PRIORITIZING ONLINE TEACHING RESOURCES BASED ON PAGE LENGTH, TIME SPAN AND CURSOR MOVEMENT ALGORITHM USING SEMANTIC WEB

Juhi Agarwal[1], KamalPreet Singh[2], Dr. Neelu Jyoti Ahuja[3]
UPES, Dehradun

Abstract
In today's era internet resources are growing with very fast rate. These resources are very helpful for students in academic courses; especially for computer science courses that need training and practical knowledge. Ontology provides solution to this problem. However, searching suitable resources in the World Wide Web is one of the most difficult and time consuming task. In order to make searching effective, efficient and less time consuming, semantic web technology is used. Ontology is our work is created using Neo4j tool. Ontology is explicit declaration and definition of the properties, interrelationships and types of the entities for a particular domain of discussion. Ontology has a machine-interpretable definition of basic concepts, which makes it very helpful for searching the efficient data. In this paper, efficient page ranking has been proposed for user to assemble e-learning data from the Word Wide Web based on page length, time span and cursor movement for collection of most relevant e-learning resources from the web. This algorithm tends to be very useful in fetching most valuable pages on the top of the result list.

Index Terms: Ontology, Word wide web, Semantic Search.

I. INTRODUCTION

The internet has a large volume of e-learning data and resources available online in the form of text books, audio or video lectures and tutorials, and animations. These online repositories of data are valuable as they make the user understand the concepts swiftly and easily through practical activities. All time access to the resources provides the users flexibility to study as per their time and place convenience. These resources also aid in clarifying the ideas of instructor in an innovative way. Internet is flooded with vast amount of information present online along with lot of irrelevant and noisy data. Thus searching for efficient and effective information from the World Wide Web perhaps is the most difficult and time consuming task in today's era. The crux of the problem lies in searching e-information from the scratch for each subject in a specific domain. Today's search engines, being keyword based, are incapable to search the efficient data and retrieve high-quality of relevant information from Web.

Internet is a source of vast amount of information which is continuously expanding in size day by day. User usually spends a lot of time in searching the data from the web to get the relevant data. It is hectic task for the user to search, filter and extract relevant data from the web. The user feels following difficulties during data mining:

1. Data is voluminous on the web
2. Most of data is semi structured.
3. The web information is dynamic.

So, the process of collecting relevant information from the web is time consuming process. Anyone who wants learning materials from web has to work hard and spend a lot of time as it is labor intensive task. [2]

After the introduction of the Web 2.0, every user has become an author[3]. Nowadays we are able to use Blogs, Wikipedia, YouTube, Social networking sites, and different web

applications. Everyone is having tools and facility to upload their creations on the web and it leads to huge amount of resources in different forms like multimedia and text, available on the web.[4][5]
So, to overpower the effect of expansion of information on the web, semantic abstraction of information is needed [5][6]. Ontology is the explicit formal specification of the terms in a specific domain and relations among them [7]. Additionally, the main barrier to extraction of relevant information from web is lack of efficient algorithm for page ranking.

Usually extended list of documents is generated when we search any information.[8] However, the user spends a lot of time to examine the first ten to twenty web pages or links of them. So an efficient algorithm is needed to rank the documents or web pages according to their relevance to user need [9][10]. Ranking algorithm finds the similarity level of document to user query [11].

Our proposed web page ranking algorithms rank the search results according to the keywords, semantics present in search query and weight factor. We calculate the weight factor using dumping factor and Priority Factor. After that we fetch the top two results and append it into the separate file and by repeating the same steps we get the most relevant data according to the syllabus. So here user can save their time to separately search the data. Searching for a single topic takes a lot of time. Previous algorithms focused on visit links for calculating the weight factor but weighing the page according to the visit only is not sufficient as highly visited page may not have good quality of information whereas time spent on the page is a good factor to get the accuracy.

Our contribution in this paper is:

- Proposed Architecture for collection of most relevant e-learning resources from the web.
- Proposed Ranking Algorithm of Page length, Time Span and Cursor Movement Algorithm.

II. Literature Review

Web page ranking plays an important role in saving the users' time.[12]. When we search any data from the internet, a lot of search results appear. It is quite difficult for the user to visit all the resulted web pages as the user does not have time to analyze all the data available. Moreover, most of data in the internet is noisy and irrelevant for the user. There are different types of ranking algorithms designed for efficient ranking based on Usage based, Content based and Link based.

Anti-spam algorithms:

There are mainly two algorithm used to deal with web spam. A threshold value has been set to differentiate between spam pages and relevant pages. So this algorithm puts the lower ranked or spa, pages to the end of the result [13] [14].

HITS Algorithm:

In Hyperlink-Induced Topic Search (HITS) [15, 16] algorithm, analyzing the textual contents again user query plays an important role to rank the web page. HITS algorithm concentrates on neglecting their textual contents.

Web spam techniques:

A web spam technique has two types: content spam and link spam technique. A content spam technique eliminates the unrelated content searching to improve page ranking and link-based ranking algorithm improves the significance ranking. “honey-pot" is common forms of link spam.[17]

III.PROPOSED ARCHITECTURE

Our proposed web page ranking system is based on the semantic web technology. Graph database, Neo4j is used to create the database. Figure1 shows the graph database of subjects where it stores the information of each and every subject like operating systems, algorithms, compiler design and software engineering etc. Graph database stores the information of each subject and connects every subject using relationship.

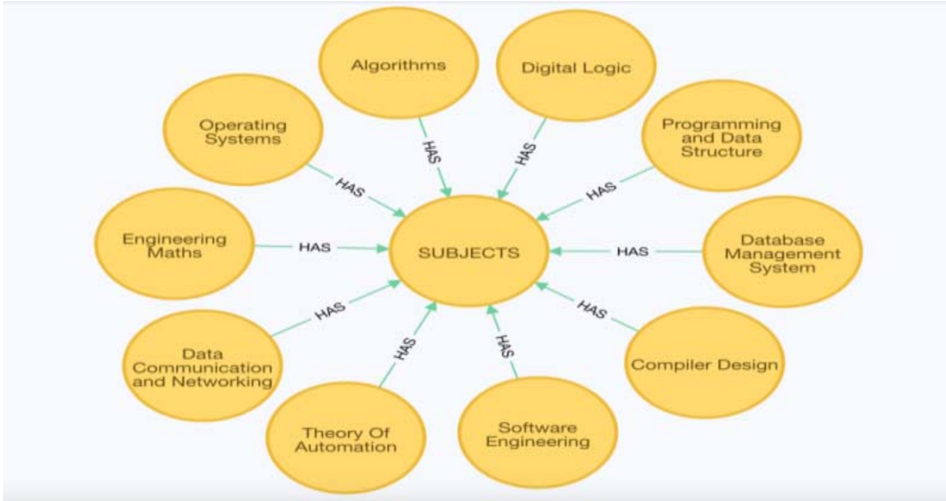

Fig. 1: Graph database of subjects

Figure 2 shows the graph database for database management systems where it stores the all the possible content of the subject in form of graph. Every node of database management systems is connected to each other by relationship has. Some nodes also have sub-nodes.

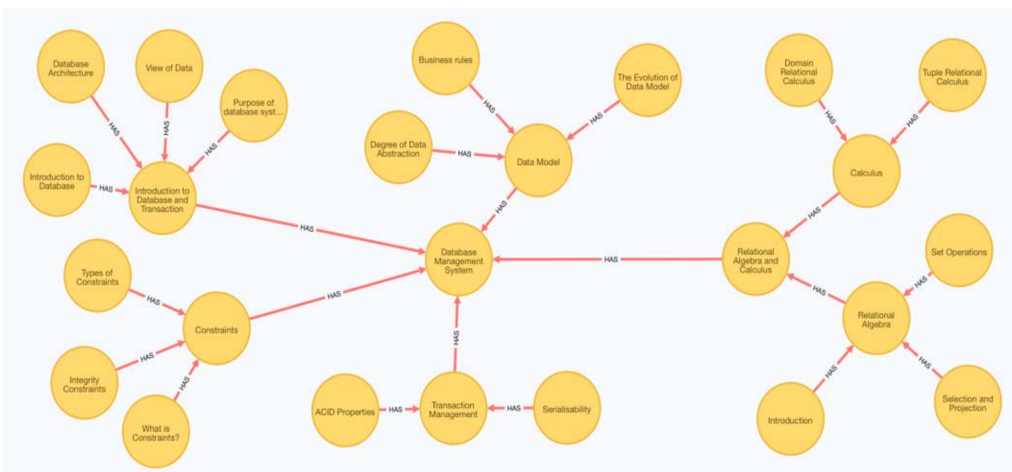

Fig. 2: Graph database for Database management system.

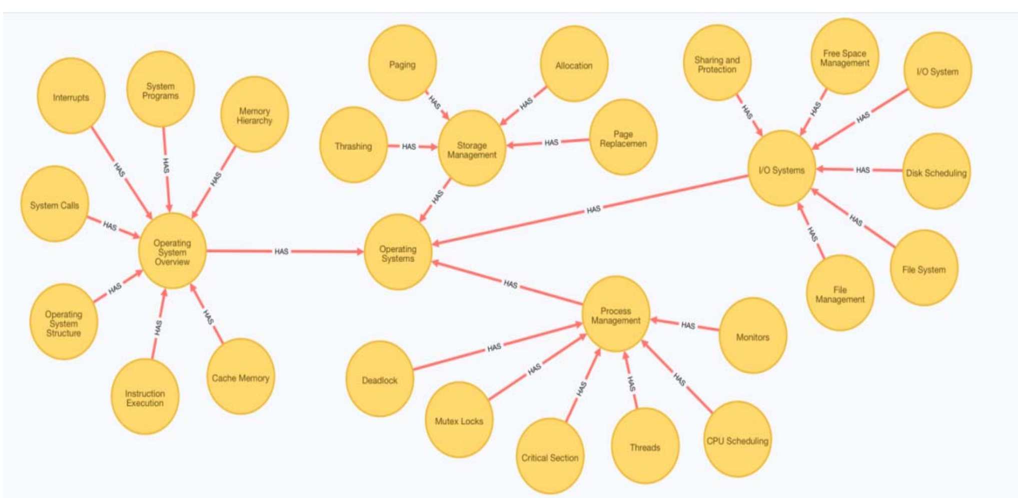

Fig. 3: Graph database for Operating systems.

Figure 4 shows the proposed architecture of the system.

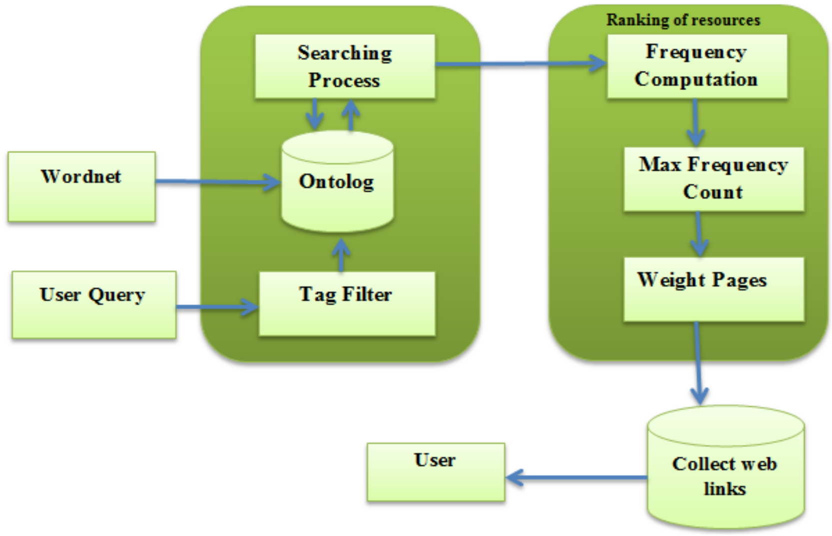

Fig. 4: Architecture of the proposed system for collection of Web pages and ranking system Using Ontology

The organization of the proposed architecture is explained below:

- User Query: User uploads the syllabus.
- Tag filter: Tag filter fetches the tags from the syllabus and starts searching the information from the ontology.
- Ontology: ontology stores the information in the form of graph.
- Wordnet: WordNet is a lexical database for the English language. It groups English words into sets of synonyms.
- Searching process: It searches the data semantically with the help of wordnet.
- Frequency computation: It counts the frequency of each tag in web document or link.
- Maximum frequency count: After counting the frequency of each tag in every web page, we take best two pages, those having max tag count value.
- Weight factor: It calculates the weight of the page. Weight factor depends upon dumping factor and priority factor.
- Collect web links: After applying the page ranking algorithm we get the most relevant results. We choose the separate file to store the selected links.

IV. Proposed Work

Calculating the most relevant pages just by counting the number of visits on page and considering the Time spent on the web pages is not relevant as the time spent on any web page may also be the idle time. Users performing different task with webpage running as background activity also account to time spent on that web page. Therefore these algorithms are not giving accurate results in prioritizing pages.

In our proposed work we describe how to retrieve more relevant web pages from the ontology. The proposed architecture of semantic web searches the required data based on keyword and semantic of the word. The Ontology based framework calculates the total count of keywords and its semantic stored in graph database. According to the keyword, semantic of word count and weight factor, the web pages or data are nominated and ranked.

Relevance Factor =Σ total count of keyword/Semantics * weight factor

Where,

$$\textbf{Weight Factor} = P_f + D_f$$

First of all, keyword and semantic words related to Tags (query) are searched from the web pages and weight factor is calculated according to the Pf and Df. The submission of Pf and Df is weight factor. It is multiplied by total count of keyword and semantic of tags to rank the page.

1. Total count of keyword/Semantics Relevance Factor:

First of all, tags are taken from the syllabus which is uploaded by the user. After that tags are searched using graph database that stores the information in the form of graph. It counts the frequency of the tags.

To get the more accuracy in searched results it calculates the weight factor in different fashion. For high accuracy in relevance factor it counts the total number of words in web page and the time spend in each web page according to the words present in the page. After that it also focuses on the cursor moment. There are three cases which we have focused on while calculating the weight factor:

Case 1: When the time spent on the web page with large word count is considerable but the cursor movement is negligible then page will be having less importance. In this case the dumping factor of the page will have some value thereby decreasing the ranking of the page.

Case 2: Whereas, the web page should be given good ranking (Pf value), when the time spend on the web page with large word count is considerable along with the good cursor movement. In this case the priority factor of the page will have max positive value.

Case 3: Priority factor is also given to the web pages having less word count since the time spend on the page or the cursor movement on web page with smaller word count might be less.

Web pages with smaller word count may have less time spent or the cursor movement. Weight factor considering only the times spend will degrade such page. However, in our algorithm, to avoid starvation of such pages we give some priority factor

2. Weight Table:

Each term in ontology is given some weight. In this weight assigning technique, more weight is assigned to specific terms, i.e. more specific the term is, more is the weight assigned to it. In comparison to the specific terms, the common terms which refer to more than one domain are assigned less weight. The table below shows sample weight table for some terms of a given ontology of the table shown below:

Weight Table:

Factor	Weight
Priority Factor(Pf)	(1-5)positive value
Dumping factor (Df)	(1-5)Negative value

Table 1: Table for calculating the weight factor

3. Relevance Calculation Algorithm:

Algorithm 1:

Algorithm for searching the semantics and keyword for tag (T) from ontology.

In this algorithm we are using tag file Tf as an input, which is a collection of various tags. For each tag in Tf, we are searching the data in our graph database. After getting the relevant url or documents, we store it in a separate page list along with the number of occurrences (Si) of the tag Ti in that particular page.

Input: Tag file Tf

Where Tf $= \sum_{i=0}^{n} Ti$

Ontology file GDb.

Output: Search item LIST.

Read Tf

Read GDb

for each Ti belongs to GDb

∀Ti € GDb

for I = 0 to n

// Where n=no of tags $\sum_{i=0}^{n} Ti = Tf$

{

Sij ← searchwordnet (Ti)

Sij→ jth search item of tag Ti

Sij← searchwordnet (Ti)//searching wordnet to particulars tag and store the result (Semantics+ keyword for that log)

end for

SIL← Resultij

Figure 5: Algorithm for searching the semantics and keyword for tag(T) from ontology.

Algorithm 2: Web page rank algorithm based on page length, time factor and cursor movement

This algorithm is used to rank the pages of the search page list created in algorithm one. Page ranking depends on the number of occurrences of tag and it's semantic in the document as well as on weight factor. The weight factor depends on time spend (Ti) on that document I, length of the document (Li) and movement of cursor (CMi) on document during time Ti. Weight factor consists of dumping factor (Df) and promoting factor (Pf).

.

If considerable time is spend on a particular document and cursor movement tends to 0 then the Df for that particular page will be having some negative value. Along with above two factors, length of the document must also be taken in account to calculate Df because the CMi might not be required if Li is less.

A page must have good ranking if good amount of time spend in the page along with the proper movement of cursor. A good length document satisfying the above two conditions is given a positive Pf value.

Weight Factor= Pf + Df

Algorithm 2: Web page rank algorithm based on page length, time factor and cursor movement.

1. Ti=time spend on that document i.
2. Wc =doc(length)
3. CM=cursor moment on doc i during Ti
4. PRi =[illegible]* Weight Factor
5. // Where Weight Factor = Df+ Pf
6. For each Pi,Df>0
7. If Ti> K && CMi→0 && WCi ≥ t
8. {
9. Else FDi=0
10. For each Pi,PFi>0
11. If Ti>P && CMi>q && WC>r
12. Else
13. PFi=0
14. Arrange page in increasing order of PRi in LIST
15. Sort(pages Pi)
16. Choose top 2
17. Repeat this for all Sij

V. EXPERIMENT AND RESULTS

The total number of pages be N.
Let the Time P_T(time spend on each page) for all N pages is greater than threshold time T_T .
T_T is the minimum time required for the to fall under good category.
F denotes the number of pages with no or negligible cursor movements. i.e these pages should not be given a good rank.

If F=0; performance of CM+T=T.

If F starts increasing, performance of T algorithm degrades gradually as it only considers time factor not the cursor movement.

The performance of CM+T algorithm remains same as it differentiates F pages from good ranking pages.

Let Pl is the length of the page and P_{time} denotes the time user spends on the page.

CASE 1: Pl is large and P_{time} is less. In this case the page rank should degrade.
CASE 2: Pl is less and P_{time} is also less than the threshold value in this case much cannot be decided about the page priority. Hence the page ranking remains unchanged.

CASE 3: Pl is large and P_{time} is also large, in this case we also need to consider cursor movement as discussed in algorithm 1.

- **3.1** : If cursor movement is large, the page rank should increase.
- **3.2** : If cursor movement is less, the particular threshold value, the page is of less priority hence the ranking of the page should decrease.

Fig. 5 compares the performance of T and CM+ T. Considering the total number of pages to be 100, given no faulty pages, the performance of both the algorithms is same. However, as the number of faulty pages F starts increasing, the efficiency of T decreases as this algorithm doesn't takes into account the cursor movement whereas the performance of CM+ T still remains unaffected.

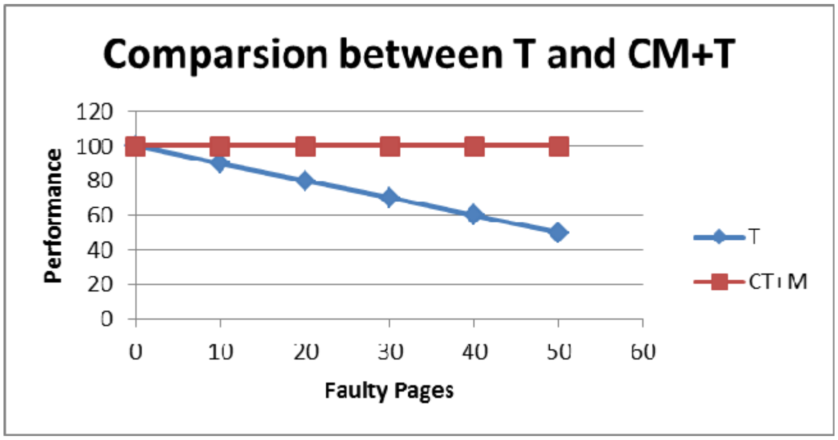

Fig. 5: Comparison between T and CM+T

VI .CONCLUSION

The proposed algorithm gives the most relevant results among various ranking algorithms. Because previous algorithms are totally based on counting the number of visits with Time Factor on the web pages which is not efficient approach as user may be idle or doing some other work in another tab. In our proposed algorithm we have also focused on Cursor Movement including Time Factor and dumping factor. Cursor Movement is must to check that user is active or not. According to the page length, time spends and Cursor Movement we are calculating the most relevant web page as compared with the traditional methods.

VI. REFERENCES

[1] Casey and M. McAlpine, "Writing and using reusable ducational materials: a beginner's guide," CETIS Educational Content Special Interest Group Publication, 2002.

[2] Shital C Patil,RR keole Content and usage based ranking for enhancing search engine delivery 2014 in volume 3 issue 7International journal of Science and Research (IJSR)

[3] E. E. Bachari, E. H. Abelwahed, and M. E. Adnani, "An adaptive teaching strategy model in e-learning using learnr's: LearnFit framework," International Journal of Web Science, vol. 1, no. 3, pp. 257-274, 2012.

[4] K. L. Chin and E. Chang, "A sustainable ICT education ontology," in Proc. 5th IEEE International Conference on Digital Ecosystems and Technologies, Korea, pp. 350-354, 2011.

[5] Vandana Dhingra, Apeejay Stya University, India & Komal Bhatia, YMCA University of Science and Technology, India, "Comparative Analysis of Ontology Ranking Algorithms", International Journal of Information Technology and Web Engineering, 7(3), 55-66, July-September 2012

[6] Yajun Du and Yufeng Hai. "Semantic ranking of web pages based on formal concept analysis". Journal of Systems and Software, 86(1):187–197, 2013.

[7] M. Sehgal et al. 2014. Enhanced Page Rank Algorithm Using Time Factor. Int. J. Eng. and Comp. 3 (July 2014), 6990-6995.

[8] Lissa Rodrigues, Shree Jaswal, "Hybrid Model for Improvised Page Ranking Algorithm", Hybrid Model for Improvised Page Ranking Algorithm", 978-1-4673-9825-1/15/20 15 IEEE.

[9] Fayyaz Ali, Irfan Ullah, Shah Khusro, "An Empirical Investigation of PageRank and Its Variants in Ranking Pages on the Web", 978-1-5090-5300-1/16 $31.00 © 2016 IEEE.

[10] P. Patel. 2014. Research of Page ranking algorithm on search engine using Damping factor. Int. J. Ad. Eng. and Res. Dev. 1 (Feb. 2014), 1-6.

[11] Singh, C. and Kautish, S. "Page Ranking Algorithms for Web Mining: A Review" International Journal of Computer Applications (0975 –8887), International Conference on

Advancements in Engineering and Technology (ICAET 2015).

[12] P. K. Sharma and Sanjay Tiwar, “A Noval Approach For Web Ranking Based On Weight of Links” International Journal on Recent Trends in Computing & Communication (IJRITCC) 2015, PP 5268-5272.

[13] Orfanou, K., Tselios, N., & Katsanos, C. (2015). Perceived usability evaluation of learning management systems: empirical evaluation of the system usability scale.

[14] S. Setayesh and A. Harounabadi et al. 2014. Presentation of an Extended Version of the Page Rank Algorithm to Rank Web Pages Inspired by Ant Colony Algorithm. Int. J. Comp. App. 85 (Jan. 2014), 975 – 8887.

[15] G. Kumar, N. Duhan and A. K. Sharma “Page Ranking Based on Number of Visits of Links of Web Page” Computer and Communication Technology (ICCCT), 2011 2nd International Conference on ©2011 IEEE, DOI 10.1109/ICCCT.2011.6075206 Page(s): 11 – 14.

[16] Tuhena Sen, Dev Kumar Chaudhary et. Al 2016 “Contrastive study of Simple PageRank, HITS and Weighted PageRank algorithms: Review” in IEEE proceedings of Cloud Computing, Data Science & Engineering - Confluence, 2017 7th International Conference on 12-13 Jan. 2017.

[17] Nidhi Shalya, Shashwat Shukla and Deepak Arora,” An Effective Content Based Web Page Ranking Approach”, International Journal of Engineering Science and Technology (IJEST) , Vol. 4 No.08 August 2012.

[18] Dr. Daya Gupta, Devika Singh, “User Preference Based Page Ranking Algorithm” ISBN: 978-1-5090-1666-2/16/2016 IEEE.

[19] Safaa I. Hajeer, Rasha M. Ismail, Nagwa L. Badr, M. F. Taiba “An Efficient Hybrid Usage-Based Ranking Model for Information Retrieval Systems and Web Search Engines” . in IEEE proceedings of 2015 6th International Conference on Information and Communication Systems (ICICS).

[20] Wei Huang and Bin Li, “An Improved Method for the Computation of PageRank” International Conference on Mechatronic Science, Electric Engineering and Computer August 19-22, 2011, Jilin, China.