SELF-OPTIMIZING REAL-TIME RECOMMENDATION MODEL DESIGN BASED ON YARN

Tao liu Shuang wang Peng wu

School of Computer Science and Technology
Wuhan University of Technology
Wuhan, China
e-mail:1272348068@qq.com
e-mail: ws123xhx@gmail.com
e-mail: 425664759@qq.com

ABSTRACT

In real-time personalized recommendation systems, aimed at some existing issues such as timely system performance requirements,large data processing capacity,the existence of bidirectional cold-start,the difficulties of spotting the potential hot news,the deficiencies of system self-optimization and etc,we propose enhanced self-learning optimization model based on the YARN platform,which illustrates algorithm scheduling framework and **s**elf optimization system revised by users' feedback**.**Furthermore,to fully exhibit the superiority of the YARN platform,the label propagation algorithm here used exemplifies the polymerization process of raw large data.Thus,a highly prompt recommendation system overall is achieved.

Keywords: Real-time recommendation model , Self-optimizing, Scheduling algorithm framework, YRAN platform

1. INTRODUCTION

In the personalized recommendation system,algorithm scheduling framework as a link connecting recommendation algorithms with product features,will directly determines the content recommended for users and the costs required to achieve the desired effects.For strongly real-time articles,such as news designed for specific subscribers,huge amount of information along with short life cycle makes the traditional collaborative filtering algorithm unavailable[1],since it lacks the ability of accumulating sufficient data in a short period of time,and overly relies on similarity calculation.Or no good results will be gained.In the paper,scheduling algorithm framework proposed is a kind of reinforcement learning[2],to maximize their cumulative returns by series of activities in uncertain circumstances.

In the model,a system self-evaluation mechanism is introduced that makes it possible to automatically select desirable algorithm from the algorithm pool according to users' feedback.In this way,we can achieve the purposes of self-optimization,which leads to a situation of fittest algorithm, so that recommended results gradually are refined.When the feedback is large enough,covering sufficient user types,algorithm diversity increasingly becomes rich.Some of them meet the universal needs of users,while some satisfy only a small fraction of users' needs.Constantly replenished by tthe algorithm to explore the parameter space in theirs, the system will greatly appeal to its users and optimize with the expansion of its users' scale.

Data processing approach based on YARN platform provides a very good solution to parallelism problems during the process of big data.YARN is the upgraded generation of Hadoop computing framework, which not only supports the original programming model MapReduce,and supports a wide variety of distributed computing frameworks such as MPI, Spark, Tez and so on.In this paper,data aggregation process is illustrated by label propagation algorithm[3] used under the YARN platform to show the superiority of the data processing.

2. REAL-TIME RECOMMENDATION MODEL

In real-time system of optimization recommendation, users' browsing history and residence time as well as feedbacks of provided content will be timely collected by the server.On server side or cloud,initial data will be preprocessed,including data cleaning,noise reduction, and then data aggregation[4],the processing of miss data[5].In this way,we would take a collection of available data.Scheduling algorithm framework first will be calculated based on recommendations provided by the optimization algorithm.But when there is feedbacks.Scheduling framework will choose a suitable algorithm based on feedbacks to select an appropriate algorithm combined with personal weight-vector model in order to present results closer to individual users.The evaluation system will in turn determines appropriate algorithm based on the feedbacks.Such non-stop loop, we form a system with the ability of self-evolution.In this way,the personalized recommendation is more accurate.Meanwhile recommendation system involved in large amounts of data processing will be tackled by taking advantage of a new generation of big data computing framework that supports multiple computing paradigms.We have different chooses on different applications.The Process above shown in Figure 1.

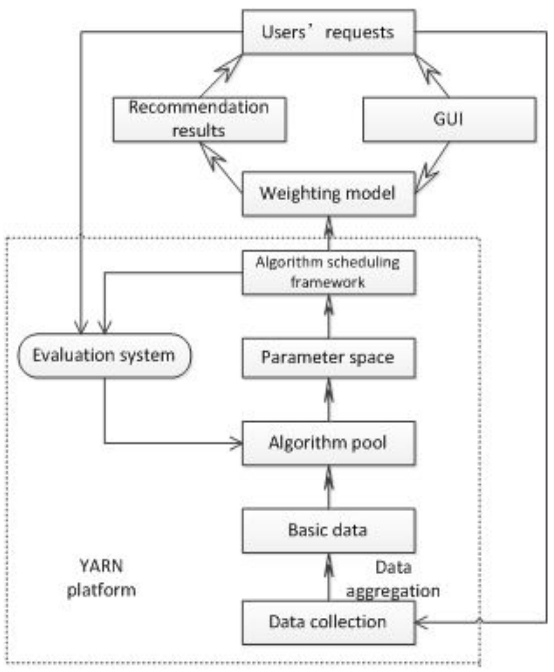

Figure 1: Recommendation System Model

Next,three major components of the system, respectively algorithm scheduling framework, evaluation system and data processing on YARN, are introduced.

3. SCHEDULING ALGORITHM FRAMEWORK

First,we have to admit environment is uncertain and information is insufficient, with the knowledge of highly dynamic real-time recommendation system. However, long-term interaction between users and recommendation system is to constantly optimize the process. In simple terms, if the system recommends an article to a user, the user does not click or does not like, and then we can conclude the recommendation system fails, but this process will be regarded as a small part of decision-making process with users' feedback as payoffs, where the dislike is treated as a negative return while like as a positive return. This framework is to solve the problem of how to get a greater return at less expense.

In classical model of gambling armed context[6], the enhance learning scheduling framework is adopted to solve the problem of return maximization effectively. On assumption that news recommendation system takes whether a piece of news is clicked as a return, its frame is defined as follows. A proceeds in discrete trials t = 1, 2, 3 … In trial t:

1. The algorithm observes the current user u_t and a set A_t of arms or actions together with their feature vectors $x_{t,a(a\in A_t)}$ that summarizes information of both the user u_t and arm a, and will be referred to as the context.
2. Based on observed payoffs in previous trials, A chooses an arm $a_t \in A_t$, and receives payoff r_{t,a_t} at whose expectation depends on both the user u_t and the arm a_t.
3. The algorithm then improves its arm-selection strategy with the new observation $(x_{t,a_t}, a_t, r_{t,a_t})$,It is important to emphasize here that no feedback (namely, the payoff $r_{t,a}$) is observed for un-chosen arms $a \neq a_t$.

In the process above, the total T-trial payoff of A is defined as $\sum_{i=1}^{T} r_{t,a_t}$. Similarly, we define the optimal expected T-trial payoff as $E[\sum_{i=1}^{T} r_{t,a_t^*}]$, which is the arm with maximum expected payoff at trial t. Our goal is to design A so that the expected total payoff $\sum_{i=1}^{T} r_{t,a_t}$ is maximized. Equivalently, we may find an algorithm so that its regret with respect to the optimal arm-selection strategy is minimized. Here, the T-trial regret of algorithm A is defined by

$$R_A(T)=E[\sum_{i=1}^{T} r_{t,a_t^*}]-E[\sum_{i=1}^{T} r_{t,a_t}] \qquad (1)$$

In the context of article recommendation, we may view articles in a pool as arms. When a presented article is clicked, a payoff of 1 is incurred; otherwise, the payoff is 0. With this definition of payoff, the expected payoff of an article is precisely described as its click through rate (CTR), and choosing an article with maximum CTR is equivalent to maximizing the expected number of clicks from users, which in turn is the same as maximizing the total expected payoff.

Algorithm strategy is improved to minimize the difference between overall return and the optimal return when implementing the scheduling framework. Different demands of access speed and computation capacity for on-line rapid responses and training models should be compromised. For example, the feature vector calculation and iteration requires offline models, whereas comparing payoffs caused by different behaviors needs an online one.

In portal, because of its total large browse amounts, and a great number of new users who do not browse history records. it is difficult for traditional offline methods to mine the characteristics of a user so as to meet the needs of time-sensitive product, a condition called cold start [7]. In the scheduling algorithm, its essence is to balance exploration and exploitation. When the system running, it will gradually get accustomed to new environment through digestion and absorption, so cold start problem in such a scheduling framework becomes a process of progressive improvement.

News Consulting, generally have strong social transmissibility, if the trend of news outbreak is able to be expected, which can be made the best use of, and then it would make a big difference to deliver the potential interesting news to the users and further affect subsequently news propagation. In the scheduling framework, we combine a variety of needs-adjusted return functions. For hot news, with good returns in the short term for all types of users, may be a potential hotspot.

4. EVALUATION SYSTEM

In personalized recommendation systems, the establishment of data-evaluation system is the key. In this model, the self-test system can effectively evaluate the merits of the algorithm to achieve self-optimization functions. When the user receives feedbacks, scheduling algorithm framework will choose different evaluation systems, depending on the circumstances. Then the results will be submitted to the algorithm pool for algorithm selection. Figure 2 show that

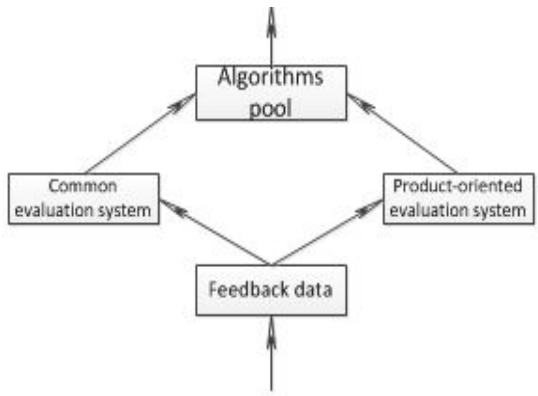

Figure2:Evaluation System

4.1. Common evaluation system.

In common evaluation system, ROC(Receiver Operating Characteristic) curves is used to measure dynamic capabilities of model and to test the effect of recommendation algorithms[8]. The common confusion matrix used in model shows the distribution of predicted results, shown in Figure 3

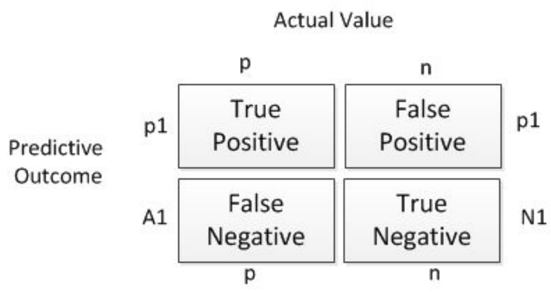

Figure3:Confusion Matrix

Based on the difference between predicted and actual values, there are four possible results: true positive, false positive, false negative and true negative. Of those, true positive rate (TPR) and false positive rate (FPR) is an indicator concerned. ROC curve is drawn to show the relations difference TPR and FPR in different parameters. showed in Figure 4

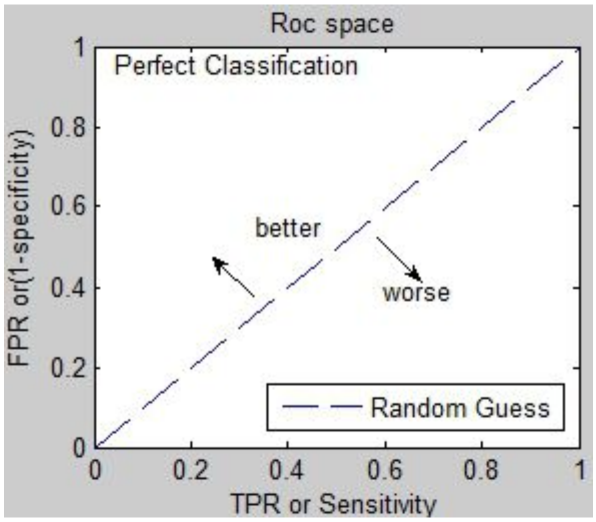

Figure4:ROC Curve

In the ROC curve, the vertical axis is TPR, characterizing the sensitivity of the model, and the horizontal axis is FPR, characterizing model's discrimination. Upper left corner of the graph features means model has perfect coverage and discrimination, and the diagonal line indicates a completely random classifier better than random classification model, the ROC curve in the upper left diagonal, while the area under the ROC curve (AUC) presents dynamic predictive ability of the model.

ROC curve can comprehensively reflect the ability to adapt to different model parameters and the external environment changes. In the personalized recommendation system, ROC curve ,as an evaluation indicator, can also be applied to optimization goals improved slightly. For example, score prediction, when prediction error rate is lower than a certain threshold, is classified correctly, otherwise considered wrong. Top-K recommendation problem is equivalent to direct classification of given data sets [9].

4.2. product-oriented evaluation system.

The common evaluation above has two major problems when applied to specific product application. First, from a technical point of view, it ignores the difference between offline and online measurements, which may result in model over-fitting [10]. Secondly, from the product point of view, it fails to consider the core goal of recommendation algorithm, ie, creating long-term value for users because even the advancement, but only turns out to be a short-term optimization course. Based on the considerations mentioned above, a optimization framework of product-oriented personalized recommendation system is proposed. Shown in Figure 5. The algorithm evaluation measurement compactly combined with products' core indicators optimizes the system to resolve problems in reality.

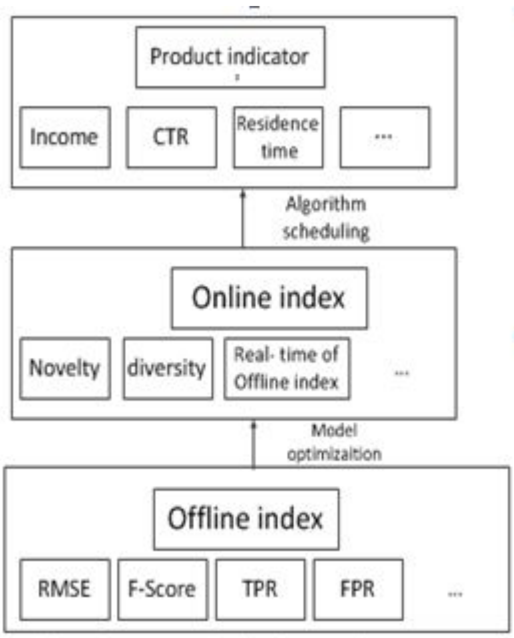

Figure5: Product-oriented Evaluation System

We want to solve the excessive reliance on offline criteria used to evaluate the merits of the algorithm. On the one hand, off-line indicators are required to be prompted in real time. By regular online calculation observe and evaluate the quality of recommendation algorithms, and make the appropriate adjustments; on the other hand, a reasonable gray on-line or A/ B testing system is established to ensure the improvement of personalization algorithm. Whether an improvement or a new algorithm is to be used on line, depends on eligible performance in offline environment, and then under the online environment stability and preliminary effects tested by a weight, further formulating improvement approach applied to corresponding fields.

If there are several parallel improvements or trials are aimed at the same user or the same algorithm function, then try to conduct rigorous A/B test to identify the final decision by comparing effects on the randomly selected groups of test users under the premise of constant environment variables. In contrast, most previous work focused on offline assessment, Our work make a shift to online assessment to optimize the recommendation system, which also improves workflow and working patterns.

5. DATA AGGREGATION PROCESSING USING LPA BASED ON YARN PLATFORM

YARN is the next generation of computing framework Hadoop, a huge reconstruction of the first-generation MapReduce framework (MPv1). YARN in architecture and design is more flexible, more extensive than MRv1. The calculation framework shown as figure 6:

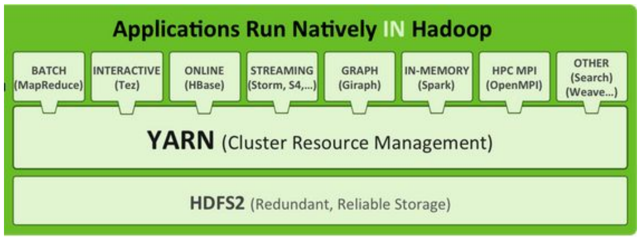

Figure 6 :Yarn Framework

In this section, we describe in detail how to use the LPA algorithm uses MapReduce calculation method to achieve data aggregation process data from a large collection data to basic data, the process is shown in Figure 7

Figure7: Data Aggregation Processing

5.1. LPA algorithm.

Label propagation algorithm is a standard graph mining algorithm [12], in which the object is a undirected graph made of nodes and edges. LPA is widely used in polarity classification[13] and social network community discovery[14]. In this paper, to model mining keyword from news or articles, each node represents a user in the diagram, where each node's ID uniquely identify itself and nodes'edges represents relationships between users, such as the same area, similar preferences, etc. For label propagation algorithm, each node needs initially labeled "tag" information. In such applications, the label of each node is the user's browsing history. For a new user, the corresponding node is labeled null, which initializes the construction of graph for LPA.

According to the executing process of LPA, node labels automatically get updated during iterations to complete the label propagation. The rounds of iterations can be specified in advance or the process does not terminate until node labels change little. In each round of iteration, nodes take turns to perform the following steps to update labels.

1. Locate the neighbor nodes with edges linked to node A, and the neighboring nodes set S, representing users associated with the user A, is comprised of these neighbors and A itself;
2. Count the frequencies of different labels S. Node A's initial label with f as its frequency, that is,a user 's own label is given a larger initial frequency ;
3. The weight of each A's label deduced from the second step then is calculated by the formulation like IF*IDF, and these labels ranking the highest K in the order of their weights form A's new label collection.

Through the process above, node A's label collection is updated.

5.2. Based YARN's MapReduce parallel data processing

Considering the large amount of users on portal websites, and the parallel feature of certain calculation, we use distributed computing framework MapReduce. The original data for the subsequent calculation derives from two sources, namely neighboring data associated with the user in the form <uid, list(uid)> to present a list of other users associated with uid, and initial tag information in the form <uid, list(word)> to represent the initial labels relevant to the uid. In order to facilitate the subsequent process, firstly, these two types of data are combined in a format <uid, list(uid),list(word). pseudo-code for this process is following:

```
    map(string key , string value)：//key: uid ; value: list(uid)+list(word)
    /* parse value*/
     list uidList=parse(value)
  list wordList=parse(value)
    /*PLA*/
     for each u in uidList :
     EmitIntermediate(u,AsString(wordList));
    If  u==key:// initial label's frequency of each node
       int i=0
       While i<f;
         EmitIntermediate(u,AsString(wordlist));
          i=i+1
    reduce(string key,Iterator values) ： //key : uid values : list(word)列表
    /*count label's frequency*/
    map<string ,float> tfCounter;
    for each v in values:
     list wordList=parse(v)
       for each w in  wordList:
          tfCounter[w]+=1
    /*count each label's weight*/
    For each lable in tfCounter.keys():
       float weight=TfIdf(lable)
    tfCounter[lable]=weight
    /*take Top K*/
    List topLabelList=GetTopK(tfCounter,K)
    Emit(AsString(topLabelList));
```

LPA algorithm processing data completes the update process of a set of user tags. The user can specify the number of iterations, the calculation process can be completed. Finally obtain an ID and a keyword.

6. CONCLUSION

Real-time optimization system based on YARN platform can be effectively self-optimized and conduct data parallel processing, to achieve well-performing effects in recommending the personalized real-time news to users. This model still has some room for improvement. Future researches focus on the following aspects:

In the context of algorithm scheduling framework, algorithm optimization processing from the algorithm pool should give consideration to both personalized recommendation effects and accepted efficiency. Sometimes the system has to be designed to strike a balance between the two. From the perspective of computational complexity and universality, in-depth analysis and improvement of the performance of algorithms is needed. But also the independence requirements of different algorithms within its parameter space should be guaranteed performing data processing. At the same time to get the best results, the results of each independent parameter space can be self-exchanged.

Taking better methods to solve data collection preprocessing work is expected. For Problem with large basic data, deficient useful data, incomplete data information, a better solution helps provide a better structure, more convenient treatment, more qualitative basic data. This is conducive to the efficiency and accuracy of subsequent system processing.

More work should be done to improve the robustness and accuracy of the self-regulating system with feedback. It's essential to promptly and effectively respond to users' request based on the previous feedback. In despite of there being gradual improvement during the feedback process, how to ensure a certain degree of accuracy at the outset will influence the performance of the whole system.

ACKNOWLEDGEMENTS

This work was funded by National College Students Innovation and Entrepreneurship Training Program .The program ID is 20131049710003.The authors also wish to thank the instructors for several helpful comments and suggestions.

REFERENCES

Sarwar, B. Karypis, G., Konstan, J., & Riedl, J. 2001,Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web, pp. 285-295. ACM.

Barto, A. G. ,1998. Reinforcement learning: An introduction. MIT press.

Zhu, X. & Ghahramani, Z. 2002. Learning from labeled and unlabeled data with label propagation, Technical Report CMU-CALD-02-107, Carnegie Mellon University.

Han J, Kamber M, Pei J. 2006. Data mining: concepts and techniques,Third Edition.Singapore Elsevier Inc.

Miller, B. N., Konstan, J. A., & Riedl, J. 2004. PocketLens: Toward a personal recommender system. ACM Transactions on Information Systems (TOIS), 22(3), pp.437-476.

Lihong Li,Wei Chu,John Langford,Robert E.Schapire, 2010. A Contextual-Bandit Approach to Personalized News Article Recommendation ,World Wide Web Conference 2010 .

S.-T. Park,D. Pennock, O.Madani,N.Good ,and D.DECoste, 2006. Naive filterbots for robust cold-start recommendations .In Proc.of the 12th ACM SIGKDD International Conf .on Knowledge Discovery and Data Mining ,pp. 699-705.

Hanley, J. A., & McNeil, B. J. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases.Radiology, 148(3), 839-843.

Babcock B, Olston C.2003. Distributed top-k monitoring[C],Proceedings of the 2003 ACM SIGMOD international conference on Management of data. ACM, pp. 28-39.

Faber, N. M., & Rajko, R , 2007. How to avoid over-fitting in multivariate calibration—The conventional validation approach and an alternative.Analytica Chimica Acta, 595(1), pp.98-106.

Arun Murthy.2012,Introducing Apache Hadoop YARN. http://zh.hortonworks.com/blog/introducing-apache-hadoop-yarn/.[accessed 10 March 2014[12]Inokuchi, A., Washio, T., & Motoda, H. 2000. An apriori-based algorithm for mining frequent substructures from graph data. In Principles of Data Mining and Knowledge Discovery pp.13-23. Springer Berlin Heidelberg.

Speriosu, M., Sudan, N., Upadhyay, S. & Baldridge, J. 2011. Twitter polarity classification with label propagation over lexical links and the follower graph. In Proceedings of the First workshop on Unsupervised Learning in NLP .pp53-63. Association for Computational Linguistics.

Ruan, J., & Zhang, W. 2007. An efficient spectral algorithm for network community discovery and its applications to biological and social networks. In Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on . pp. 643-648. IEEE