Semester : VII Subject : Big Data Analytics Academic Year: 2024 – 2025

MODULE 4: Mining Data Stream

Mining data stream is a process to extract knowledge in real time from a large amount of volatile data, which comes in an infinite stream. The data is volatile because it is continuously changing and evolving over time. The system does not store the data in the database due to limited amount of resources.

Model For Data Stream Processing

An infinite amount of data arrives continuously in a data stream. Assume *D* is the data stream which is the sequence of transactions and can be defined as:

$$D = (T_1, T_2, \ldots., T_i, T_{i+1}, \ldots., T_j)$$

Where: T_1: is the 1st transaction, T_2: is the 2nd transaction, T_i: is the ith transaction and T_j: jth transaction.

There are three different models for data stream processing, namely, Landmark, Sliding Windows and Damped, as shown in Figure 1 and discussed as follows:

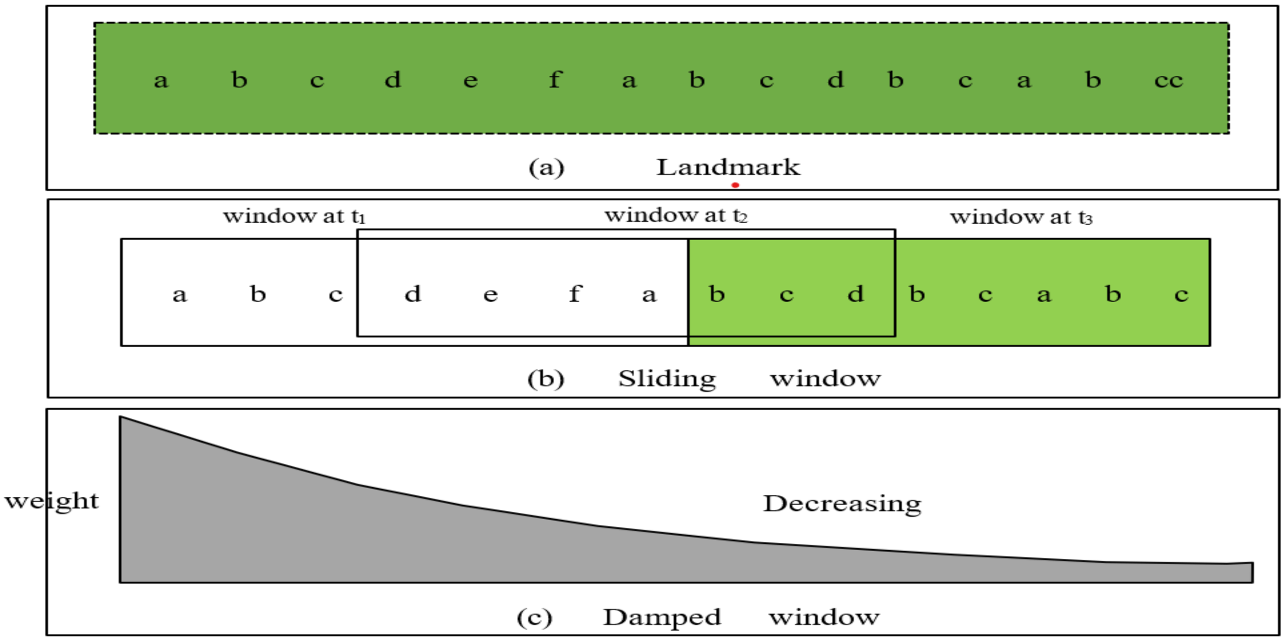


Figure1: Model for data stream processing

Landmark model:

This model finds the frequently used items in entire data stream from a specific time (known as landmark) till present. In other words, the model finds out frequent items starting from T_i to current time T_t from the window *W[i,t]*, where *i* represents the landmark time. However, if *i*=1, then the model finds out the frequent items over entire data stream. In this type of model, all time-points are treated equally after the starting time. find items in the most recent data streams. The examples of landmark model include stock monitor system, which observes and reports on global stock market.

Sliding Windows model:

This model stores recent data in sliding window from a certain range and discard old data items. The size of the sliding window may vary according to the type of application used. Suppose the size of the sliding window is *w* and current time is *t*, the model finds the data on the sliding window- *W[t-w+1, t]*. The window will update its size according to the current time. The model does not store the data that arrive before the time *t-w+1*. In other words, the part of the data stream that is in the range of the sliding window are retrieved at a particular time point.

Damped model:

This model is also called as Time-Fading model as it assigns more weight to the recent transactions in data stream and this weight keeps on decreasing with age. In other words, the older transactions have less weight as compared to the newer transactions in the data stream. This model is mostly used in those applications where new data has more impact on mining results in comparison to the old data and the impact of old data decreases with time.

Data Stream Management System

Data Stream Management System (DSMS) extracts knowledge from multiple data streams by eliminating undesirable elements, as shown in Figure 2. DSMS is important where the input data rate is controlled externally. For example, Google queries.

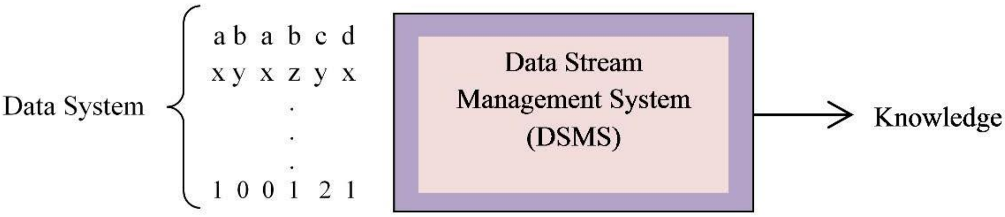
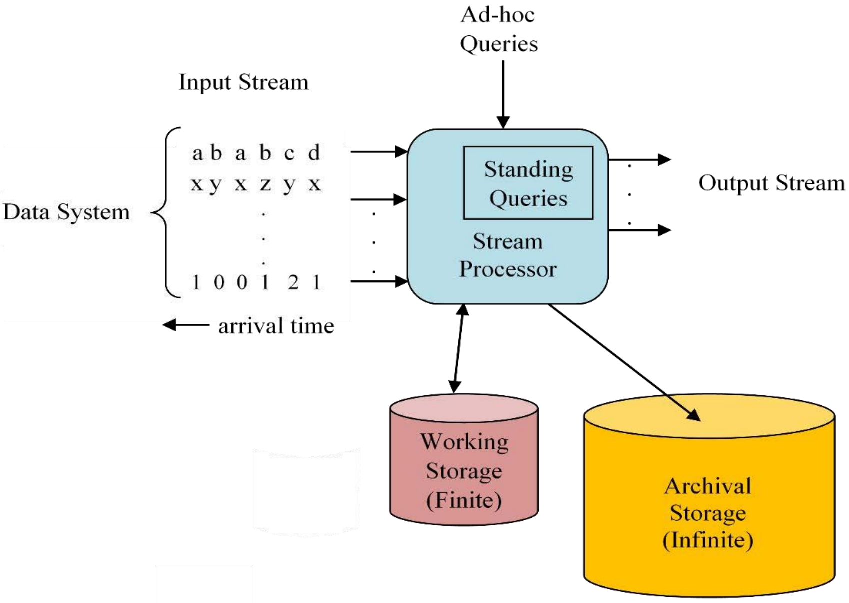


Figure 2: A simple outline of Data Stream Management System

The sources of data streams include Internet traffic, online transactions, satellite data, sensors data, live event data, real-time surveillance systems, etc. Figure 3 shows the detailed view of a data stream management system. The components of this system are described as follows:



Processor: The processor is a software that executes the queries on the data stream. There can be multiple processors working together. The processor may store some standing queries and also allows ad-hoc queries to be issued by the users.

Streams Entering: There can be several streams entering in the system. Conventionally, we will assume that the elements at the right end of the stream have arrived more recently. The time goes backward to the left i.e. the further left the earlier the element entered the system.

Output: The system makes output in response to the standing queries and the ad-hoc queries.

Archival Storage: There is a massive archival storage and we cannot assume the archival storage is architected like a database system. Further, we can use appropriate indices or other tools to efficiently answer the queries from that data. We only know that if we had to reconstruct the history of the streams it could take a long time.

Limited Working Storage: It might be a main memory, or a flash storage, or even magnetic disk. But we assume that it holds important parts of the input streams in a way that supports fast execution of query

Queries Of Data Stream

Streams can be carried in two modes:

Ad-hoc queries:

This is similar to the way we query a database system in which we make a query once and expect an answer to the query based on the current state of the system. *For example,* what is the maximum value seen so far in data stream, *D*, from its beginning to the exact time the query is asked?

This question can be answered by keeping a single value- the maximum- and updating it (if necessary), every time a new stream element arrives.

Standing queries:

In this type of query, the users write the query once. However as compared to ad-hoc queries, the difference is that here, the users expect the system to report the answer available at all times perhaps outputting a new value each time the answer changes.

For example, report each new maximum value ever seen in data stream, *D*. This question can be answered by keeping one value- the maximum (MAX)- and each new element is compared with the MAX. If it is larger, then we output the value and update the MAX to be that value.

Issues And Challenges Of Data Stream

Input tuples: The input elements are the tuples of a very simple kind such as bits or integers. There are one or more input ports at which data arrives. The arrival rates of input tuples are very high on these input ports.

Arrival rate: The arrival rate of data is fast enough that it is not feasible for the system to store all the arriving data and at the same time make it instantaneously available for any query that we might want to perform on the data.

Critical calculations: The algorithms for data stream are general methods that use a limited amount of storage (perhaps only main memory) and still enables to answer important queries about the content of the stream. However, it becomes difficult to perform critical calculations about the data stream with such a limited amount of memory.