

ABSTRACT

SYSTEM AND METHOD FOR VARYING MEMORY SIZE IN A DATA STREAM PROCESSING

A system and method for systematically varying the memory size in data stream processing to improve the connection degree sketch is provided. The system and method receives input data such as IP address or TV program identifiers. The length of the address is encoded in order to increase the memory size of the input data using suitable error correction technique. Connection degree sketch is built using bit arrays and hash functions based on Chinese-Remainder Theorem (CRT). Reverse sketching technique is applied using Chinese-Remainder Theorem (CRT) on the encoded data. Appropriate decoding technique is selected in order to decode the encoded data such that the decoded data is of the length of the original input data. The system provides better performance by reducing false positives and false negatives.

[To be published with Figure 4]

I/WE CLAIM:

1. A method for varying a memory size in a data stream processing while improving a connection degree sketch, the method comprising:

encoding an input data by using an error correction technique to produce an encoded data, the encoded data further results in a modified memory size produced by changing the memory size of the input data ;

developing a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data in order to obtain an estimated encoded data; and

decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data;

wherein the encoding, the developing and the decoding are performed by a processor.

2. The method of claim 1, wherein the input data comprises IP address, TV programs, tuples such as HouseID, ChannelID
3. The method of claim 1, wherein error coding technique comprises Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.
4. The method of claim 1, wherein the modified memory size of the encoded data is always greater than the memory size of the input data.
5. The method of claim 1, wherein the set of parameters further comprises of number-of-rows and number-of-hash functions.
6. The method of claim 1, wherein the decoding is applied to detect and correct one or more anomalies in the data stream.

7. A system for varying a memory size in a data stream processing while improving a connection degree sketch, the system comprising:

a processor;

a memory coupled to the processor, wherein the processor is capable of executing a plurality of modules stored in the memory, and wherein the plurality of module comprising:

an encoding module configured to encode an input data by using an error correction technique to produce an encoded data, the encoded data further results in a modified memory size produced by changing the memory size of the input data ;

a connectivity development module configured to develop a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data in order to obtain an estimated encoded data; and

a decoding module configured to decode the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data.

8. The system of claim 7, wherein the input data comprises IP address, TV programs represented by appropriate tuples such as <HouseID, ChannelID>.

9. The system of claim 7, wherein error coding technique comprises Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.

10. The system of claim 7, wherein the encoding module ensures the modified memory size of the encoded data is always greater than the memory size of the input data.

11. The system of claim 7, wherein the decoding module is further configured to detect and correct one or more anomalies in the data stream.

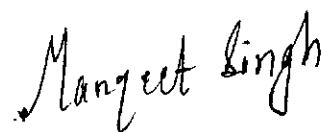
12. A computer program product having embodied thereon a computer program for varying a memory size in a data stream processing while improving a connection degree sketch, the computer program product comprising:

a program code for encoding an input data by using an error correction technique to produce an encoded data, the encoded data further comprises of a modified memory size produced by modifying the memory size of the input data ;

a program code for developing a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data, such that the modified memory size of the encoded data further modifies a size of the host connectivity; and

a program code for decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data.

Dated this 20th Day of May, 2013



Manjeet Singh

Agent for Applicant

IN-PA-1748

FIELD OF THE INVENTION

0001 The present invention relates to a field of data stream processing. More particularly, the present invention relates to varying a memory size in a data stream processing while improving a connection degree sketch.

BACKGROUND OF THE INVENTION

0002 Estimating the connection degree of hosts in a high-speed network at a given time interval is a main problem in Network-anomaly detection. Network traffic anomalies such as failures and attacks are common in today's network. Identifying network traffic anomalies rapidly and accurately is critical for large network operators. Estimation of the connection degree identifies hosts that are responsible for anomalies like DDOS (Distributed Denial-Of-Service), flash crowds and network failures.

0003 In the current scenario, depending on traffic volume and link speeds, a suitable detection method has to be selected to identify network anomalies. Recent research efforts have been directed towards developing scalable heavy hitter detection techniques for accounting and anomaly detection purposes. Heavy hitter techniques do not correspond to flows, experiencing significant changes in the network traffic.

0004 Researchers have developed various systems like Snort, Flowscan etc. Snort is network intrusion prevention and detection system which utilizes rule based language combining signature, protocol and anomaly inspection methods.

0005 Flowscan analyzes and reports on flow data exported by Internet Protocol routers. Flowscan groups PERL scripts and modules such as flow collection engine,

20 MAY 2013

a high performance database, and a visualization tool. After all the tools are assembled, the Flowscan system produces graphic images, appropriate for use in web pages. Flowscan provides a continuous, near real-time view of the network traffic through a network's border. However, these systems are not suitable because of the massive amount of network traffic in high-speed links.

0006 Further, data monitoring algorithms based on efficient data structures have been in use for high traffic user detection and traffic-volume queries. The data monitoring algorithms allow monitoring of data network traffic without tracking data individually for each separate key. The data monitoring algorithm utilizes parallel hash tables to identify large flows using a memory that is only a small constant and larger than the number of large flows. However, this technique only detects high traffic users, and does not detect users having significant changes in traffic.

0007 Another approach in identifying network anomalies is by using data streaming techniques. The data streaming techniques includes method for locating hosts with large connection degree based on the reversible connection degree sketch in order to monitor anomalous network traffics. The technique presented uses bit array and hash functions based on Chinese-Remainder Theorem (CRT) to create a connection degree sketch of network data stream. The problem associated with the technique is the presence of false positives or false negatives in the final result. The false positives or false negatives in the final result reduce the quality of the result. The existing method facilitates in retrieving the host address of the large connection degree hosts from the connection degree sketch. However, bit flips occurs while retrieving host address and occurrence of bit flip results to wrong addresses and increases the number of false positives and false negatives in the final result.

SUMMARY OF THE INVENTION

0008 This summary is provided to introduce concepts related to system and method for varying a memory size in data stream processing to improve connection degree sketch and the concepts are further described below in the detailed description. This summary is not intended to identify essential features of the claimed subject matter nor is it intended for use in determining or limiting the scope of the claimed subject matter.

0009 In one embodiment, a system for varying memory size in a data stream processing to improve connection degree is provided. The system comprises a processor, a memory coupled to the processor. The processor is capable of executing plurality of modules stored in the memory. The plurality of modules comprises an encoding module configured to encode an input data by using an error correction coding technique to produce an encoded data. The encoded data results in a modified memory size produced by changing the memory size of the input data. The plurality of modules comprises a connectivity development module configured to develop host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data. Upon applying the reverse sketching the estimated encoded data is obtained. The plurality of modules comprises a decoding module configured to decode the encoded data after the host connectivity is developed by using a decoding technique and to obtain an output data, such that a memory size of the output data is similar to the memory size of the input data.

00010 In one embodiment, a method for varying memory size in data stream processing to improve connection degree sketch is provided. The method comprises encoding an input data by using an error coding technique to produce an encoded data. The encoded data further comprises of a modified memory size produced by modifying the memory size of the input data. The method comprises developing host

connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data such that the host connectivity is expressed in terms of encoded data. The method comprises decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data. The method of encoding, the developing and the decoding are performed by a processor.

00011 The invention further comprises of a computer program product having embodied thereon a computer program for varying a memory size in a data stream processing while improving a connection degree sketch. The computer program product comprises of a program code for encoding an input data by using an error correction technique to produce an encoded data. The encoded data further comprises of a modified memory size produced by modifying the memory size of the input data. The computer program product further comprises of a program code for developing a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data, such that the modified memory size of the encoded data further modifies a size of the host connectivity and a program code for decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data.

BRIEF DESCRIPTION OF THE DRAWINGS

00012 The detailed description is described with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The same numbers are used throughout the drawings to refer like features and components.

00013 Figure 1 illustrates a network implementation of a system for varying memory size in data stream processing to improve connection degree sketch is shown, in accordance with an embodiment of the present subject matter.

00014 Figure 2 illustrates the system, in accordance with an embodiment of the present subject matter.

00015 Figure 3 illustrates a method for varying memory size in data stream processing to improve connection degree sketch, in accordance with an embodiment of the present subject matter.

00016 Figure 4 illustrates method of implementing error control technique with reverse sketching technique, in accordance with an embodiment of the present subject matter.

DETAILED DESCRIPTION OF THE INVENTION

00017 Systems and methods for varying memory size in data stream processing to improve connection degree sketch are described. The system and method receives input data such as IP address or TV program such as tuples like <HouseID, ChannelID>. The length of the address is encoded in order to increase memory size of the input data using error control technique. The error control techniques may comprise Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.

00018 Connection degree sketch of the encoded input data is built using the bit arrays of appropriate dimensions and hash functions based on Chinese-Remainder Theorem (CRT). For each encoded input data, several bits are set in the connection degree by group of hash functions. By using the reversible property of the

connection degree sketch, in-degree and out-degree associated with each input is estimated appropriately.

00019 Appropriate decoding technique is selected in order to decode the encoded data such that the decoded data is of the length of the original input data. The estimation of connection degree using error control technique and applying reverse connection degree sketch may be extended to TV programs such that the false positives and false negatives present in the final result can be reduced.

00020 While aspects of described system and method for varying memory size in data stream processing to improve connection degree sketch may be implemented in any number of different computing systems, environments, and/or configurations, the embodiments are described in the context of the following exemplary system.

00021 Referring now to Figure 1, a network implementation 100 of a for varying memory size in data stream processing to improve connection degree sketch is illustrated, in accordance with an embodiment of the present subject matter. In one embodiment, the system receives input data such as IP address or TV program. Consider the input data is of the length k bit. The system encodes the input data using error control technique. The input data of k bit length is increased n bit length upon encoding the data such that n bits are greater than k bits. The encoded n bits are sent to build a connection degree sketch.

00022 The system facilitates in building connection degree sketch using bit arrays and hash functions based on Chinese-Remainder Theorem (CRT). The connection degree sketch builds a compact summary of host connection degrees efficiently and accurately. Reverse sketching technique is applied using Chinese-Remainder Theorem (CRT) on the encoded data. The encoded data is decoded using decoding technique. Upon decoding, the output of the encoded data is received as k bits length similar to the input data.

00023 Although the present subject matter is explained by considering a scenario that the system 102 is implemented as an application on a server. It may be understood that the system 102 may also be implemented in a variety of computing systems, such as a laptop computer, a desktop computer, a notebook, a workstation, a mainframe computer, a server, a network server, and the like. It will be understood that the system 102 may be accessed by multiple users through one or more user devices 104-1, 104-2...104-N, collectively referred to as user 104 hereinafter, or applications residing on the user devices 104. Examples of the user devices 104 may include, but are not limited to, a portable computer, a personal digital assistant, a handheld device, and a workstation. The user devices 104 are communicatively coupled to the system 102 through a network 106.

00024 In one implementation, the network 106 may be a wireless network, a wired network or a combination thereof. The network 106 can be implemented as one of the different types of networks, such as intranet, local area network (LAN), wide area network (WAN), the internet, and the like. The network 106 may either be a dedicated network or a shared network. The shared network represents an association of the different types of networks that use a variety of protocols, for example, Hypertext Transfer Protocol (HTTP), Transmission Control Protocol/Internet Protocol (TCP/IP), Wireless Application Protocol (WAP), and the like, to communicate with one another. Further the network 106 may include a variety of network devices, including routers, bridges, servers, computing devices, storage devices, and the like.

00025 Referring now to Figure 2, the system 102 is illustrated in accordance with an embodiment of the present subject matter. In one embodiment, the system 102 may include at least one processor 202, an input/output (I/O) interface 204, and a memory 206. Then at least one processor 202 may be implemented as one or more microprocessors, microcomputers, microcontrollers, digital signal processors, central

processing units, state machines, logic circuitries, and/or any devices that manipulate signals based on operational instructions. Among other capabilities, the at least one processor 202 is configured to fetch and execute computer-readable instructions stored in the memory 206.

00026 The I/O interface 204 may include a variety of software and hardware interfaces, for example, a web interface, a graphical user interface, and the like. The I/O interface 204 may allow the system 102 to interact with a user directly or through the client devices 104. Further, the I/O interface 204 may enable the system 102 to communicate with other computing devices, such as web servers and external data servers (not shown). The I/O interface 204 can facilitate multiple communications within a wide variety of networks and protocol types, including wired networks, for example, LAN, cable, etc., and wireless networks, such as WLAN, cellular, or satellite. The I/O interface 204 may include one or more ports for connecting a number of devices to one another or to another server.

00027 The memory 206 may include any computer-readable medium known in the art including, for example, volatile memory, such as static random access memory (SRAM) and dynamic random access memory (DRAM), and/or non-volatile memory, such as read only memory (ROM), erasable programmable ROM, flash memories, hard disks, optical disks, and magnetic tapes. The memory 206 may include modules 208 and data 210.

00028 The modules 208 include routines, programs, objects, components, data structures, etc., which perform particular tasks or implement particular abstract data types. In one implementation, the modules 208 may include an encoding module, a connectivity development module 212, a decoding module, and other modules 216. The other modules 216 may include programs or coded instructions that supplement applications and functions of the system 102.

00029 The data 230, amongst other things, serves as a repository for storing data processed, received, and generated by one or more of the modules 208. The data 230 may also include a system database 232 and other data 234. The other data 234 may include data generated as a result of the execution of one or more modules in the other module 216.

00030 In one embodiment of the invention, referring to figure 2 and figure 3, the system 102 comprises the encoding module 210 configured to encode an input data (step 302) by using an error coding technique to produce an encoded data. The input data may comprise IP address, TV programs. The error control coding techniques may comprise *Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.*

00031 The encoding module 210 takes k bit input and gives n bit encoded output by adding $(n - k)$ redundant bits. There are many possibilities to choose the code.

00032 In one embodiment one can use BCH code with appropriate rate. For example, if we

00033 Chose rate of $1/2$, 32-bit input is changed to 64-bit encoded block resulting in a memory expansion by a factor of 2.

00034 In another embodiment, one can also use Reed-Solomon codes, where k symbols are encoded into n symbols where each symbol corresponds to a group of bits, for example, a symbol can be a group of 8 bits.

00035 In the following paragraphs, we are considering the case of applying BCH codes:

00036 In one embodiment of the invention, the system 102 comprises the connectivity development module 212 configured to develop host connectivity (step 304) by using a set of parameters and applying a reverse sketching technique over the encoded data. The set of parameters includes number-of-rows, number-of-hash functions. Further, reverse sketching technique is applied using Chinese-Remainder Theorem.

00037 The reversible sketching technique develops a compact summary of host connection degrees efficiently and accurately. For each packet received as input, reversible sketching technique requires set of several bits selected in a bit array by a group of hash functions. The hash functions are designed by the connectivity development module 212 based on the Chinese Remainder Theorem such that the in-degree or out-degree associated with a given host may be accurately estimated. By applying reverse sketch method abnormal hosts are located. The host addresses are reconstructed in association with large connection degrees or significant changes in connection degrees by a simple calculation purely based on the characteristics of the hash functions. In one embodiment, the modified memory size of the encoded data further modifies size of the host connectivity.

00038 In one embodiment of the invention, the system 102 comprises the decoding module 214 configured to decode the encoded data (step 306) after the host connectivity is developed by using a decoding technique to obtain an output data, such that the memory size of the output data is similar to the memory size of the input data. The decoding procedure generally comprises of the following steps:

1. Calculate the syndromes for the received vector
2. Determine the number of errors and the error locator polynomial from the syndromes
3. Calculate the roots of the error location polynomial to find the error locations
4. Calculate the error values at those error locations

5. Correct the errors

00039 Apart from the above mentioned decoding scheme, many variants and implementations are also available. In one embodiment, the decoding module 214 applies decoding technique to the n -bit encoded data to detect and correct one or more anomalies in the data stream. The decoding module 214 generates a k -bit output that is identical to the original input data. Depending on the coding scheme, different types and numbers of errors may be corrected in the encoded data.

00040 The decoding technique depends on error correction code chosen. For BCH codes the decoding technique discussed above is used. If LDPC is selected for encoding then decoding may be carried out by using different flavors of message passing algorithm.

00041 Figure 4 illustrates method of implementing error control technique with reverse sketching technique according to one embodiment of the invention. The system 102 receives input data stream of length k bits (step 402). The encoding module 210 encodes the input data of k bits length to n bits length (step 404), wherein the length of n bits is greater than k bits. The encoded data is sent to build connection degree sketch (step 406) using bit arrays and hash functions in order to find the anomalies in the input data. The reverse sketching technique is applied (step 408) using Chinese remainder Theorem. The encoded data upon applying reverse sketching technique reduces the false positive and the false negatives present in the output. Encoded data of n bits is decoded (step 410) using decoding module 214 such that the decoded data is of k bits length.

00042 In one exemplary embodiment, varying memory size in data stream processing to improve connection degree sketch is explained. For example, consider input data stream having 200 IP addresses wherein each input data is of 32 bit in

length. The sample results obtained by building the connection degree sketch on the data and applying reverse sketch subsequently is captured in Table 1.

Input	Output	True positives count	False positives count
0.1.4.4	0.1.4.4	1	4
0.2.4.3	0.2.4.3		
4.6.4.3	4.6.4.3		
1.3.7.16	0.6.6.4		
2.1.4.17	1.1.7.24		
1.2.7.11			
0.3.12.25			
1.3.4.14			
2.0.6.9			
2.1.2.24			
2.2.11.5			
0.3.4.7			
1.1.3.14			
2.2.7.23			
2.2.10.10			
2.1.4.28			
1.0.15.14			
1.1.7.24			
2.2.12.17			
2.2.2.5			
2.1.0.15			

1.0.9.16			
1.1.0.16			
2.2.11.18			
0.1.3.14			
0.2.13.12			
2.0.5.12			
2.0.14.21			
0.1.15.4			
2.3.15.21			
1.0.11.18			
1.1.4.3			
0.2.12.11			
1.3.0.20			
0.2.2.26			
1.3.12.11			
2.1.4.28			
0.1.9.14			
0.3.6.28			
1.2.12.11			

Table 1

00043 Table 1 shows the final result of the data stream processing without using error control technique. The final result had 5 IP addresses as the host with large connection degree, wherein one host 0.1.4.4 is the only true positive and other four hosts are false positives.

00044 For the above example, error control technique is implemented with reverse sketching technique. BCH encoding technique is applied to the IP addresses

received as input data wherein code rate is 50 percent or 0.5. Consider the input data is of k bit length in the high speed data stream. The k bit length of the input data is increased to n bit length after applying error coding technique. Consider code rate k/n as 0.5. The encoding module encodes and increases IP addresses length from k of 32 bits to n of 64 bit. The IP addresses after applying reverse sketching technique is sent to the decoding module 214. The decoding module 214 decodes the encoded data using decoding technique wherein the output of the decoded data is of the length k bits i.e., 32 bits.

Input	Output	True positives count	False positives Count
0.1.4.4	0.1.4.4	1	1
0.2.4.3	4.6.4.3		
4.6.4.3			
1.3.7.16			
2.1.4.17			
1.2.7.11			
0.3.12.25			
1.3.4.14			
2.0.6.9			
2.1.2.24			
2.2.11.5			
0.3.4.7			
1.1.3.14			
2.2.7.23			
2.2.10.10			
2.1.4.28			

1.0.15.14			
1.1.7.24			
2.2.12.17			
2.2.2.5			
2.1.0.15			
1.0.9.16			
1.1.0.16			
2.2.11.18			
0.1.3.14			
0.2.13.12			
2.0.5.12			
2.0.14.21			
0.1.15.4			
2.3.15.21			
1.0.11.18			
1.1.4.3			
0.2.12.11			
1.3.0.20			
0.2.2.26			
1.3.12.11			
2.1.4.28			
0.1.9.14			
0.3.6.28			
1.2.12.11			

Table 2

00045 Using error control technique with reverse sketching technique, the amount of false positives for 200 IP addresses identified are 2 as shown in Table 2, wherein

the output consists of one true positive and one false positive, thus showing an improvement with error correction coding.

ADVANTAGES

1. Systematic way of increasing the memory in data stream processing using error correction techniques.
2. Reduction in the amount of false positives/negatives in the final result.

I/WE CLAIM:

1. A method for varying a memory size in a data stream processing while improving a connection degree sketch, the method comprising:

encoding an input data by using an error correction technique to produce an encoded data, the encoded data further results in a modified memory size produced by changing the memory size of the input data ;

developing a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data in order to obtain an estimated encoded data; and

decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data;

wherein the encoding, the developing and the decoding are performed by a processor.

2. The method of claim 1, wherein the input data comprises IP address, TV programs, tuples such as HouseID, ChannelID
3. The method of claim 1, wherein error coding technique comprises Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.
4. The method of claim 1, wherein the modified memory size of the encoded data is always greater than the memory size of the input data.
5. The method of claim 1, wherein the set of parameters further comprises of number-of-rows and number-of-hash functions.
6. The method of claim 1, wherein the decoding is applied to detect and correct one or more anomalies in the data stream.

7. A system for varying a memory size in a data stream processing while improving a connection degree sketch, the system comprising:

a processor;

a memory coupled to the processor, wherein the processor is capable of executing a plurality of modules stored in the memory, and wherein the plurality of module comprising:

an encoding module configured to encode an input data by using an error correction technique to produce an encoded data, the encoded data further results in a modified memory size produced by changing the memory size of the input data ;

a connectivity development module configured to develop a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data in order to obtain an estimated encoded data; and

a decoding module configured to decode the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data.

8. The system of claim 7, wherein the input data comprises IP address, TV programs represented by appropriate tuples such as <HouseID, ChannelID>.

9. The system of claim 7, wherein error coding technique comprises Hamming, Bose, Chaudhuri and Hocquenghem (BCH), Reed-Solomon, Low Density Parity-Check (LDPC), Reed-Muller, Convolutional codes.

10. The system of claim 7, wherein the encoding module ensures the modified memory size of the encoded data is always greater than the memory size of the input data.

11. The system of claim 7, wherein the decoding module is further configured to detect and correct one or more anomalies in the data stream.

12. A computer program product having embodied thereon a computer program for varying a memory size in a data stream processing while improving a connection degree sketch, the computer program product comprising:

a program code for encoding an input data by using an error correction technique to produce an encoded data, the encoded data further comprises of a modified memory size produced by modifying the memory size of the input data ;

a program code for developing a host connectivity by using a set of parameters and applying a reverse sketching technique over the encoded data, such that the modified memory size of the encoded data further modifies a size of the host connectivity; and

a program code for decoding the encoded data after the host connectivity is developed by using a decoding technique and obtaining an output data, such that a memory size of the output data is similar to the memory size of the input data.

Dated this 20th Day of May, 2013



Manjeet Singh

Agent for Applicant

IN-PA-1748