bluetooth Low Energy, Zig-Bee, Wi-Fi, etc. These technologies differ in their data transferring range, throughput,

and power consumption. Given the extreme amounts of generated data, a potentially promising solution is to perform data filtering/aggregation – i.e. data fusion – as close to the source of data as possible, thus minimizing the amount of ‗noisy' data being sent over throughput-limited wireless links. However, this is not always possible due to the lack of computing resources on embedded systems and constrained devices. There-fore, a solution, able to meet local resource restrictions, while reducing the overhead by keeping computation as close to data sources as possible, is required.

• Description: This paper proposed a hierarchical multi-level architecture for data fusion in IoT systems. According to the proposed approach, data should be firstly processed on-board (i.e. locally on IoT nodes) whenever possible, or pushed to network devices and services, and finally, to the cloud in a hierarchical manner. The proposed architecture includes three conceptual levels, which are also aligned with geographical areas, from which sensor data are collected. Local area data fusion (LADF) is supposed to take place on edge devices, which collect data, coming from embedded sensors. The amount of data is relatively small, and data fusion can be performed on-board. Wide area data fusion (WADF) refers to performing more intensive data analytics, pushed from a wider net-work of edge devices. Following the principles of Edge Computing, WADF is performed on communication and processing units. Global area data fusion (GADF) refers to the highest level of data fusion, which provides a global view on the whole managed system of edge devices and networking nodes. This involves processing of large amounts of data, and is therefore expected to be implemented in a data Centre or a cloud platform. The prototype utilizes Drools Fusion as the underlying CEP middleware. As a result, the fusion middleware was deployed at three levels of the hierarchical architecture. The low-level smart objects, equipped with multiple sensing devices (i.e. temperature, humidity, light sensors and motion detectors) are represented by Rasp-berry Pi boards. These boards are responsible for collecting relevant raw sensor data from a local area (i.e. a room) and perform CEP – that is, data fusion takes place as close to the source of data as possible, such that only filtered/aggregated values are transferred to an upper-level processing node. The goal of this LACEP is to detect if someone is still indoors, or if electrical appliances (e.g. heaters, A/C, lights) are still on, and to send a corresponding alert. The middle-level communication/processing units are represented by a server, responsible for collecting data from multiple Raspberry Pi boards and perform data fusion over a larger area of interest. This way, WACEP provides a more extensive perspective on the man-aged area (i.e. a university building). The highest-level processing location is represented by the Amazon Web Services cloud, which is responsible for GA-CEP – i.e. collecting data from all the managed buildings and performing data fusion over the whole university campus area.

• Method Analysis: This approach can handle mega-/gigabytes of data. This requirement can be addressed by pushing intelligence to the edge of an IoT net-work – i.e. closer to the data source and exploiting resources and capabilities of edge nodes for data processing. Accordingly, this paper proposes a data fusion approach based on a three-level hierarchical architecture using CEP techniques. Data are first sensed and processed locally on-board (i.e. sensor fusion) and then, if required, further processed by higher-level (wide and/or global) data fusion engines deployed on a server and/or a cloud platform. This way, a scalable architecture is achieved, in which local computational resources, if insufficient, are extended by the higher levels.

4 TEXT SUMMARIZATION

In this section, several related summarization works are discussed.

4.1 Text: In information technology, text is a human -readable sequence of character s and the words they form that can be encoded into computer-readable formats such as ASCII. Text is usually distinguished from non-character encoded data, such as graphic images in the form of bitmap s and program code, which is sometimes referred to as being in "binary" (but is actually in its own computer-readable format).

4.2 Summarization: Summarization is an important problem in many domains involving large datasets. Summarization can be essentially viewed as transformation of data into a concise yet meaningful representation which could be used for efficient storage or manual inspection.

4.3 Text Summarization: Text summarization is the problem of creating a short, accurate, and fluent summary of a longer text document. Automatic text summarization methods are greatly needed to address the ever-growing amount of text data available online to both better help discover relevant information and to consume relevant information faster. There is an enormous amount of textual material, and it is only growing every single day. Think of the internet, comprised of web pages, news articles, status updates, blogs and so much more. The data is unstructured and the best that we can do to navigate it is to use search and skim the results. There is a great need to reduce much of this text data to shorter, focused summaries that capture the salient details, both so we can navigate it more effectively as well as check whether the larger documents contain the information that we are looking for. “Textual information in the form of digital documents quickly accumulates to huge amounts of data. Most of this large volume of documents is unstructured: it is unrestricted and has not been organized into traditional databases. Processing documents is therefore a perfunctory task, mostly due to the lack of standards”. Text summarization is the process of distilling the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks). Automatic text summarization, or just text summarization, is the process of creating a short and coherent version of a longer document. We (humans) are generally good at this type of task as it involves first understanding the meaning of the source document and then distilling the meaning and capturing salient details in the new description. As such, the goal of automatically creating summaries of text is to have the resulting summaries as good as those written by humans. There are many reasons and uses for a summary of a larger document. One example that might come readily to mind is to create a concise summary of a long news article, but there are many more cases of text summaries that we may come across every day.

4.4 Approaches for Text Summarization:

There are two main approaches to summarizing text documents; they are:

- Extractive Methods.

- Abstractive Methods.

The different dimensions of text summarization can be generally categorized based on its input type (single or multi document), purpose (generic, domain specific or query-based) and output type (extractive or abstractive). Extractive text summarization involves the selection of phrases and sentences from the source document to make up the new summary. Techniques involve ranking the relevance of phrases in order to choose only those most relevant to the meaning of the source. Abstractive text summarization involves generating entirely new phrases and sentences to capture the meaning of the source document. This is a more challenging approach, but is also the approach ultimately used by humans. Classical methods operate by selecting and compressing content from the source document. Classically, most successful text summarization methods are extractive because it is an easier approach, but abstractive approaches hold the hope of more general solutions to the problem.

4.5 Modules of Summarizing

A Text Summarization tool must have three components corresponding to the above three states – Text pre-processor, Text Analyzer and Machine Language engine. There are numerous algorithms applicable to each of the above components. The process of text summarization can be decomposed into 'Analysis' - analyzing the input text/documents and selecting a few salient features, 'Transformation' - transforming the results of the analysis into a summary representation, and 'Synthesis' - taking the summary riper sensation, and producing an appropriate summary aligned to users' specifications.

Type of summarization methods	Subtype	Concept	Advantages	Disadvantages	Application/ Work Done
1.Approaches Figures	Abstractive	It is the process of reducing a text document in order to create a summary that is semantically related	Good compression ratio. More reduced text and semantically related summary	Difficult to compute	SUMM RIST
	Extractive	It consists of selecting important sentences from original document based on statistical features	Easy to compute because it does not deal with the semantics and more successful	Suffers from inconsistencies, lack of balance, results in lengthy summary	Summ It applet, designed by Surrey University [15]
2.Details	Indicative	It only presents main idea of text to user. They can be used to quickly decide whether a text is worth reading	Encourages the users to read the main document in depth. Used for quick categorization and easier to produce	Detailed information is not present	Information present on the back of the movie pack or novels Length 5 to 10%
	Informative	Gives concise information of the main text	Serves as a substitution for the main document	Does not provide quick overview	SumUM Length 20 to 30%
3.Content	Generic	Generalized summary irrespective of the type of user. Information is at same level of importance	Can be used by any type of user	It provides an author's view not user specific	SUMM ARIST
	Query based	User has to determine the topic of original text in the form of query and system only extract that information	Specific information can be searched. It reflects user's interest	Not used by any type of user. It is based on type of user	Mitre's WebSumm
4.Limitation	Genre specific	Accept only special type of text as input	Overcome s the problem of summarizing heterogeneous document	Limitation template of the text	News blaster
	Domain Independent	Can accept any type of text.	Any type of text input is accepted. It is not domain dependent	Difficult to implement	Copy and Paste system
5.Number of input document	Single document	Can accept only one input document	Less overhead	Cannot summarize multiple documents of related topics	Copy and paste system
	Multi document	Can accept multiple input documents	Multiple document s of same topic can be summarized to single document	Difficult to implement	SUMM ONS Designed by Columbia university [15]
6.Language	Mono Lingual	Can accept input only with specific language and output is based	Need to work with only one language	Cannot handle different language	FarsiSum

		on that language			
	Multi Lingual	Can accept documents in different language	Can deal with multiple language	Difficult to implement.	SUMM ARIST(English, Japanese, Spanish)

5. DISTRIBUTIVE TEXT SUMMARIZATION

5.1 Distributive System

In distributed database system, the database is shared on several computers. The computers in a distributed system communicate with one another through various communication media, such as high-speed networks or telephone lines. They do not share main memory or disks. The computers in distributed system may vary in size and function, ranging from workstations up to mainframe systems. The computers in distributed system are referred to by a number of different names, such as Sites or Nodes depending on the context in which they are mentioned. A distributed database system consists of single logical database which is split into different fragments. Each fragment is stored on one or more computers under the control of separate DBMS with computers connected by communication network. Each site is capable of independently processing user requests that require access to local data or file system and is also capable of performing processing on remote machines in the network.

Characteristics of Distributed System:

- Data set can be split in to fragments and can be distributed across different nodes within network.
- Individual data fragments can be replicated and allocated across different nodes.
- Data at each site is under control of a DBMS.
- DBMS at each site can handle local applications autonomously.
- Each DBMS site will participate in at least one global application.

Differences between shared nothing Parallel System and

Distributed system is:

1. In distributed system, databases are geographically separated; they are administered separately and have slower interconnection.

2. In distributed systems, we differentiate between local and global transactions. Local transaction is one that accesses data in the single site at that the transaction was initiated. Global transaction is one which either accesses data in a site different from the one at which the transaction was initiated or accesses data in several different sites.

Advantages of Distributed System:

 Sharing Data: There is a provision in the environment where user at one site may be able to access the data residing at other sites.

 Autonomy: Because of sharing data by means of data distribution each site is able to retain a degree of control over data that are stored locally.

 In distributed system there is a global database administrator responsible for the entire system. A part of global data base administrator responsibilities is delegated to local data base administrator for each site. Depending upon the design of distributed database

 Each local database administrator may have different degree of local autonomy.

 Availability: If one site fails in a distributed system, the remaining sites may be able to continue operating. Thus a failure of a site doesn't necessarily imply the shutdown of the System.

Disadvantages of Distributed Systems:

The added complexity required to ensure proper co-ordination among the sites, is the major disadvantage. This increased complexity takes various forms :

 Software Development Cost: It is more difficult to implement a distributed database system; thus it is more costly.

 Greater Potential for Bugs: Since the sites that constitute the distributed database system operate parallel, it is harder to ensure the correctness of algorithms, especially operation during failures of part of the system, and recovery from failures. The potential exists for extremely subtle bugs.

 Increased Processing Overhead: The exchange of information and additional computation required to achieve interstice co-ordination are a form of overhead that does not arise in centralized system.

5.2 Algorithm for Distributive Text Summarization

STEP-1. Divide the given graph into „n" chunks

STEP- 2. At each node, there is a text file.

STEP-3. Generate Ki ; Ki C(Keyword, weight) by parsing each text file. Weight Є (occurrence, position, highlight).

STEP-4. Compute ∑ = K ; parallely at each chunk using Pregel (distributive computing).

STEP-5. Map weight of K, say (0-1). Declare one threshold value of weight say Wt.

STEP-6. Now, match each text file (line by line) with „K"

if weight is >= Wt
then combine
else
discard.

STEP-7. Hence, in this way a final text summarization will obtain.

6 RESULT ANALYSIS

To analyze our designed tool, we have trained our tool from DUC (Document Understanding Conference) – 2007 (https://duc.nist.gov/duc2007/tasks.html). Here, data is in the form of text. ROUGE metric was introduced by Lin et al. (2003)19 and has been adopted by the DUC and leading conferences on Natural Language Processing. ROUGE calculates the overlap between the candidate summaries and the reference summaries and it has been found that correlation of ROUGE-1 and ROUGE-2 is the most with human summaries (Lin, C.Y. et al.21). ROUGE-N is a recall measure that computes the number of matches between the candidate summaries and the reference summaries.

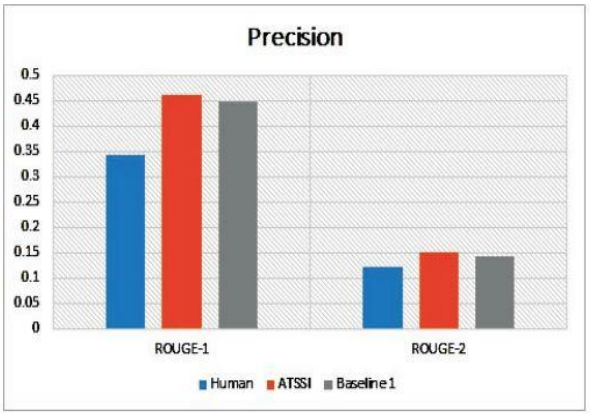

***Fig. 6.1**. Evaluated Precision on Dataset 37 with Human Summary and Baseline 1 vs ATSSI*

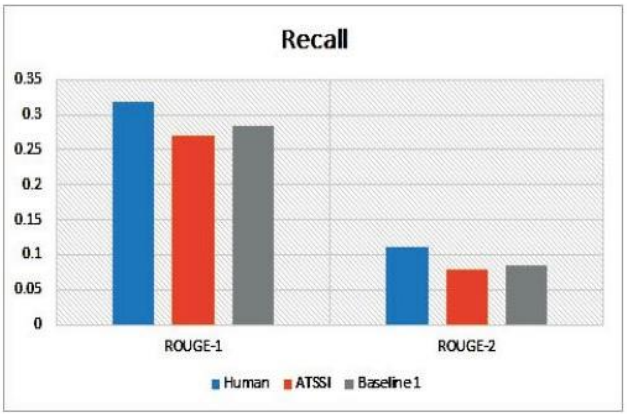

***Fig. 6.2.** Evaluated Recall on Dataset 37 with Human Summary and Baseline 1 vs ATSSI.*

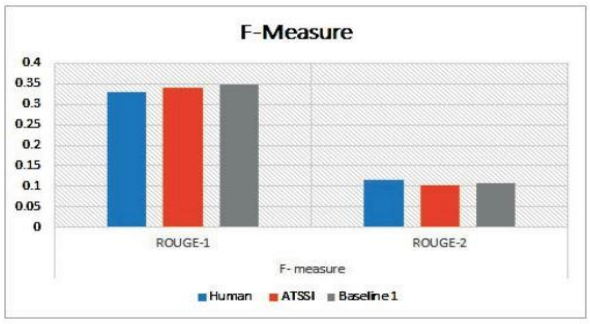

***Fig.6.3**. Evaluated F-measure on Dataset 37 with Human Summary and Baseline 1 vs ATSSI.*

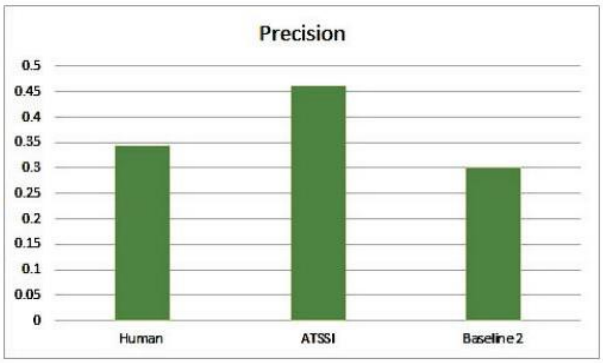

***Fig. 6.4.** Evaluated Precision on DUC-2007 Dataset with Human Summary and Baseline 2 vs ATSSI.*

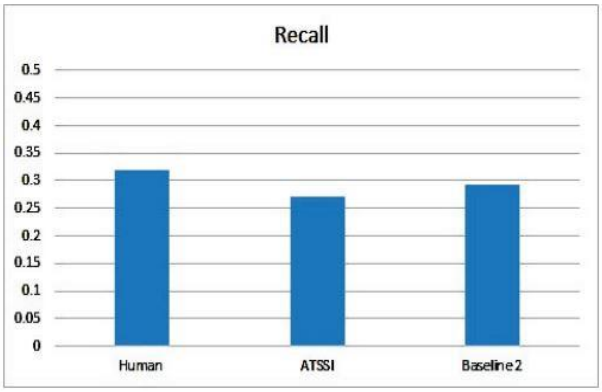

***Fig. 6.5**. Evaluated Recall on DUC-2007Dataset with Human Summary and Baseline 2 vs ATSSI.*

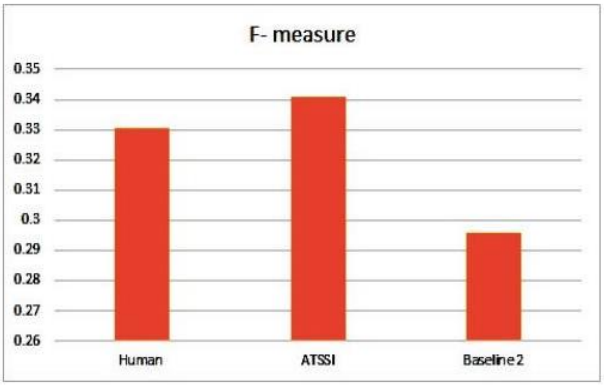

***Fig. 6.6.** Evaluated F-measure on DUC-2007 Dataset with Human Summary and Baseline 2 vs Proposed System.*

Figure 6.1 shows that ATSSI has higher precision over the baseline 1, this is because we have overcome the demerit of the approach as stated by 37, that it can only connect the sentences if there is a pre-existing connector. Since the dataset used by 37 already had redundant data with connectors, ATSSI is only showing a marginal difference in precision. Recall on Dataset 37 is resulting in low recall as shown in Fig. 6.2 because there is a high presence of redundant information on dataset which is leading to infusing of maximum number sentiments in a particular sentence, which in turn results in low frequency of sentences. When comparing to Baseline 2 37, our algorithm outperforms by 13 percent (Fig. 6.6), since Baseline 2 37 describes rigid rules for ensuring sentence correctness and has no provision for fusing sentences. ATSSI outperforms Baseline 2 37 by a huge margin in precision comparison since ATSSI incorporates sentiment infusion and a provision for removing redundancy. Also recall of ATSSI is marginally low as that of Baseline 2 for similar reasons mentioned for baseline 1. Finally, it is worth noting that generating summaries that are purely abstractive in nature is an onerous task, as shown by F. Liu et al. 20 where F-measure values are in the range 13% to 18%.

7 CONCLUSIONS

With the fast growing of technologies, eventually, data is also increasing rapidly. So, it is necessary to store data efficiently

and to process fast, the data must present in a distributive fashion. For any query, required set of data must be fetched from multiple sources, after fusion of these data, desired query can be executed as early as possible. Here, we presents brief introduction of data fusion techniques and the analysis of different survey papers related to data fusion and its techniques used in different areas. During study of literature it have been seen that in most of the papers the traditional data fusion technique is compared to new proposed technique. Though there are several approaches, techniques, models and algorithms exist to deal with fusion of data from multiple sources but no one is perfect till date. Some acquiring time complexity issue, some with lack of accuracy, some unable to process different data formats. On our model, the accuracy is higher and time complexity is lesser as compared to existing models. As accuracy matters a lot for data summarization to get the proper information while at the same time, on summarization, time must not consume more so that desired query can be executed in short span of time, which also decrease traffic.

ACKNOWLEDGMENT

We express my sincere thanks to all those people and coordinator associated with Information Technology department, NEHU Shillong-793022 for their valuable suggestions and help in preparing this paper. Finally, it is our foremost duty to thank my respondents who helped us to complete this work without which this paper would not have been possible.

REFERENCES

[1] Das, S. S., Deka, N., Sinha, N., Dhar, S., Bhattacharjee, D.,and Gupta, S. Environmental monitoring using sensor data fusion. In Radar, Communication and Computing (ICRCC), International Conference on, 83-86. IEEE, 2012.

[2] Yadaiah, N., Singh, L., Bapi, R. S., Rao, V. S., Deekshatulu, B. L., and egi, A. Multisensor data fusion using neural networks. In Neural Networks, IJCNN'06. International Joint Conference on, 875-881. IEEE, 2006.

[3] Anitha, R., Renuka, S., and Abudhahir, A. Multi sensor data fusion algorithms for target tracking using multiple measurements. In Computational Intelligence and Computing Research (ICCIC), IEEE International Conference on, 1-4. IEEE, 2013.

[4] Dautov, Rustem, and Salvatore Distefano. "Distributed Data Fusion for the Internet of Things." In International Conference on Parallel Computing Technologies, pp. 427-432. Springer, Cham, 2017.

[5] Nicholson, D., C. M. Lloyd, S. J. Julier, and J. K. Uhlmann. "Scalable distributed data fusion." In Information Fusion, Proceedings of the Fifth International Conference on, vol. 1, pp. 630-635. IEEE, 2002.

[6] Kumar, Rajnish, Matthew Wolenetz, Bikash Agarwalla, JunSuk Shin, hillip Hutto, Arnab Paul, and Umakishore Ramachandran. "DFuse: A framework for distributed data fusion." In Proceedings of the 1st international conference on Embedded networked sensor systems, pp. 114-125. ACM, 2003.

[7] Sanchez-Riera, Jordi, Kai-Lung Hua, Yuan-Sheng Hsiao, Tekoing Lim, Shintami C. Hidayati, and Wen-Huang Cheng. "A comparative study of data fusion for RGB-D based visual recognition." Pattern Recognition Letters 73 1-6 2016.

[8] Fourati, H. Multisensor Data Fusion: From Algorithms and architectural Design to Applications. CRC Press, 2015.

[9] Rashinkar, P. and Krushnasamy, V. An overview of data fusion techniques. In Innovative Mechanisms for Industry Applications (ICIMIA), International Conference on, 694-697. IEEE, 2017.

[10] Laaraiedh, M. Implementation of kalman filter with python language. arXiv preprint arXiv:1204.0375 2012.

[11] Bhattacharya, S. and Raj, R. A. Performance evaluation of multisensor data fusion technique for test range application. Sadhana 29(2), 237-247 2004.

[12] Sapna, S. Fusion of big data and neural networks for predicting thyroid. In Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), International Conference on, 243-247. IEEE, 2016.

[13] David, J., Krishnan, R., and Kumar, S. Neural network based retinal image analysis. In Image a nd Signal Processing, CISP'08. Congress on, volume 2, 49-53. IEEE, 2008.

[14] Ramgopal, Iyer Abhiram, and P. V. Manivannan. "Development of multi-sensor data fusion technique for the automated Bus Rapid Transport System." In Robotics: Current Trends and Future Challenges (RCTFC), International Conference on, pp. 1-6. IEEE, 2016.

[15] Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. Multisensor data fusion: A review of the state-of-the-art. Information Fusion 14(1), 28-44 2013.

[16] Azimirad, E., Haddadnia, J., and Izadipour, A. A comprehensive review of the multi-sensor data fusion architectures. Journal of Theoretical & Applied Information Technology 71(1) 2015.

[17] Mahler, R. P. Statistical multisource-multitarget information fusion. Artech House, Inc.,.F.: Article title. Journal 2(5), 99–110 2016.

[18] Das, Siddharth Sankar, et al. "Environmental monitoring using sensor data fusion." Radar, Communication and Computing (ICRCC), 2012 International Conference on. IEEE, 2012.

[19] Sanchez-Riera, Jordi, et al. "A comparative study of data fusion for RGB-D based visual recognition." Pattern Recognition Letters 73 (2016): 1-6.

[20] Zikopoulos, Paul, and Chris Eaton. Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.

[21] Rathee, Sanjay, and Arti Kashyap. "Adaptive-Miner: an efficient distributed association rule mining algorithm on Spark." Journal of Big Data 5.1 (2018): 6.

[22] FERNANDEZ-BASSO, C. A. R. L. O. S., M. DOLORES RUIZ, and MARIA J. MARTIN-BAUTISTA. "Extraction of association rules using big data technologies." International Journal of Design & Nature and Ecodynamics 11.3 (2016): 178-185.

[23] Han, Jiawei, Jian Pei, and Yiwen Yin. "Mining frequent patterns without candidate generation." ACM sigmod record. Vol. 29. No. 2. ACM, 2000.

[24] Borgelt, Christian. "An Implementation of the FP-growth Algorithm." Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations. ACM, 2005.

[25] Kaur, Manjitkaur. "ECLAT Algorithm for Frequent Itemsets Generation." International Journal of Computer Systems 1.3 (2014): 82-84.

[26] Borgelt, Christian. "Efficient implementations of apriori

and eclat." FIMI'03: Proceedings of the IEEE ICDM workshop on frequent itemset mining implementations. 2003.
[27] Hazarika, Manalisha, and Mirzanur Rahman. "Mapreduce based eclat algorithm for association rule mining in datamining: mr_eclat." International Journal of Computer Science and Engineering 3.1 (2014): 19-28.
[28] Sun, Zhaohao. "10 Bigs: Big Data and Its Ten Big Characteristics." (2018).
[29] Zikopoulos, Paul, and Chris Eaton. Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.
[30] Xiao-Long, WANG Yuan-Zhuo JIN, and Xue-Qi CHENG. "Network Big Data: Present and Future [J]." Chinese Journal of Computers 6 (2013): 001.
[31] Sutton, Richard S., and Andrew G. Barto. Introduction to reinforcement learning. Vol. 135. Cambridge: MIT press, 1998.
[32] Jaseena, K. U., and Julie M. David. "Issues, challenges, and solutions: Big data mining." Computer Science & Information Technology (CS & IT) (2014): 131-140.
[33] Shvachko, Konstantin, et al. "The hadoop distributed file system." Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on. Ieee, 2010.
[34] Wani, Mudasir Ahmad, and Suraiya Jabin. "Big Data: Issues, Challenges, and Techniques in Business Intelligence." Big Data Analytics. Springer, Singapore, 2018. 613-628.
[35] Zikopoulos, Paul, and Chris Eaton. Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.
[36] Ularu, Elena Geanina, et al. "Perspectives on big data and big data analytics." Database Systems Journal 3.4 (2012): 3-14.