IRSTI 20.23.17

Modeling the processing of a large amount of data

Balakayeva G.T., Al-Farabi Kazakh National University, Almaty, Kazakhstan,
+77013201802, E-mail: gulnardtsa@gmail.com,
Darkenbayev D.K., Al-Farabi Kazakh National University, Almaty, Kazakhstan,
+77012591891, E-mail: dauren.kadyrovich@gmail.com

The definition of large amounts of data, Big Data, is used to refer to technologies such as storing and analyzing a significant amount of data that requires high speed and real-time decision-making when processing. Typically, when serious analysis is said, especially if the term DataMining is used, hat there is a huge amount of data. There are no universal methods of analysis or algorithms suitable for any cases and any volumes of information. Data analysis methods differ significantly in performance, quality of results, usability and data requirements. Optimization can be carried out at various levels: equipment, databases, analytical platform, preparation of source data, specialized algorithms. Big data is a set of technologies that are designed to perform three operations. First, to process large amounts of data compared to "standard" scenarios. Secondly, be able to work with fast incoming data in very large volumes. That is, the data is not just a lot, but they are constantly becoming more and more. Thirdly, they must be able to work with structured and poorly structured data in parallel in different aspects. Large data suggest that the input algorithms receive a stream of not always structured information and that more can be extracted from it than any one idea. The results of the study are used by the authors in modeling large data and developing a web application.
Key words: Large amounts of data, data processing, analysis, modeling, methods.

Үлкен өлшемді деректерді өңдеуді модельдеу

Балақаева Г.Т., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан,
+77013201802, email: gulnardtsa@gmail.com,
Даркенбаев Д.Қ., әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан,
+77012591891, email: dauren.kadyrovich@gmail.com

BigData деректерінің үлкен көлемін анықтау жоғары жылдамдықты және нақты уақыттық шешімдер қабылдауды талап ететін деректердің үлкен көлемін сақтау және талдау сияқты технологияларды қолдану үшін пайдаланылады. Әдетте, егер аналитикалық өңдеудің маңыздылығы туралы айтқанда, әсіресе, DataMining термині пайдаланылған болса, ол үлкен деректердің бар екендігін білдіреді. Кез келген жағдайларда және кез келген ақпарат көлеміне сәйкес келетін әмбебап анализ немесе алгоритмдер жоқ. Деректерді талдау әдістері өнімділік, нәтиже сапасы, қол жетімділік және деректер талаптарына айтарлықтай ерекшеленеді. Оңтайландыру түрлі деңгейлерде жүргізілуі мүмкін: жабдықтар, дерекқорлар, аналитикалық платформалар, бастапқы деректерді дайындау, арнайы алгоритмдер. Үлкен деректер – үш операцияны орындауға арналған технологиялар жиынтығы. Біріншіден, "стандартты" сценарийлермен салыстырғанда одан үлкен көлемді деректерді өңдей алады. Екіншіден, жедел түскен үлкен деректермен жұмыс жасай алады, яғни жай ғана көп емес уақыт өткен сайын дерек көлемі көбейе береді. Үшіншіден, олар құрылымдалған және нашар құрылымдалған деректермен әр түрлі аспектілерде параллельді түрде жұмыс жасай алуы қажет. Үлкен өлшемді деректер алгоритмдері әрдайым құрылымдалған ақпарат ағынын ғана ала бермейді, осыдан көп ой түйіндеуге болады. Зерттеу нәтижелерін мақала авторлары үлкен өлшемді деректерді модельдеуде және Веб-қосымша әзірлеуде қолдану үстінде.
Түйін сөздер: Үлкен өлшемді деректер, ақпаратты өңдеу, талдау, модельдеу, әдістер.

Моделирование обработки большого объема данных

Балакаева Г.Т., Казахский национальный университет имени аль-Фараби, Алматы, Казахстан, +77013201802, E-mail: gulnardtsa@gmail.com,
Даркенбаев Д.К., Казахский национальный университет имени аль-Фараби, Алматы, Казахстан, +77012591891, E-mail: dauren.kadyrovich@gmail.com

Определение больших объемов данных, BigData, используется для обозначения таких технологий как хранение и анализ значительного объема данных, при обработке которых требуется высокая скорость, и принятие решений в режиме реального времени. Обычно, когда говорят о серьезной аналитической обработке, особенно если используют термин DataMining, подразумевают, что данных огромное количество. Не существует универсальных способов анализа или алгоритмов, пригодных для любых случаев и любых объемов информации. Методы анализа данных существенно отличаются друг от друга по производительности, качеству результатов, удобству применения и требованиям к данным. Оптимизация может производиться на различных уровнях: оборудование, базы данных, аналитическая платформа, подготовка исходных данных, специализированные алгоритмы. Большие данные – это совокупность технологий, которые призваны совершать три операции. Во-первых, обрабатывать большие по сравнению со "стандартными" сценариями объемы данных. Во-вторых, уметь работать с быстро поступающими данными в очень больших объемах. То есть данных не просто много, но их постоянно становится все больше и больше. В-третьих, они должны уметь работать со структурированными и плохо структурированными данными параллельно в разных аспектах. Большие данные предполагают, что на вход алгоритмы получают поток не всегда структурированной информации и что из него можно извлечь больше, чем какую-то одну идею. Результаты исследования используются авторами при моделировании больших данных и разработке веб-приложения.

Ключевые слова: Большие объемы данных, обработка данных, анализ, моделирование, методы.

1 Introduction

The analysis of a large volume of data requires a special technique, because when technical difficulties arise, it is only necessary to use them with the force of force, i.e., to use powerful equipment.

Of course, we can increase the speed of data processing, especially on modern servers and workstations, with multi-core processors, mass memory and powerful disk arrays. However, there are many other ways to scale-up large-scale data that does not require enormous hardware upgrades and endless hardware updates.

2 Review of literature

Consider the possibility of using a database management system. Modern databases cover different mechanisms, which utilizes significantly the speed of analytical processing:

– During the analysis, the most frequently used data can be pre-processed and can be stored in a special table in the form of multidimensional cubes in the database server ready in the upcoming processing.

– Cache tables to basic memory. During the analysis, you can cache the most commonly used data, such as definitions, by using the database resources. Reduces the access of the drive to a smaller number of times (Sosnov, 2002).

– Split tables into sections and table spaces. You can place data on separate discs, indices, and helper tables. This allows the parallel disk to read and write information to the database

management system. In addition, the table can be divided into sections. For example, we can use a logical table with historical data, when we need to divide it into small physical sections, this small part is read and not read all the history data.

Increasing the speed of searching information from the database in other ways: triggering rational indexing, query scheduling, parallel processing of SQL queries, using clusters, preparing data analyzed using stored procedures, database server triggers, and many of these mechanisms can be used with free databases (Vishnevskii, 2009).

3 Methods and models

Integration of data includes the integration of data from different sources and the provision of data to users in a unified form. This process becomes significant both in commercial tasks (when two similar companies need to combine their databases), and in scientific (combining research results from various bioinformation repositories, for example). The role of data integration increases when the volume and need for data sharing increases. Data integration systems can integrate data at the physical, logical and semantic level. Integration of data at the physical level from the theoretical point of view is the simplest task and is reduced to the conversion of data from various sources into the required uniform format for their physical representation. Data integration at the logical level provides the ability to access data contained in various sources in terms of a single global scheme that describes their joint presentation, taking into account the structural and possibly behavioral (using object models) data properties. Semantic data properties are not taken into account in this case. Support for a single presentation of data, taking into account their semantic properties in the context of a single ontology of the subject area, is ensured by the integration of data at the semantic level. The integration process is hindered by heterogeneity of data sources, in accordance with the level of integration. So, at integration on a physical level in data sources various file formats can be used. At the logical level of integration, there may be heterogeneity of the data models used for different sources or different data schemes, although the same data model is used. Some sources can be websites, and others – object databases, etc. When integrating on a semantic level, different ontologies can correspond to different data sources. For example, it is possible that each of the sources represents information resources modeling a fragment of the domain, to which its own conceptual system corresponds, and these fragments intersect.

The ability to increase speeds is not limited to optimize databases, and many things can be done by combining different models. The editing speed depends on the complexity of the mathematical device used. Analysis mechanisms are much simpler, the faster data is analyzed.

The data processing scenario can make the data run by using the sio data model (Glushakov, 2000). Here's a simple idea: no need to waste time on the data that does not require analysis. First of all simple algorithms are used. A part of the data that can be processed by these algorithms and cannot be processed even more complicated, are further analyzed. The rest of the data is replaced by the following processing steps, which are used in the following complex algorithms and chains. The final node of the development scenario uses the most complex algorithms, but the analysis data is several times smaller than the original model. As a result, all data will be diminished according to the total time needed to process.

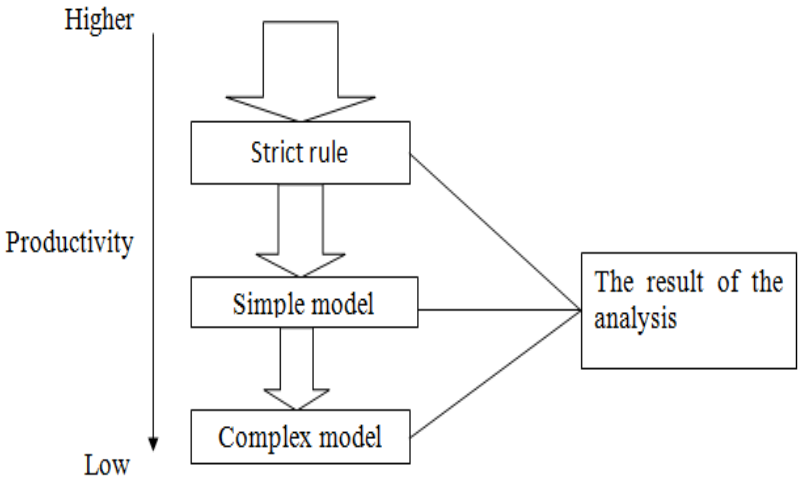

Figure 1 – Data processing using multiple templates

Here's an example of how to use this approach. In solving the demand forecasting problem, it is recommended to conduct an XYZ analysis, which will primarily determine how stable the demand for various goods is. Products of group X are sold regularly, and the use of predictive algorithms allows to obtain high-quality results. Products of Y are sold less often, so use of predictive algorithms gives them a good quality forecast. Z The range of products is very good, so it is not necessary to create these predictive models for them, and the need for them is calculated on the basis of simple formulas, for example, the average monthly sales volume. According to statistics, 70% of the product range is the product group Z. Products of Y are 25% and X-group products are 5%. Using a sophisticated model here is a 30% product only for the goods. Therefore, the use of the above method reduces analysis and prediction time by 5 to 10 times.

3.1 Parallel processing

Another effective strategy to process large volumes of data is to divide data into segments and separate segments of each segment and further consolidate the results. In most cases, we can say a few different subgroups of data. Maybe this is a group of consumers that can be a group of goods that can lead to a particular model.

In this case, instead of creating all sophisticated models, there are few simple things to do in each segment (Batini, 1992). This approach allows you to increase the speed of the analysis and reduce the amount of memory. Also, in this case analytical processing can be parallel, which also positively affects the time spent. In addition, models for each segment can create different analyzes.

In addition to speeding, this approach has one more important advantage: several simple models are easy to create and maintain apart from anyone else. The models can be launched in periods and the first results can be obtained shortly.

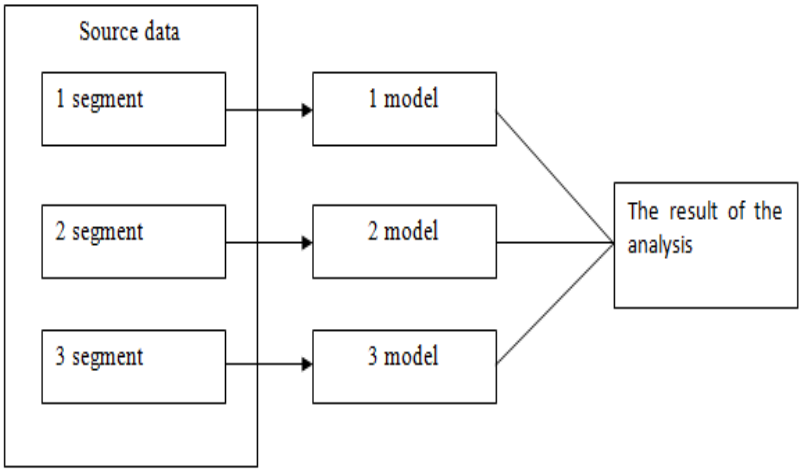

Figure 2 – Data segmentation and modeling

3.2 Representative selection

A representative selection is not all information, but some internal subset modeling can be used with large amounts of data. Properly representativeized representation contains the information needed to build a qualitative model (Blaha, 1997).

The analytical process is divided into 2 parts: modeling and applying it to new data. Creating a complex model is a resource process. Depending on the algorithms used, data is cached, scanned thousands of times, and auxiliary set of parameters is calculated. Applying the model to new data requires dozens and even hundreds of times less resources.

Thus, the model is designed for a small number of sets and will be applied to all data in the future. The result will be reduced to all data with full processing sequence.

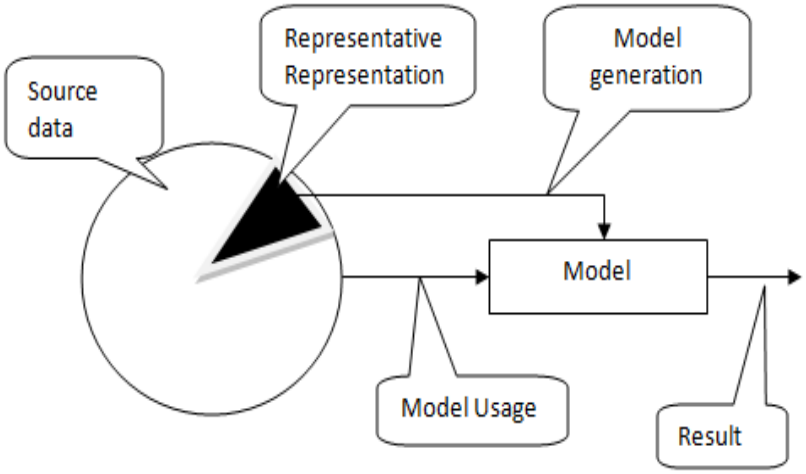

Figure 3 – Sampling (Blaha, 1997)

There are specific ways to get a representative selection, using analytical techniques we can analyze processing speed without losing quality. For example, sampling (The sampling

approach, which is English translation, means sampling.). There are special ways to increase sampling (see Figure 3).

3.3 New types of databases

Database – a set of interrelated data that can be used for a large number of applications, quickly receive and modify the necessary information. Database models are based on a modern approach to information processing. The structure of the database information allows to form logical records of their elements and their interrelations. Interconnections can be: one to one, one to many and many to many. The application of this or that type of interconnection is determined by three models of the database: hierarchical, network, relational. The hierarchical model is represented in the form of a tree-like graph. The advantage of this model is that it allows you to describe the data structure both at a logical and a physical level. Its drawback is a rigid fixed relationship between the elements. In this regard, any changes in relationships require a change in its structure. In addition, the speed of access was achieved due to the loss of information flexibility, i.e. it is impossible to obtain information located on another branch of the communication in one pass through the tree. This model implements the one-to-many type of communication. The network model of the database is represented as a link diagram. In a network model, any kind of relationship between records is allowed; there are no restrictions on the number of feedbacks. The principle of many to many is used. The advantage of this model is the greater information flexibility in comparison with the hierarchical model, but there is a drawback – the rigidity of the structure. If you need a frequent reorganization of the information base, you use the most advanced model of the database – the relational database, in which there are no differences between objects and relationships. The type of connection of such a model is one to one. In this model, the relationships between objects are represented as two-dimensional tables-relations. Since any data structure can be converted into a simple two-dimensional table, and this representation is most convenient for both the user and the machine, the overwhelming majority of modern information systems work precisely with such tables, i.e. with relational databases.

Due to the growing volume of information, hard disk space is difficult to deal with (it is relatively easy to solve some of the difficulties) and it is important to get timely access to the data. Sophisticated caches can be used, but it does not help in the end. You can divide the database into each part and insert each part into our databases. When the database volume increases, the speed of the system decreases dramatically. One way to save time access is to place the database in RAM (Obukhov, 2014). This technique is 100 times faster than a rational method. The in-memory database, the IMDB database, uses the computer's express drive to store data, which means the Quick Storage Device is a place to store data in such systems. Due to the fact that the cost of memory is rising day by day as a place of storage, it is efficient and at speeds of data processing (Frenk, 2014). There are new types of databases that can be self-analyzed to work with large volumes of data. Currently, this statement is used by the general database. Terrada's developers created the first self-analysis database (Boncz, 2005). Also one of the database types is the column data store. In recent years, a number of database systems have been created, including MonetDB (Boncz, 1999), (Stonebraker, 2005) and C-Store (Abadi, 2008), which store data on the column.

4 Results and discussion

Analysis of large databases – this is a major scalable report, and in many cases it is not actually solved. Modern databases and analytical platforms offer several ways to handle these issues. If we use them efficiently, you can edit data terabytes that are accurate at speeds. The results of the analytical study are used by the authors in modeling large volumes of data and developing the Web application, and for improving the results of large-scale data processing in the model coordination.

5 Conclusion

These methods are only a small part of the methods that allow you to analyze large amounts of data. There are other methods, such as special scalable algorithms, hierarchical models, reading windows, and more application. According to the design of these systems, these systems give a good result in performing some workloads, especially in database applications with many requests, providing the desired results when reading data in a workload (Balakayeva, 2013).

References

[1] Abadi J. Daniel, Madden Samuel, Hachem Nabil. ColumnStores vs. RowStores: How Different Are They Really?, *Proceedings of the ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, June 2008*, vol. 3, (2008): 57-61.

[2] Balakayeva G. and Nurlybayeva K. Simulation of Large Data Processing for Smarter Decision Making. *AWERProcedia Information Technology Computer Science, 3rd World Conference on Information Technology*, vol. 03, (2013): 1253-1257

[3] Batini C., Ceri S. and Navathe S. Conceptual Database Design: An Entity-Relationship Approach. *Redwood City, CA: Benjamin Cummings*, (1992): 185 p.

[4] Blaha M. and Premerlani W. Object-oriented modeling and Design for Database Applications. *Prentise Hall*, (1997): 201 p.

[5] Boncz P., Zukowski M. and Nes N. MonetDB/X100: Hyper-pipelining query execution. *In CIDR*, (2005): 324 p.

[6] Boncz P. A. and Kersten M. L. MIL primitives for querying a fragmented world. *VLDB Journal*, vol. 8, no 2 (1999): 101-119.

[7] Frenk B. Ukrashenie bol'shikh dannykh: kak izvlekat' znanie iz massivov informacii s pomosh'iu glubokoi analitiki [Exploitation of most data: how to search for information from mass media analysts with help]. *Moscow*, (2014). 127 p.

[8] Glushakov S.V. Lomat'ko D.V. Bazy dannykh: uchebnyi kurs [Databases: training course]. *Moscow: OOO "Izdatel'stvo ACT"*, (2000): 504 p.

[9] Obukhov A. In-Memory. Baza dannykh v operativnoj pamjati [In-Memory. Databases in RAM], (2014): 128 p. http://ecm-journal.ru/post/In-Memory-Baza-dannykh-v-operativnojj-pamjati.aspx

[10] Sosnov A. Osnovy proektirovanie informacionnykh sistem [Basics of projection of information systems]. *Moscow: DMK Press*, (2002): 1020 p.

[11] Stonebraker M., Abadi D. J., Batkin A., Chen X., Cherniack M., Ferreira M., Lau E., Lin A., Madden S. R., O'Neil E. J., O'Neil P. E., Rasin A., Tran N., and Zdonik S. B. C-Store: A Column-Oriented DBMS. *In VLDB*, (2005): 553-564.

[12] Vishnevskii A. SQL Server. Effektnaya rabota [SQL Server. Effective work]. *Sankt-Peterburg*, (2009): 541 p.