Research Article Impact Factor: 4.226 ISSN: 2319-507X
M Jaganathan, IJPRET, 2014; Volume 3 (5): 1-12 IJPRET

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

A NEXT-GEN DATA FUSION – BIG DATA FUSION

M. JAGANATHAN

Asst. Prof., CSE dept., Apollo Engineering College, Chettipedu, Poonamalle, Tiruvallur.

Accepted Date: 28/11/2014; Published Date: 01/01/2015

Abstract: Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process them using traditional data processing applications. Big data is a popular term used to describe the exponential growth and availability of data, both structured and unstructured. More accurate analyses may lead to more confident decision making. And better decisions can mean greater operational efficiencies, cost reductions and reduced risk. The challenges include analysis, capture, curation, search, sharing, storage, transfer, visualization, and privacy violations. The trend to larger data sets is due to the additional information derivable from analysis of a single large set of related data, as compared to separate smaller sets with the same total amount of data, allowing correlations to be found to "spot business trends, prevent diseases, combat crime and so on.

Keywords: Big Data Fusion, Storage, Visualization

Corresponding Author: MR. M. JAGANATHAN

Access Online On:

www.ijpret.com

How to Cite This Article:

M Jaganathan, IJPRET, 2014; Volume 3 (5): 1-12

PAPER-QR CODE

INTRODUCTION

Big data defined:

As far back as 2001, industry analyst Doug Laney (currently with Gartner) articulated the now mainstream definition of big data as the three Vs of big data: volume, velocity and variety1.

Volume. Many factors contribute to the increase in data volume. Transaction-based data stored through the years. Unstructured data streaming in from social media. Increasing amounts of sensor and machine-to-machine data being collected. In the past, excessive data volume was a storage issue. But with decreasing storage costs, other issues emerge, including how to determine relevance within large data volumes and how to use analytics to create value from relevant data.

Velocity. Data is streaming in at unprecedented speed and must be dealt with in a timely manner. RFID tags, sensors and smart metering are driving the need to deal with torrents of data in near-real time. Reacting quickly enough to deal with data velocity is a challenge for most organizations.

Variety. Data today comes in all types of formats. Structured, numeric data in traditional databases. Information created from line-of-business applications. Unstructured text documents, email, video, audio, stock ticker data and financial transactions. Managing, merging and governing different varieties of data is something many organizations still grapple with.

At SAS, we consider two additional dimensions when thinking about big data:

Variability. In addition to the increasing velocities and varieties of data, data flows can be highly inconsistent with periodic peaks. Is something trending in social media? Daily, seasonal and event-triggered peak data loads can be challenging to manage. Even more so with unstructured data involved.

Complexity. Today's data comes from multiple sources. And it is still an undertaking to link, match, cleanse and transform data across systems. However, it is necessary to connect and correlate relationships, hierarchies and multiple data linkages or your data can quickly spiral out of control.

Big data Integration=Big data +data integration.

2-Why bigdata?:-

The real issue is not that you are acquiring large amounts of data. It's what you do with the data that counts. The hopeful vision is that organizations will be able to take data from any source, harness relevant data and analyze it to find answers that enable 1) cost reductions, 2) time reductions, 3) new product development and optimized offerings, and 4) smarter business decision making. For instance, by combining big data and high-powered analytics, it is possible to:

Determine root causes of failures, issues and defects in near-real time, potentially saving billions of dollars annually.

Optimize routes for many thousands of package delivery vehicles while they are on the road.

Analyze millions of SKUs to determine prices that maximize profit and clear inventory.

Generate retail coupons at the point of sale based on the customer's current and past purchases.

Send tailored recommendations to mobile devices while customers are in the right area to take advantage of offers.

Until recently, organizations have been limited to using subsets of their data, or they were constrained to simplistic analyses because the sheer volumes of data overwhelmed their processing platforms. But, what is the point of collecting and storing terabytes of data if you can't analyze it in full context, or if you have to wait hours or days to get results? On the other hand, not all business questions are better answered by bigger data. You now have two choices:

Incorporate massive data volumes in analysis. If the answers you're seeking will be better provided by analyzing all of your data, go for it. High-performance technologies that extract value from massive amounts of data are here today. One approach is to apply high-performance analytics to analyze the massive amounts of data using technologies such as grid computing, in-database processing and in-memory analytics.

Determine upfront which data is relevant. Traditionally, the trend has been to store everything (some call it data hoarding) and only when you query the data do you discover what is relevant. We now have the ability to apply analytics on the front end to determine relevance based on

context. This type of analysis determines which data should be included in analytical processes and what can be placed in low-cost storage for later use if needed

Data silo and complexity challenge – Effective predictive analytics applications leverage data from multiple internal and external sources, including relational, semi-structured XML, dimensional MDX, and the new "Big Data" data types such as Hadoop.

Query performance challenge – Large volumes of data must be analyzed making query performance a critical success factor.

Agility challenge – Dynamic businesses require new and ever changing analyses. This means new data sources need to be brought on board quickly and existing sources modified to support each new analytic requirement.

Massively Parallel Processing based Appliances – Examples include EMC Greenplum, HP Vertica, IBM Netezza, SAP Sybase IQ, and more

Columnar/tabular NoSQL Data Stores – Examples include Hadoop, Hypertable, and more

XML Document Data Stores – Examples include CouchDB, MarkLogic, and MongoDB, and more

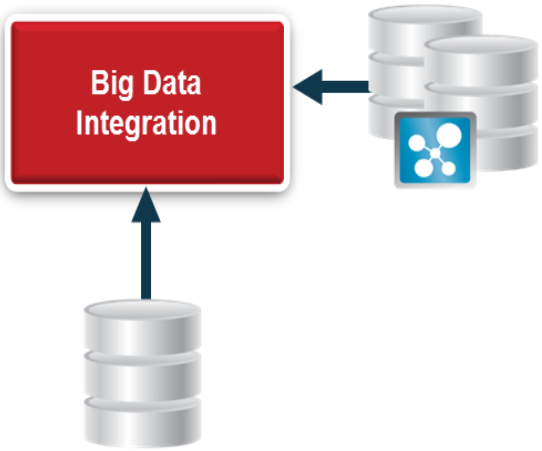


Figure 1

3- Big Data Integration Challenges:-

3.1) BDI: Data Fusion :-

Data fusion refers to resolving conflicts from different sources and finding the truth that reflects the real world [2], [1]. Unlike schema mapping and record linkage, data fusion is a new field

that has emerged only recently. Its motivation is exactly the veracity of data: the Web has made it easy to publish and spread false information across multiple sources and so it is critical to separate the wheat from the chaff for presenting high quality data. Data fusion is the process of integration of multiple data and knowledge representing the same real-world object into a consistent, accurate, and useful representation

Figure 2

Data fusion techniques have been extensively employed on multisensor environments with the aim of fusing and aggregating data from different sensors; however, these techniques can also be applied to other domains, such as text processing. The goal of using data fusion in multisensor environments is to obtain a lower detection error probability and a higher reliability by using data from multiple distributed sources.

The available data fusion techniques can be classified into three nonexclusive categories:

(i) data association, (ii) state estimation, and (iii) decision fusion. Attending to the relations between the input data sources, as proposed by Durrant-Whyte [3]. These relations can be defined as (a) complementary, (b) redundant, or (3) cooperative data; (1)complementary: when the information provided by the input sources represents different parts of the scene and could thus be used to obtain more complete global information. For example, in the case of visual sensor networks, the information on the same target provided by two cameras with different fields of view is considered complementary;(2)redundant: when two or more input sources provide information about the same target and could thus be fused to increment the

confidence. For example, the data coming from overlapped areas in visual sensor networks are considered redundant;(3)cooperative: when the provided information is combined into new information that is typically more complex than the original information. For example, multi-modal (audio and video) data fusion is considered cooperative.

(2)according to the input/output data types and their nature, as proposed by Dasarathy [4];

(1)data in-data out (DAI-DAO): this type is the most basic or elementary data fusion method that is considered in classification. This type of data fusion process inputs and outputs raw data; the results are typically more reliable or accurate. Data fusion at this level is conducted immediately after the data are gathered from the sensors. The algorithms employed at this level are based on signal and image processing algorithms;(2)data in-feature out (DAI-FEO): at this level, the data fusion process employs raw data from the sources to extract features or characteristics that describe an entity in the environment;(3)feature in-feature out (FEI-FEO): at this level, both the input and output of the data fusion process are features. Thus, the data fusion process addresses a set of features with to improve, refine or obtain new features. This process is also known as feature fusion, symbolic fusion, information fusion or intermediate-level fusion;(4)feature in-decision out (FEI-DEO): this level obtains a set of features as input and provides a set of decisions as output. Most of the classification systems that perform a decision based on a sensor's inputs fall into this category of classification;(5)Decision In-Decision Out (DEI-DEO): This type of classification is also known as decision fusion. It fuses input decisions to obtain better or new decisions.

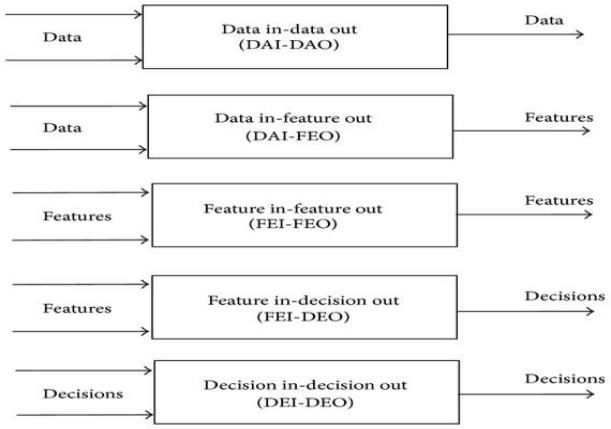
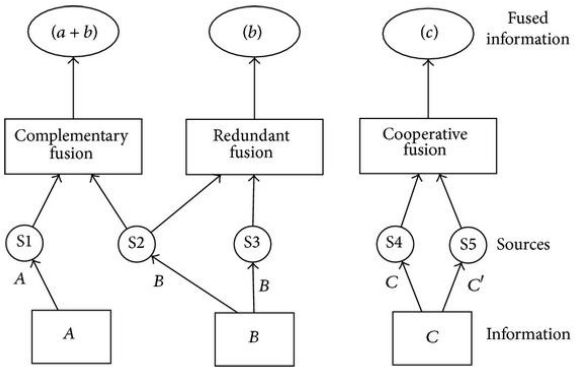


Fig 3 Input data sources, as proposed by Durrant-Whyte

(3)following an abstraction level of the employed data: (a) raw measurement, (b) signals, and (c) characteristics or decisions;

Information fusion typically addresses three levels of abstraction: (1) measurements, (2) characteristics, and (3) decisions. Other possible classifications of data fusion based on the abstraction levels are as follows:(1)low level fusion: the raw data are directly provided as an input to the data fusion process, which provide more accurate data (a lower signal-to-noise ratio) than the individual sources;(2)medium level fusion: characteristics or features (shape, texture, and position) are fused to obtain features that could be employed for other tasks. This level is also known as the feature or characteristic level;(3)high level fusion: this level, which is also known as decision fusion, takes symbolic representations as sources and combines them to obtain a more accurate decision. Bayesian's methods are typically employed at this level;(4)multiple level fusion: this level addresses data provided from different levels of abstraction (i.e., when a measurement is combined with a feature to obtain a decision).



(4)Based on the different data fusion levels defined by the JDL;

(1)level 0—source preprocessing: source preprocessing is the lowest level of the data fusion process, and it includes fusion at the signal and pixel levels. In the case of text sources, this level also includes the information extraction process. This level reduces the amount of data and maintains useful information for the high-level processes;(2)level 1—object refinement: object refinement employs the processed data from the previous level. Common procedures of this level include spatio-temporal alignment, association, correlation, clustering or grouping techniques, state estimation, the removal of false positives, identity fusion, and the combining of features that were extracted from images. The output results of this stage are the object discrimination (classification and identification) and object tracking (state of the object and orientation). This stage transforms the input information into consistent data structures;(3)level 2—situation assessment: this level focuses on a higher level of inference than level 1. Situation assessment aims to identify the likely situations given the observed events and obtained data. It establishes relationships between the objects. Relations (i.e., proximity, communication) are

valued to determine the significance of the entities or objects in a specific environment. The aim of this level includes performing high-level inferences and identifying significant activities and events (patterns in general). The output is a set of high-level inferences;(4)level 3—impact assessment: this level evaluates the impact of the detected activities in level 2 to obtain a proper perspective. The current situation is evaluated, and a future projection is performed to identify possible risks, vulnerabilities, and operational opportunities. This level includes (1) an evaluation of the risk or threat and (2) a prediction of the logical outcome;(5)level 4—process refinement: this level improves the process from level 0 to level 3 and provides resource and sensor management. The aim is to achieve efficient resource management while accounting for task priorities, scheduling, and the control of available resources.

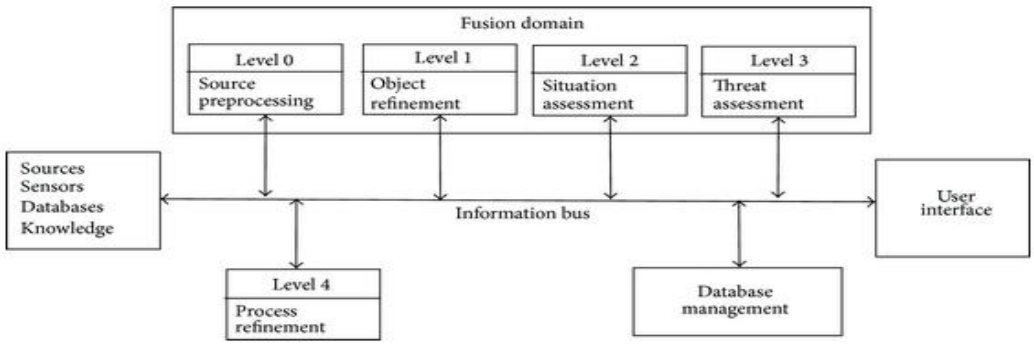


Fig 4:-Data fusion levels defined by the JDL;

(5)Depending on the architecture type: (a) centralized, (b) decentralized, or (c) distributed.

One of the main questions that arise when designing a data fusion system is where the data fusion process will be performed. Based on this criterion, the following types of architectures could be identified:(1)centralized architecture: in a centralized architecture, the fusion node resides in the central processor that receives the information from all of the input sources. Therefore, all of the fusion processes are executed in a central processor that uses the provided raw measurements from the sources. In this schema, the sources obtain only the observationas measurements and transmit them to a central processor, where the data fusion process is performed. If we assume that data alignment and data association are performed correctly and that the required time to transfer the data is not significant, then the centralized scheme is theoretically optimal. However, the previous assumptions typically do not hold for real systems. Moreover, the large amount of bandwidth that is required to send raw data through the network is another disadvantage for the centralized approach. This issue becomes a bottleneck

when this type of architecture is employed for fusing data in visual sensor networks. Finally, the time delays when transferring the information between the different sources are variable and affect the results in the centralized scheme to a greater degree than in other schemes;(2)decentralized architecture: a decentralized architecture is composed of a network of nodes in which each node has its own processing capabilities and there is no single point of data fusion. Therefore, each node fuses its local information with the information that is received from its peers. Data fusion is performed autonomously, with each node accounting for its local information and the information received from its peers. Decentralized data fusion algorithms typically communicate information using the Fisher and Shannon measurements instead of the object's state [7]. The main disadvantage of this architecture is the communication cost, which is at each communication step, where is the number of nodes; additionally, the extreme case is considered, in which each node communicates with all of its peers. Thus, this type of architecture could suffer from scalability problems when the number of nodes is increased;(3)distributed architecture: in a distributed architecture, measurements from each source node are processed independently before the information is sent to the fusion node; the fusion node accounts for the information that is received from the other nodes. In other words, the data association and state estimation are performed in the source node before the information is communicated to the fusion node. Therefore, each node provides an estimation of the object state based on only their local views, and this information is the input to the fusion process, which provides a fused global view. This type of architecture provides different options and variations that range from only one fusion node to several intermediate fusion nodes;(4)hierarchical architecture: other architectures comprise a combination of decentralized and distributed nodes, generating hierarchical schemes in which the data fusion process is performed at different levels in the hierarchy.

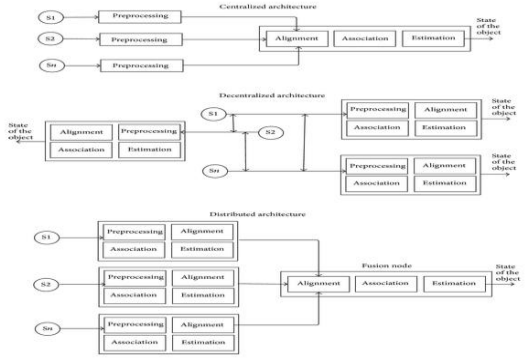


Fig 5 Big Data Integration

3.2) BDI: Schema Mapping:-

Schema mapping in a data integration system refers to (i) creating a mediated (global) schema, and (ii) identifying the mappings between the mediated (global) schema and the local schemas of the data sources to determine which (sets of) attributes contain the same information [1]. Early efforts in integrating a large number of sources involved integrating data from the Deep Web. Two types of solutions were proposed. The first is to build mappings between Web forms (interfaces to query the Deep Web) as a means to answer a Web query over all Deep Web sources [5]. The second is to crawl and index the Deep Web data [4], [5]. More recent efforts include extracting and integrating structured data from Web tables and Web lists. The number of sources also increases the variety of the data. Traditional data integration systems require a significant schema mapping effort before the system can be used, so is obviously infeasible when the heterogeneity is at the BDI scale. The basic idea of dataspace systems is to provide best-effort services such as simple keyword search over the available data sources at the beginning, and gradually evolve schema mappings A related notion becoming popular in the Hadoop community is "schema on read" which, in contrast to the traditional approach of defining the schema before loading data (i.e.,schema on write), gives one the freedom to define the schema after the data has been stored.

4 -Talend Open Studio for Data Integration in JAVA:-

Big data integration is a key operational challenge for today's enterprise IT departments. IT groups may find their skill sets, workload, and budgets over-stretched by the need to manage terabytes or petabytes of data in a way that delivers genuine value to business users. Talend, the leading provider of open source data management solutions, helps organizations large and small meet the big data challenge by making big data integration easy, fast, and affordable.

Fig 6 - Talend Open Studio for Data Quality

Talend Open Studio for Big Data greatly simplifies the process of working with Hadoop, Apache's open source MapReduce implementation that's rapidly become the leading framework for computational processing of massive data sets. Hadoop and associated Hadoop applications like HDFS and Hadoop Hive have delivered tremendous value in some of the world's most demanding big data environments, but at the same time are complicated to use and require new skill sets that many IT shops currently lack.

[9]With Talend's open source big data integration software, you can move data into HDFS or Hive, perform operations on it, and extract it without having to do any coding. In the Eclipse-based Talend GUI, you simply drag, drop, and configure graphical components representing a wide array of Hadoop-related operations, and the Talend software automatically generates the corresponding code (including Hadoop Pig Latin code for transforming data stored in HDFS). This easily created code can then be deployed as a stand-alone job, an executable, or a big data integration service.

Talend Open Studio for Big Data also enables you to incorporate big data services into your overall data management services architecture. The Talend graphical workspace includes hundreds of components that make it easy to move data between a Hadoop environment and any major database or file format.

Big Data Integration Made Fast

With Talend, big data integration jobs can be built in minutes rather than days or weeks. And with its metadata repository that facilitates easy reuse of jobs and job parts, Talend Open Studio for Big Data saves you more time the more you use it. As an open source solution freely available for download, Talend also helps you to start resolving those big data integration challenges now rather than later.

Big Data Integration Made Affordable

Talend is the only pure open source solution to today's big data integration challenges. Talend's fully functional open source software is free to download and to use for as long as you want.For enterprises seeking even more big data management functionality, Talend also offers the subscription-based Talend Enterprise Data Integration suite. Costing far less than competing commercial products, Talend Enterprise Data Integration extends Talend Open Studio for Big Data with enterprise-grade features like multi-job load balancing, application failover, and SOA enablement, as well as first-rate technical support.

Other familiar open source available for big data analytics is Apache™ Hadoop®[8] it is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance. Rather than relying on high-end hardware, the resiliency of these clusters comes from the software's ability to detect and handle failures at the application layer.

5-CONCLUSION

This article concludes that Big data usually includes data sets with sizes beyond the ability of commonly used software tools to capture, curate, manage, and process data within a tolerable elapsed time. Big data comes with several challenges like Fusion, Integration, Legality, security and analytics so on. And two challenges data fusion and schema mapping are focused here. Also some open source has been developing for big data analytics. It is hoped that future trends in analyzing the big data will be great challenging.

REFERENCES:-

1. Z. Bellahsene, A. Bonifati, and E. Rahm, editors. Schema Matching and Mapping. Springer, 2011.

2. J. Bleiholder and F. Naumann. Data fusion. ACM Computing Surveys, 41(1):1–41, 2008.

3. A Review of Data Fusion Techniques by proposed by Durrant-Whyte National Center for Biotechnology Information. 2013.

4. N. N. Dalvi, A. Machanavajjhala, and B. Pang. An analysis of structured data on the web. PVLDB, 5(7):680–691, 2012.

5. X. Dong, A. Y. Halevy, and C. Yu. Data integration with uncertainties. In VLDB, 2007.

6. X. L. Dong, L. Berti-Equille, and D. Srivastava. Integrating conflicting data: the role of source dependence. PVLDB, 2(1), 2009.

7. X. L. Dong and F. Naumann. Data fusion–resolving data conflicts for integration. PVLDB, 2009.

8. http://en.wikipedia.org/wiki/Apache_Hadoop.

9. https://www.talend.com/products/talend-open-studio.