2017 International Conference on Computational Science and Computational Intelligence

Tag Ranking Multi-Agent Semantic Social Networks

Rushdi A. Hamamreh
Computer Engineering Department,
Faculty of Engineering.
Al-Quds University.
rhamamreh@eng.alquds.edu

Sameh Awad
Computer Engineering Department,
Faculty of Engineering.
Al-Quds University.
sameh.awad@student.alquds.edu

***Abstract*—Social Media has become one of the most popular platforms to allow users to communicate, and share their interests without being at the same geographical location. With the rapid growth of Social Media sites such as Facebook, LinkedIn, and Twitter, etc. There is vast amount of user-generated content. Thus, the improvement in the information quality has become a great Challenge to all social media sites, which allows users to get the desired Content or be linked to the best link relation using improved search / link technique. So introducing semantics to media networks will widen up the representation of the social media networks. Semantic Social Networks representation of social links will be extended by the semantic relationships found in the vocabularies which are known as (tags) in most of social media networks.**

This paper proposes a new model of semantic social media networks from the perspective of multi-agent systems. The multi-agent system is composed of two main functionalities: semantic indexing and tag ranking.

***Index Terms*—Social Media, Semantic Social Ranking, Big Data, Multi-Agent Systems, Semantic Indexing, Tag Rank, LDA.**

I. Introduction

THE Social media are emerging field in information interchange, worldwide used and wanted. It is a challenging subject to do a research in social media field as it was and still affecting us in every aspect of our lives. The improvement in retrieved contents in social media should be given attention as it reflects the quality and integrity of social media in general. The new perspective was to introduce semantics into social network to get Semantic Social Network in which relations and social graph will be composed according to the words meanings, especially keywords that are widely known as Tags in the social media network.

In current social media networks, Links between contents are constructed by many ranking techniques according to the way to deal with data and importance and priority of data. Such as posts in Facebook, hashtags in Twitter, Job and Experiences in LinkedIn,..Etc. and so data must be ranked in a way that links constructing the social graph will reflect natural distribution and connection between nodes of the social media. Rank of each node is given by making iterative process of weights in network. In Semantic Social Networks, this weight can be given according to semantic content of the social media node.

Semantic Content of Semantic Social Network, which is large and complex collections of data and that, is known nowadays as "***Big Data***" must be indexed before ranking process. Introducing semantic indexing algorithms to process content of Semantic Social Networks can achieve this. Improving indexing output and choosing the proper ranking algorithm will affect the quality of the social graph and how nodes will be linked in semantic social network.

The existence of various ranking algorithms depending on how dealing with content which affects the quality of the output of the ranking. Therefore, the ranking of contents in social media should be based on some criteria that reflects related topics or links to the content. This can be achieved by depending on semantic indexing algorithms that gives the actual relations depending on the topic of the contents.

For Indexing and Ranking processes. The Concept of Multi-Agent system is a great addition to give good, improving, and self-learning mechanism especially in social networks. Multi-Agent Systems are computerized system composed of multiple interacting intelligent agents within an environment which can be used to solve problems.

An agent is a computer system that is capable of independent action on behalf of its user or owner (figuring out what needs to be done to satisfy design objectives, rather than constantly being told).

To improve the output of ranking, improve the performance of indexing and ranking agents to get autonomous semantic social network linking and building of social graph. Improving indexing should be introduced by enhancement in algorithm to be applied in indexing process. Moreover, improving ranking must be met by semantic content analysis that makes the linking similar according to subjects or keywords on the social media content. In addition, processing time must be taken in consideration.

II. Ranking Algorithms

Ranking is a link analysis algorithm and it assigns a numerical weighting to each element of a hyperlinked set of documents, such as the World Wide Web, with the purpose of "measuring" its relative importance within the set. The top popular algorithms used in social media are:

A. Page Rank (PR):

Commonly in Google, in Page Rank if a page contains important links towards it then the links of this page towards the other page are also to be considered as important pages.

And the Page Rank considers the back link in deciding the

978-1-5386-2652-8/17 $31.00 © 2017 IEEE
DOI 10.1109/CSCI.2017.156

rank score.

PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that websites that are more important are likely to receive more links from other websites. The Page Rank considers the back link in deciding the rank score. So assume we have two pages $\boldsymbol{u}$ and $\boldsymbol{v}$:

$$\boldsymbol{PR(u)} = \sum_{\boldsymbol{v \in B_u}} \frac{\boldsymbol{PR(v)}}{\boldsymbol{L(v)}} \quad (1)$$

Where $\boldsymbol{B_u}$ is the set of all pages linking to page $\boldsymbol{u}$. And $\boldsymbol{L(v)}$ is the number of links from page $\boldsymbol{v}$.

Considering damping factor the page rank will be:

$$\boldsymbol{PR(u)} = \boldsymbol{(1-d)} + \boldsymbol{d} \sum_{\boldsymbol{v \in B_u}} \frac{\boldsymbol{PR(v)}}{\boldsymbol{L(v)}} \quad (2)$$

This rank algorithm does not reflect the content of pages but it concentrate on the number of links related to a page.

B. Weighted Page Rank (WPR):

Weighted page rank algorithm (WPR) is the modification of the original page rank algorithm. WPR decides the rank score based on the popularity of the pages by taking into consideration the importance of both the in-links and out-links of the pages. This algorithm provides high value of rank to the more popular pages and does not equally divide the rank of a page among its out-link pages. Every out-link page is given a rank value based on its popularity. Popularity of a page is decided by observing its number of in and out links. For example for the pages $\boldsymbol{u}$, $\boldsymbol{p}$ and $\boldsymbol{v}$ the weight rank is:

$$\boldsymbol{W^{in}_{(v,u)}} = \frac{\boldsymbol{I_u}}{\sum_{\boldsymbol{p \in R(v)}} \boldsymbol{I_p}} \quad (3)$$

Where $\boldsymbol{I_u}$, $\boldsymbol{I_p}$ are numbers of in-links of pages $\boldsymbol{u}$ and $\boldsymbol{p}$, $\boldsymbol{R(v)}$ reference page list of page $\boldsymbol{v}$

$$\boldsymbol{W^{out}_{(v,u)}} = \frac{\boldsymbol{O_u}}{\sum_{\boldsymbol{p \in R(v)}} \boldsymbol{O_p}} \quad (4)$$

Where $\boldsymbol{O_u}$, $\boldsymbol{O_p}$ are numbers of out-links of pages $\boldsymbol{u}$ and $\boldsymbol{p}$.

C. Edge Rank

Is the name commonly given to the algorithm that Facebook uses to determine what articles should be displayed in a user's News Feed. Every action their friends take is a potential newsfeed story. Facebook calls these actions "Edges." That means whenever a friend posts a status update, comments on another status update, tags a photo, joins a fan page, or RSVP's to an event it generates an "Edge," and a story about that Edge might show up in the user's personal newsfeed.

It would be completely overwhelming if the newsfeed showed all of the possible stories from your friends. Therefore, Facebook created an algorithm to predict how interesting each story will be to each user. Facebook calls this algorithm "EdgeRank" because it ranks the edges. Then they filter each user's newsfeed to only show the top-ranked stories for that particular user. The general equation of this algorithm is:

$$\boldsymbol{Edge\ Rank} \sum = \boldsymbol{u_e} * \boldsymbol{w_e} * \boldsymbol{d_e} \quad (5)$$

Where $\boldsymbol{u_e}$ is the affinity score (between viewing users and edge creator). $\boldsymbol{w_e}$ *Weight for the edge type (create, comment, like, tag, etc.)* and $\boldsymbol{d_e}$ *time decay factor.*

D. Tag Rank

Tag Rank is new suggested technique that is similar to page rank but it works on tags and links between nodes according to existence of tag in contents of social media.

This algorithm digs the annotation behavior of the web users, calculates the heat of the tags. By using time factor of the new data source tag and the annotations behavior of the web users. It can response the true quality of tags more externally and improve the veracity of page ranking. This algorithm provides a better authentication method for ranking the web pages. The results of this algorithm are very accurate and this algorithm index new information resources in a better way.

In Simple way, as we get the semantic index by indexing agent, the ranking agent job is to build Tag-Pair Weight Matrix (TWM) as a rank matrix depending on the indexing result.

The mathematical model of the TagRank is:

First creating TFM which is (Tag Frequency Matrix) which is the sum of Tag Matrices TM depending on tag simultaneous appearance:

So $\boldsymbol{TM_{(i,j)} = 0}$: $tag\ i\ and\ tag\ j$ do not appear simultaneously on certain content.

And $\boldsymbol{TM_{(i,j)} = 1}$: $tag\ i\ and\ tag\ j$ appear simultaneously on certain content.

And so Tag Frequency Matrix is

$$\boldsymbol{TFM_{(i,j)}} = \sum_{\boldsymbol{k=1}}^{\boldsymbol{m}} \boldsymbol{TM_k(i,j)} \quad (6)$$

Lastly, the Tag-Pair Weight Matrix will be:

$$\boldsymbol{TWM_{(i,j)}} = \boldsymbol{TSM_{(i,j)}} \times \boldsymbol{TFM_{(i,j)}} \quad (7)$$

Where $\boldsymbol{TSM_{(i,j)}}$ is an entry of tag-pair similarity matrix.

Based on semantic social network perspective, we find that Tag Rank is the best option to go on with in our proposed model.

III. Indexing Algorithms

Indexing algorithms -generally in search engines- collects, parses, and stores data to facilitate fast and accurate information retrieval. Index design incorporates interdisciplinary concepts from linguistics, cognitive psychology, mathematics, informatics, and computer science. An alternate name for the process in the context of search engines designed to find web pages on the Internet is web indexing.

Popular engines focus on the full-text indexing of online, natural language documents. Media types such as video, audio and graphics are searchable.

To get the best result in indexing semantic indexing algorithms introduced to get actual index of contents of the content of social media.

Many Algorithms of semantic indexing were developed to reflect actual overview of semantic content of the documents, pages and other contents of social media. Such as:

A. TF-IDF (Term Frequency –Inverse Document Frequency)

It is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, but is often offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general. The mathematical equation for this technique is:

$$\boldsymbol{tf - idf_{t,d}} = \boldsymbol{tf_{t,d}} \times \boldsymbol{idf_t} \quad (8)$$

Where $\boldsymbol{tf - idf_{t,d}}$ is the score between query $\boldsymbol{t}$ and document $\boldsymbol{d}$. $\boldsymbol{tf_{t,d}}$ is the term frequency and $\boldsymbol{idf_t}$ is the Inverse Document Frequency.

Nowadays, tf-idf is one of the most popular term-weighting schemes. Variations of the tf–idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query

B. LSI (Latent Semantic Indexing)

It is another technique in natural language processing to discover information about the meaning behind words. LSA analyzes relations between set of documents and terms they contain and assume that words that are close in meaning will occur in similar pieces of text. Then LSI constructs matrix of words (terms) per document, and using singular value decomposition to reduce and divide the big matrix into small orthogonal components. To finally represent vectors of words in documents.

C. PLSI (Probabilistic Latent Semantic Indexing)

Is a statistical technique for analysis of co-occurrence data. Compared to standard latent semantic analysis which stems from linear algebra and downsizes the occurrence tables (usually via a singular value decomposition), PLSA is based on a mixture decomposition derived from a latent class model. Instead of matrices, PLSI uses probability methods to represent semantic. Instead of using matrices, (PLSA) uses a probabilistic method. Its graphical model is as

$$\boldsymbol{P(w|d)} = \boldsymbol{P(d)} \sum_{\boldsymbol{c}} \boldsymbol{P(c|d)P(w|c)} \quad (9)$$

Where $\boldsymbol{d}$ is the document index, $\boldsymbol{c}$ is word's topic drawn from $\boldsymbol{P(c|d)}$, and $\boldsymbol{w}$ is word drawn from $\boldsymbol{P(w|c)}$. And both $\boldsymbol{P(c|d)}$ and $\boldsymbol{P(w|c)}$ are modeled as multinomial distributions.

D. LDA (Latent Dirichlet Allocation)

is an improvement of PLSI by generalizing it using Dirichlet Prior as a variable reflects normal distribution of words in documents.

LDA is a mixture model. It assumes that each document contains various topics, and words in the document are generated from those topics. All documents contain a particular set of topics, but the proportion of each topic in each document is different.

The generative process of the LDA model can be described as follows:

1- Choose a multinomial distribution φ_z for each topic z from a Dirichlet distribution with parameter β
2- For each document d choose a multinomial distribution θ_d from a Dirichlet distribution with parameter a
3- For each word token w in document d Choose a topic $z \in \{1, \dots K\}$ from the multinomial distribution θ_d
4- Choose word w from the multinomial distribution $\emptyset_z$
5- Thus the likelihood of generating corpus is:

$$\boldsymbol{P(Doc_1, \dots, Doc_N|\alpha, \beta)} = \iint \prod_{z=1}^{N} \boldsymbol{P(\varphi_z|\beta)} \prod_{d=1}^{N} \boldsymbol{P(\theta_d|\alpha)} \left(\prod_{i=1}^{N_d} \sum_{z_i=1}^{K} \boldsymbol{P(z_i|\theta)P(w_i|z,\varphi)}\right) \boldsymbol{d\theta d\varphi} \quad (10)$$

As previous researches compared between semantic indexing algorithms LDA was the best according to the quality of output, which can be measured by perplexity, log-likelihood, precision and recall.

In this paper, we will use this algorithm and we will find the best output of LDA indexing process based on modifying the parameters affecting the result $(a, \beta \; and \; K)$.

IV. Proposed Model

A. System Architecture

Our proposed model consists of the following:

-Documents: which are sets of raw data from social networks to be processed.

-Indexer (Indexing Agent): In this part, three main processes are carried out; initializing documents, parsing document and indexing using semantic indexing algorithm.

-Index: it is the output of the document after indexer agent job completes. It contains the topic probabilities per document. And the word probability per topic. Topics are taken to be Tags for the first time. To be processed in ranking process.

-Ranking Agent: In this part, we get the probabilities of topics per document. Then process them to be ranked as tags. Using certain ranking algorithm.

-Social Graph: the output of ranking agent will be used to build links between social nodes. Figure1. Shows how this model is composed.

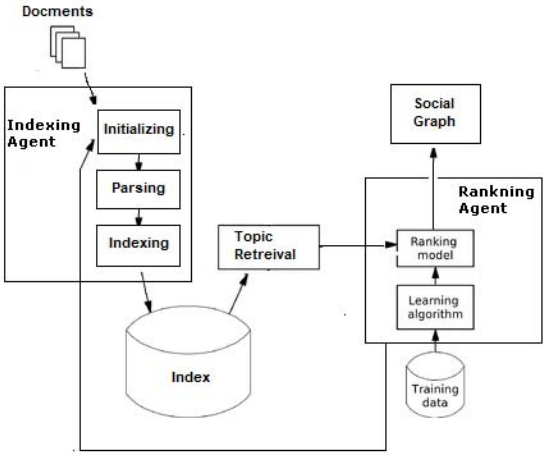

Figure1: System Architecture.

B. Algorithm

In the beginning, data in documents is collected and then parsed.

After that, Indexing Agent uses LDA algorithm starts to build the index of semantics, the result index that contains the topics and the most common word in each topic will be used to get the tags. Finally, Ranking Agent builds rank matrix of Tags to give the social graph link network. See the next Block for Algorithm1. MSSNT, which is abbreviation of (**M**ulti-agent **S**emantic **S**ocial **N**etwork **T**agRank) algorithm that show how agents work.

Algorithm1. MSSNT

Input: Document Collocation Dataset
Start
//Indexing Agent{
Rule 1: Get Document
Rule 2: Parse Document Content
for *i*=1 to *n* **do**
Rule 3: Start LDA Indexing Algorithm
end for
Output Index $(\theta_{t_1}, \theta_{t_2}, \dots, \theta_{t_n})$
end } //end of indexing agent job

//Ranking Agent{
Start
for *j*=1 to *n* **do**
//repeat until all tags which have larger ranks than threshold $\boldsymbol{\tau}$
Repeat{
//select tag 1 and tag 2 which are columns and rows of Max$(\theta_{t_j}, \theta_{t_{j+1}})$
Select Max$(\theta_{t_j}, \theta_{t_{j+1}})$
Condition: While (Max$(\theta_{t_j}, \theta_{t_{j+1}}) \geq \tau$) { // τ is threshold
Select Max$(\emptyset_{w_j}, \emptyset_{w_{j+1}})$
}
j=*j*+1;
} // until (all tags which are larger thanτ processed).
Build Links between Tags
end} //end of Ranking Agent job.
Output social graph

V. Simulation Results

In this section, we will do brief test on both algorithms to show how the system will work.

For LDA we used the Gibbs Sampling and its LDA model. The dataset is used was *psychreview* dataset. Which contains Psychology Review Abstracts and collocation Data.

Using Matlab, we tried to check what values of

We use perplexity and log-likelihood as the criteria to choose the three effective parameters in the LDA algorithm which are: α, β, and K.

Perplexity is a measurement of how well a probability distribution or probability model predicts a sample. It may be used to compare probability models. A low perplexity indicates the probability distribution is good at predicting the sample.

Log-likelihood is the natural logarithm of the likelihood function, called the log-likelihood. Likelihood function is a function of the parameters of a statistical model given data. Likelihood functions play a key role in statistical inference, especially methods of estimating a parameter from a set of statistics.

Log-likelihood is more convenient to work with. Because the logarithm is a monotonically increasing function, the logarithm of a function achieves its maximum value at the same points as the function itself, and hence the log-likelihood can be used in place of the likelihood in maximum likelihood estimation and related techniques.

Best choices of α and K have strong correlation, which chooses parameters as $\alpha = 50/K$, $\beta = 0.1$. For both datasets, we have the true label of every document. Therefore, an intuitive guess of K would be the number of classes. Grid search for all three parameters requires too much work, so we first fix K as the number of classes, and apply grid search of $\alpha$and β. After deciding best α and β for fixed K, we then search for best K with fixed β and changing α,under the assumption that $\alpha \propto 1/K$

.

In simulation we select $\alpha = \{\frac{0.01}{K}, \frac{0.1}{k}, .., \frac{1}{k}\}$

And $\beta = \{0.1, 1, 2\}$

And K={2, 3, 4, 5,6}

First, we fix the values of to find the best value of K. The results was as shown on Figure-2.

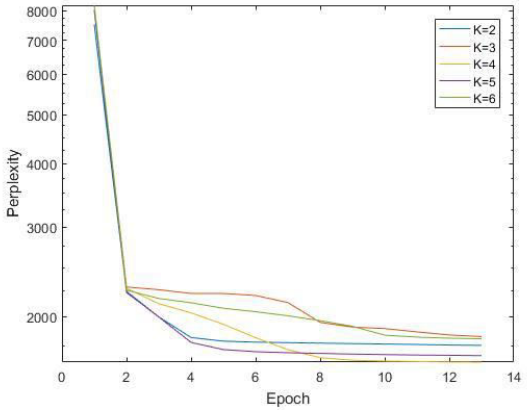

Figure2: Perplexity with changing Topic Classes number (K)

As shown in Figure2. When K=4 we have the minimum perplexity, which means when we classify collocation into 4 topic classes we get the best results in indexing.
With fixing K value to 4 and testing α and β we get the effective parameters where test results shows that $\alpha = 0.7/K$ and $\beta = 0.1$.
The Output of the Indexing process done by the Indexer Agent is used as input for Ranking Agent.

Using MATLAB tag rank is designed as finding matrix of maximum product between tags. And we set that links between tags must be when probability of tag in document is more than 0.4 as a threshold. The output for the first 50 documents of the collocation are as shown in Figure3.

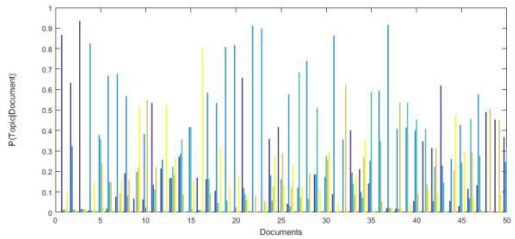

Figure3. Topic per Document distribution..

It is seen in Figure3. That the topics per document distribution is representing all the possibilities of linking documents to each other. However, after using TagRank to filter links according to specific threshold (τ =0.4) links will be as shown in Figure4. That means Tags links are constructed among this criterion.

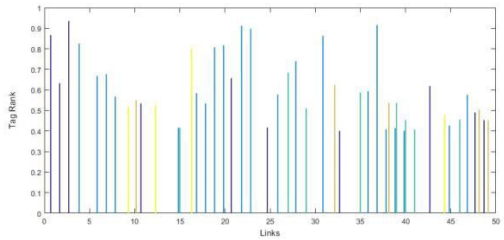

Figure4: Tag Rank Linking distribution

VI. Conclusion and Future Work

In this paper, we propose new model of social networks depending on semantics, with using semantic indexing methods and rank algorithms. In addition, show in test how this idea will be implemented.
In the Future, more modification on Tag rank algorithm needed. Also, further improvement of LDA to be observed.

References

[1] Obar, Jonathan A., Wildman, Steve. *Social media definition and the governance challenge: An introduction to the special issue.* Telecommunications policy. 39 (9): 745–750. doi:10.1016/j.telpol.2015.

[2] Stephen Downes. *The Semantic Social Network***.** February 14, 2004.

[3] Wooldridge, Michael. *An Introduction to MultiAgent Systems*. John Wiley & Sons. (2002) p. 366. ISBN 0-471-49691-X.

[4] Kubera, Yoann; Mathieu, Philippe; Picault, Sébastien. *Everything can be Agent.* Proceedings of the ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'2010), Toronto, Canada: 1547–1548.

[5] Boyd, Dana; Crawford, Kate. *Six Provocations for Big Data*. Social Science Research Network: A Decade in Internet Time: Symposium on the Dynamics of the Internet and Society. (September 21, 2011). doi:10.2139/ssrn.1926431.

[6] Mathijs deWeerdt, Yingqian Zhang, Tomas Klos, *Multiagent Task Allocation in Social Networks.* Autonomous Agents and Multi-Agent Systems, 2012, Volume 25, Number 1, Page 46

[7] Wasserman, S., Faust, K. *Social network analysis: methods and applications*. Cambridge University Press, Cambridge (UK) (1994).

[8] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*. Cambridge University Press. 2008.

[9] S.Prabha, K.Duraiswamy,J.Indhumathi, *A Comparative Analysis of Different Page Ranking Algorithms*. International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:8, No:8, 2014

[10] http://edgerank.net/ accessed in September 1, 2017.

[11] S. Jie, C. Chen, Z. Hui, S. Rong-Shuang, Z. Yan and H. Kun. *TagRank: A New Rank Algorithm for Webpage Based on Social Web*. 2008 ZTechnology, Singapore, 2008, pp. 254-258.

[12] DaeHoon Hwang, *Comparison and Evaluation of Highly Related Tag-Pairs Extraction Methods*. International Journal of Multimedia and Ubiquitous Engineering Vol.10, No.9 (2015).

[13] Dae-Hoon Hwang, *Design of Tag Ranking Algorithm Based on Cluster*. Advanced Science and Technology Letters Vol.133 (Information Technology and Computer Science 2016), pp.194-200.

[14] Scott Deerwester; Susan T Dumais; George W Furnas; Thomas K Landauer; Richard. *Indexing by Latent Semantic Analysis*. Journal of the American Society for Information Science (1986-1998); Sep 1990.

[15] Hofmann, Thomas. *Probabilistic Latent Semantic Indexing*. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1999.

[16] Blei, David M.; Andrew Y. Ng; Michael I. Jordan. *Latent Dirichlet Allocation* . Journal of Machine Learning Research. 3: 993–1022. doi:10.1162/jmlr. 2003.3.4-5.993.2003.

[17] Wang, Y., Lee, J.-S. and Choi, I.-C. *Indexing by Latent Dirichlet Allocation and an Ensemble Model*. Journal of the Association for Information Science and Technology, 67: 1736–1750. doi:10.1002/asi.23444. 2016.

[18] Rushdi Hamamreh. Intelligent Focused Agent for Building Databases from Distributed Web Systems, International Conference on Computer Science and Applications 2008, San Francisco, USA, 22-24 Oct. 2008.