Acta Technica *62* (2017), No. 7A, 269–276

Research on nonlinear regression SVM algorithm for large data cluster spark architecture

Jianfei Shen[1]

Abstract. A kind of effective searching method for big data is improved. When search request appears, search request intention of user will be analyzed, key words will be provided to user for selection, and key words will be segmented after key words finally used by user are determined, and thematic words and auxiliary words will be extracted. Thematic words and auxiliary words are matched with history search successively to obtain search result. Contrastive analysis of simulation experiment shows that improved searching method based on big data increases successful match probability through word segmentation, utilizes history search result better, and saves searching time and improves search efficiency.

Key words. Big data, Searching, Cloud database search, Cluster, Regression algorithm.

1. Introduction

IBM research shows that human created 90% data of the world in the past 2 years; IDC research shows that data storage content in the future 10 years will increase by 50 times[2]. With approaching of big data[3] era, to process big data better, many researchers realize parallel processing to big data through the mechanism where computational efficiency is improved by cloud computing technology, which improves processing capacity to big data from different angles. Parallel programming frame Google MapReduce[4] of Google, Twister[5] and Haloop[6] improving iterative computation processing, Pregel[7] improving processing capacity to big data of graphic calculation are mainly contained, and Clustera, Dryad[8], Spark[9] and Hadoop++[10] etc. are also contained.

[1]Hunan Mass media Vocational Technical College, Hunan Changsha, 410100, China

2. Improved big data searching frame

Fig.1 shows basic frame of improved big data searching. The frame analyzes search request intention of user according to new search request from user, provides key words to user for selection, and segments key words after key words finally used by user are determined, and extracts thematic words and auxiliary words. Then it judges whether precedent aimed at the search or partial search exists in history search according to thematic words in key words, and if any, then history search result can be shared to reduce time consumption caused by all searches aimed at big data set again.

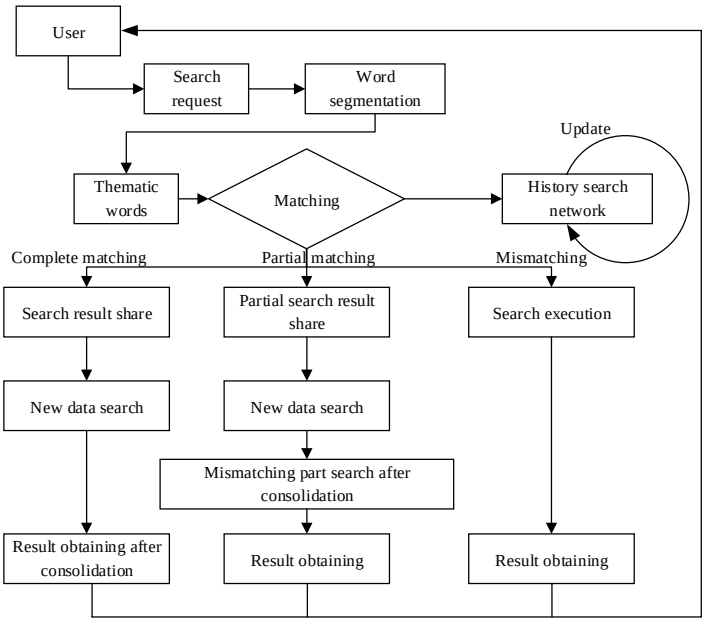

Fig. 1. Improved big data searching frame

The frame contains following parts:

Step 1: user proposes search request firstly.

Step 2: analyze search intention of search request, provide key search words with similar search intention to user for selection, and obtain final key search words of user. Segment key words and extract thematic words and auxiliary words.

Step 3: match thematic words for search with history search network, and 3 kinds of condition of matching result appear.

(1) Complete matching. Complete matching shows that new search request from user only contains thematic words and thematic words appeared once, so search result in the past search can be utilized by the search directly, which means that the search result of same history search can be shared. Because history search is just search to data before a certain period of time, new data record may be produced after the history search, and therefore, newly increased data shall be searched to

obtain new search result. Combine history search result and new search result to obtain final result required by user.

(2) Partial matching. Partial matching shows that thematic words and auxiliary words have been extracted from key words of new search request from user, of which thematic words appeared once, so search result in the same past search can be utilized by the search directly, which means that the search result of same partial history search can be shared. Because history search is just search to data before a certain period of time, new data record may be produced after the history search, and therefore, newly increased data shall be searched to obtain new search result. Combine history search result and new search result to obtain result of match part in search. Search auxiliary words continuously in result of match part in search to obtain final result required by user.

(3) Mismatching. Complete mismatching shows that there is no history search record for share in new search request from user, and all searches shall be re-executed to obtain result required by user.

Step 4: feed back search result required by user to user.

Step 5: update history search network.

3. Key searching technology of big data after improvement

Under improved big data searching frame, key technologies, such as method for provision of key search words of search intention in the frame, matching technology of key search words and history search network, realization method for newly increased data search, and update method of history search network etc., will be mainly introduced as follows.

3.1. Method to extract key search words based on search intention

Search request of user is a series of independent short text in fact. After system receives search request of user, it will analyze search intention of search request firstly, provide key search words with similar search intention to user for selection, and obtain final key search words of user. Then it will segment key words[19]. A word vector[20] will be obtained after word segmentation, of which every word is marked with the characteristic, such as noun, verb, adjective and noun of locality etc. Contribution of words with different characteristics to thematic expression is different, of which verb and noun have the maximum contribution to thematic expression and identification, so the 2 kinds of characteristics can be mainly considered in word frequency statistics and words with other characteristics can be ignored. Finally, thematic words and auxiliary words will be extracted from search request of user.

HTTPCWS, CC-CEDICT, IK, Paoding, MMSEG4J. There are numerous different algorithms and tools for Chinese word segmentation. Existing word segmentation algorithms[11] can be divided into three types: word segmentation method based on string matching, word segmentation method based on understanding and

word segmentation method based on statistics. Common word segmentation tools include SCWS, FudanNLP, ICTCLAS, HTTPCWS, CC-CEDICT, IK, Paoding, and MMSEG4J etc.

3.2. Matching of new search request and history search network

Fig.2 shows a schematic diagram on matching of new search request and history search network.

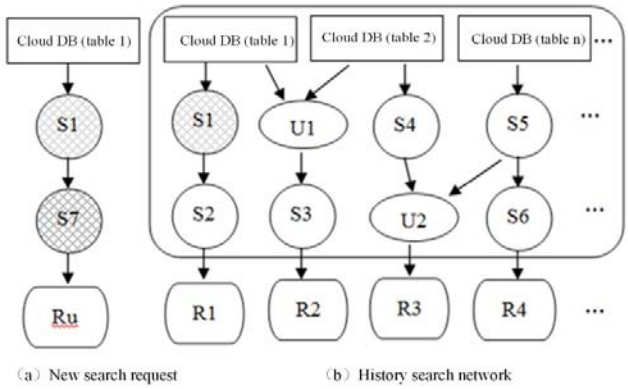

Fig. 2. Schematic diagram on matching of new search request and history search network

For application of big data, cloud database is generally used to store data record, such as BigTable of Google, and Hbase cloud database of Apache etc.

Improved matching algorithm for matching of search request and history search network is as follows:

Input: new search request (such as S1 and S7), history search network; output: matching result of new search request and history search network.

Analyze search intention of user according to search request, provide key search words concerned with search intention and obtain final key search words of user.

Match with history search network in cloud database;

do case: key words search (such as S1; S1 and S7; S7)

case: complete matching (such as S1)

Obtain history search result matching with S1 at node S1 in cloud database;

Search newly increased data after history search result deadline and obtain newly increased search result;

Combine history search result with newly increased search result to obtain final result, feed back it to user and update content of node S1.

case: partial matching (such as S1 and S7)

Obtain history search result partially matching with S1 at node S1 in cloud database;

Search newly increased data after history search result deadline and obtain newly increased search result;

Combine history search result with newly increased search result to obtain final result, and update content of node S1;

Search S7 continuously in S1 to obtain result Ru, and feed back it to user and record the content to node Ru.

If there are additional key words, make matching continuously, obtain result and feed back it to user and record the content to corresponding node.

case: mismatching (such as S7)

Search S7 in cloud database, obtain result and feed back it to user and record the content to node (S7).

End Case

End of algorithm.

3.3. Realization method for newly increased data search

Assumed that cloud database table is used to store all records. One important feature of cloud database is that when new record is stored, because all records are supplemented in supplemental mode, all new records will be supplemented according to supplement time sequence, which makes it quite convenient to obtain supplemental record, meaning that the last record of history search shall be found. Therefore it is quite easy to obtain the number of newly increased record.

Assumed that a user needs to search all records containing "automobile" in microblog records of a website in this year, and someone made the same search historically, but what is searched by the person is all records containing "automobile" before May of this year, because new microblog records are supplemented continuously, only all history records of the history search before May can be shared, and newly increased data after May shall be searched again.

If the number of microblog record for "automobile with automatic catch" is to be searched, the condition of "automatic catch" shall be searched continuously in 70000 microblog records meeting "automobile" condition to obtain final result.

3.4. Update method of history search network

Relationship on matching of new search request and history search network is analyzed in Fig.2. Through analysis, it is obtained that search result of partial searches can be shared, but new search condition S7 is not contained in history search network, so (a) and (b) in Fig.2 shall be combined, and updated search network obtained is as shown in Fig.3.

4. Simulation experiment

(1) Construction of experimental environment

Hadoop is used as experimental environment, 1 set is Master node, 1 set is shadow node of Master node, and 2 sets are Slave nodes in 4 sets of machine used. Linux operating system and various modules concerned with Hadoop are installed for 4 sets of machine.

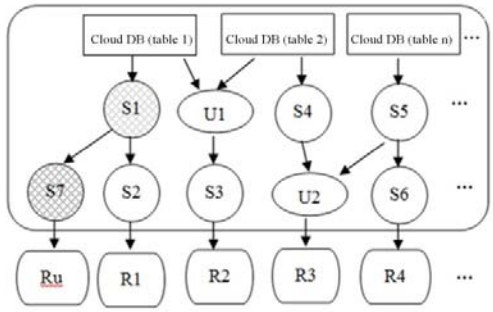

Fig. 3. Updated search network

(2) Comparison of experimental methods

The experiment compares improved big data search method with unimproved big data search method.

Comparison on simple query searching. Effective unimproved searching method based on big data and effective improved searching method based on big data are mainly used to compare big data search.

Comparison on complex query searching. Effective unimproved searching method based on big data and effective improved searching method based on big data are used to realize join search between 2 datasets, because the most representative search in complex search is join search, and it involves big data computing problem among numerous datasets. Its efficiency will affect implementation of real-time search for complex search directly.

(2) Simulation experiment

Simulation experiment is as follows:

1) Comparison on simulation experiment of simple query searching

Simulation experiment 1 mainly verifies time comparison of single key word (such as "automobile") and numerous key words (such as "automobile with automatic catch") searching in unimproved and improved methods. Conclusion of simulation experiment 1 is as shown in Fig.4.

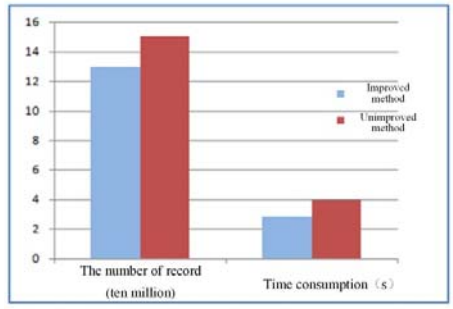

Fig. 4. Simulation result of simulation experiment 1

Improved searching method firstly makes word segmentation when executing

searching, thematic word is "automobile" and auxiliary word is "automatic catch". Match thematic word "automobile" with history search network, and because there is searching on records containing "automobile" in past history, only record of newly increased part shall be searched again. "Automatic catch" shall be searched based on record after history search result is combined with newly increased search part to obtain result. Fig.4 shows a comparison on searching record of improved searching method and unimproved searching method, from which it can be seen that fewer searching records need to be searched in improved searching method, thus reducing searching time greatly and improving search efficiency.

2) Comparison on simulation experiment of complex query searching

Simulation experiment 2 mainly verifies and searches Join connection of 2 datasets. The simulation experiment chooses 2 tables from dataset directly for complex query calculation of Join connection. Result of simulation experiment is as shown in Fig.5. Fig.5 shows comparison on time consumption of unimproved searching method and improved searching method, from which it can be seen that our searching method reduces searching time and improves search efficiency because it only needs fewer searches to record.

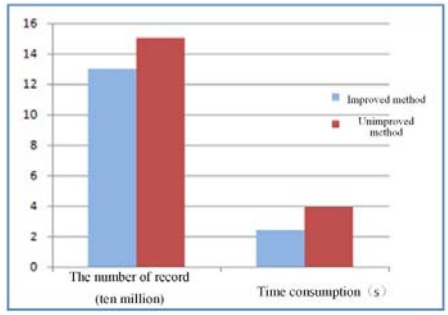

Fig. 5. Result of simulation experiment 2

5. Conclusion

Aimed at low search efficiency of big data, this paper improves a kind of effective searching method. Main improvement point is that when new search request appears, search request intention of user will be analyzed, key words will be provided to user for selection, and key words will be segmented after key words finally used by user are determined, and thematic words and auxiliary words will be extracted. Successful match probability is increased through word segmentation, and history search result can be utilized better to quicken search. Contrastive analysis of simulation experiment shows that improved searching method based on big data utilizes history search result better, reduces a great deal of repetitive computation to history search, and saves searching time and improves search efficiency.

Acknowledgement

Project supported by Hunan Natural Science Foundation of China in 2017 (No. 2017JJ5008).

References

[1] S. Shu, L. Ren, Y. Ding, et al.: *SVM optimization algorithm based on dynamic clustering and ensemble learning for large scale dataset*[C]// IEEE International Conference on Systems, Man and Cybernetics. IEEE, (2014), 2278-2283.

[2] K. B. Lin, W. Weng, R. K. Lai, et al.: *Imbalance data classification algorithm based on SVM and clustering function*[C]// International Conference on Computer Science & Education. IEEE, (2014), 544–548.

[3] J. Sai, B. Wang, B. Wu: *BPPGD: Budgeted Parallel Primal Gradient Descent Kernel SVM on Spark*[C]// IEEE International Conference on Data Science in Cyberspace. IEEE, (2017), 74–79.

[4] T. Li, X. Liu, Q. Dong, et al.: *HPSVM: Heterogeneous Parallel SVM with Factorization Based IPM Algorithm on CPU-GPU Cluster*[C]// Euromicro International Conference on Parallel, Distributed, and Network-Based Processing. IEEE, (2016), 74–81.

[5] S. Daengduang, P. Vateekul: *Applying One-Versus-One SVMs to classify multi-label data with large labels using spark*[C]// International Conference on Knowledge and Smart Technology. IEEE, (2017), 72–77.

[6] S. M. H. Ho, M. Wang, H. C. Ng, et al.: *Towards FPGA-assisted spark: An SVM training acceleration case study*[C]// International Conference on Reconfigurable Computing and Fpgas. IEEE, (2017), 1–6.

[7] K. Manimala, I. G. David, K. Selvi: *A novel data selection technique using fuzzy C-means clustering to enhance SVM-based power quality classification*[J]. Soft Computing, *19* (2015), No. 11, 3123–3144.

[8] X. L. Xie, Z. Y. Liao, G. Y. Cai: *Data Classification of SVM Based on PSO*[J]. Lecture Notes in Electrical Engineering, *270* (2014), 311–317.

[9] R. Ayyagari, A. Sivakumar, K. Kannan: *Development of K- means based SVM regression (KSVMR) technique for boiler flue gas estimation*[J]. International Journal on Electrical Engineering & Informatics, *6* (2014), No. 2, 359–373.

[10] J. L. Reyes-Ortiz, L. Oneto, D. Anguita: *Big Data Analytics in the Cloud: Spark on Hadoop vs MPI/OpenMP on Beowulf* [J]. Procedia Computer Science, *53* (2015), No. 1, 121–130.

[11] Z. Zhang, T. Wen, W. Huang, et al.: *Automatic epileptic seizure detection in EEGs using MF-DFA, SVM based on cloud computing.* [J]. Journal of X-ray science and technology, *25* (2017), No. 2, 261.

[12] W. Lin, Z. Wu, L. Lin, et al.: *An Ensemble Random Forest Algorithm for Insurance Big Data Analysis*[J]. IEEE Access, *5* (2017), No. 99, 16568–16575.

[13] H. Tao, B. Wu, X. Lin: *Budgeted mini-batch parallel gradient descent for support vector machines on Spark*[C]// IEEE International Conference on Parallel and Distributed Systems. IEEE, (2015), 945–950.

[14] C. Binnig, A. Crotty, A.Galakatos, et al.: *The end of slow networks: it's time for a redesign*[J]. Proceedings of the Vldb Endowment, *9* (2016), No. 7, 528–539.

Received May 7, 2017