DEIM Forum 2013 A3-4

探索的検索ためのテキスト景観

趙　夢† 大島 裕明† 田中 克己†

† 京都大学大学院情報学研究科社会情報学専攻 〒 606–8501 京都府京都市左京区吉田本町
E-mail: †{zhao,ohshima,tanaka}@dl.kuis.kyoto-u.ac.jp

Text Landscape for Exploratory Search

Meng ZHAO†, Hiroaki OHSHIMA†, and Katsumi TANAKA†

† Department of Social Informatics, Graduate School of Informatics, Kyoto University
Yoshida Honmachi, Kyoto, 606–8501 Japan
E-mail: †{zhao,ohshima,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract With the help of state-of-the-art information retrieval technologies, most relevant documents come to the very top of ranked list in the reponse of user's query. They are well performed for navigational search. However, for informational search, or more general case, exploratory search, whose search process is iterative, multi-tactical, spanning multiple queries or search sessions, they don't seem to be effective enough. One reason is that "relevance" is considered to be the most crucial feature rather than"relatedness" which contains more, such as rival information. In this work, we introduce a method to present the whole landscape view of related information according to a user-indicated document in order to show a complete knowledge environment.

Key words text landscape, exploratory search, text similarity, text adjacency

1. Introduction

With the help of state-of-the-art information retrieval technologies, most relevant documents come to the very top of the ranked list in the response of user's query. For example, Fig.1 shows the top search result under the query "iphone 5" returned by Google (注1). If our goal is to find the official web site of "iphone 5", then the first page of Apple iphone 5 (注2) gives us the exactly right answer. In another case, if our goal is to learn something about "iphone 5", the first page may be one of our choices, but obviously it is not enough. It is likely that we will click and read several pages to satisfy this information need. Or even sometimes we would reformulate our query in order to get a better result. It is conceivable that this kind of search covers several search session. As a result, it will cost user much time to complete a whole search process. And most of the time is spent to read and eliminate irrelevant information. Another example is when we issue a search query to Google, if the input query is fortunately well-formed to retrieve the target, it will help us reach the target within a short time. However, well-formed queries are not always easily created. In this case, how we can achieve our target or potential goal, is an important issue. Nowdays, query suggestion is widely used for guidance in the information space. We think it is an indirect guidance since it only offers users possible search directions. Still the "iphone 5" example. Google recommends queries such as "iphone 5 cases", "iphone 5 reviews", "iphone 5 accessories" and so forth. But it does not show user feasible relation between "iphone 5 cases" and "iphone 5 accessories", while user's potential goal is hidden in such relations.

Why do searches bog down in those cases? The key reason is the state-of-the-art information retrieval technologies do not show searchers the "surrounding information" about the query topic. Here so-called "surrounding information" is referred to as related documents of the query topic, containing but not limited to rival documents, documents of superordinate concepts. By showing searchers the "surrounding information" and their relations in a more organized way, we

(注1)：http://www.google.com/
(注2)：http://www.apple.com/iphone/

図 1　A search example by Google.

believe not only the satisfaction of searcher experience but also the efficiency to locate search goal will be substantially improved.

In this paper, we propose a novel kind of information search that given a web page of search result of a query, a "text landscape" (See detials in Section 3.) of this query will be returned. In general, "text landscape" provides a whole knowledge environment of the query topic, composed mainly by the "surrounding information". In order to discover documents or web pages which belongs to the same landscape, we incorporate adjacency into our scheme.

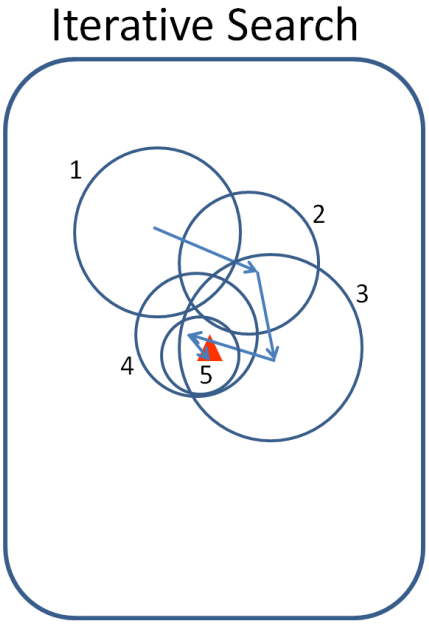
The remainder of the paper is organized as follows: Section 2. introduces some related works; in Section 3., the definition of "text landscape" is stated; in Section 4., the basic idea and the problems are addressed; Section 5. explains our proposed method PanoramaRank for text landscape; finally, conclusions and possible directions for future work are stated in Section 6..

2. Related Work

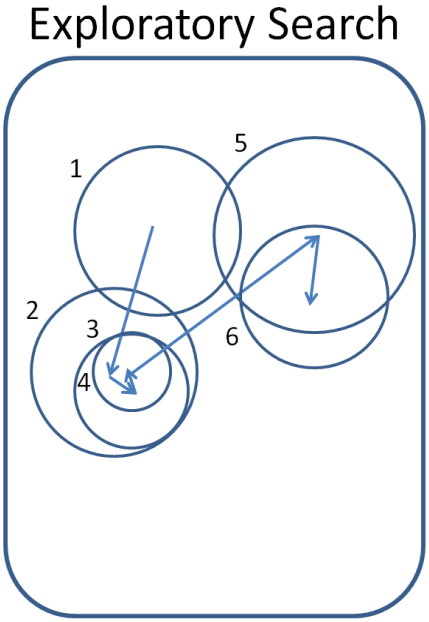
Some researches are devoted to classify web searches. In [6], Broder stated there are three types of web queries: "navigational", "informational" and "transactional". Usually there is only one right result in the case of navigational queries. User makes an effort to find a particular web page which has in his mind before doing a search. While in the case of informational queries, user tries to find information about a topic, but with no further interaction predicted. Transactional searches are intended to reach a site including some further interactions. Broder found that about 48 percents of queries belong to informational, 30 percents belong to transactional, and only 20 percents belong to navigational. In [7], Rose and Levinson proposed a slightly different classification method which is based on why people are searching rather than how they search and what they are searching for. In their framework, "resource" category replaces Broder's notion of transactional queries. Both [6] and [7] reach an agreement about what "navigational" query is. However, Rose and Levinson believed in their "informational" queries, there include some further interactions, since the user's goal is to obtain information about the query topic. The completely different one, "resource" queries are intended to get something, such as download a song, view a video. In their experiment, it was pointed out that around 60 percent of queries are "informational". This leads to the conclusion over 75 percents of user queries are non-navigational ones.

Exploratory search is defined to be used to describe an information-seeking problem context and information-seeking processes in [13]. Especially, the information-seeking problem context is "open-ended, persistent, and multi-faceted" and processes are "opportunistic, iterative, and multi-tactical". It subtly differes from information retrieval (IR): in IR, the search target is usually known before the user query is issued. In contrast, in information seeking, it is uncertain about the existance of the information being sought and the ability that the searcher can find it. In the worst case, searcheres are even unsure about their goals. Take the search process of iterative search and exporatory search for instance. In Fig.2(a), as the search target is known, our task is to create the well-formed query that will retrieve it so that during the search process, query is reformulated to make the search result close to the target. In contrast, when we seek some information of our interest, although we have a certain information need, our search is likely to be varied. For example, in Fig.2(b), obviously the first-time search result is not what we need, so query is reformulated and the second-time search is started. At this time, it seems to be relevant to our information need so that the following searches tend to converge. However, after we understand this field of information to some extent, we get to realize that it is not what we are seeking for. So a new search journey is started. That means in exploratory search, with the deepening of understanding, the information need becomes clear and definite. If the whole knowledage environment about a query topic is well organized in advance, it is possible for users to realize their ambiguous information need at the early stage so that the number of searches can be reduced substantially.

"Neighbor information", a similar concept of our "surrounding information", is effectively used in some studies. Castillo et al. incorporated neighbor information to detect



(a) Iterative search process



(b) Exploratory search process

図 2 Examples of search process of iterative search and exploratory search. The red triangle in (a) represents the search target. Both iterative search and exploratory search are within the whole information space. Each result set obtained by a query is denoted as a circle, whose size means the total number of returned documents. The nearby number means the search sequence.

web spam in [9]. They were inspired by the hypothesis that linked hosts tend to belong to the same class: either both are spam or both are non-spam. In their link-based feature analysis, neighbor is referred to as neighboring documents in a hyperlink environment. With the help of predicted labels of neighboring hosts, spam classifier can be retrained to get a better classification result. In [12], Wan and Xiao stated an approach to improve single document keyphrase extraction. They also exploited neighbor information. But in their work, the neighbor documents are obtained by using document similarity search techniques, which is a content-based neighbor information.

Tajima el al. stated a query framework for hypertext data in [4]. It is mainly about using a series of nodes corresponding to one topic as the data units in queries, rather than just an individual node. This query framework can help us when our aim is to discover unknown data concerned with a topic of our interest since it returns a collection of nodes talking about the same topic. In their paper, a connected subgraph corresponding to one topic is regarded as a cut. They concentrated in partitioning a graph into precise cuts by cutting all edges if there exists a topic change. An important assumption is that similarity between two neihboring nodes must be higher when they are discussing the same topic than when they are discussing different topics. Therefore, similarity between the contents of two nodes is used to detect topic changes. As our work also contains the same topic detection, in this degree, ours is similar to the above-mentioned work by Tajima el al. But their approach can only be applied to tree structures. As a result, if we want to apply this approach to web pages, we have to transform the general graph generated by links between web pages to a tree structure in advance. Besides, their cut in web pages, does not extend over multiple web sites, which means this approach can only discover web pages discussing the same topic within a single web site. While on the other hand, our text landscape discovery has no limitation in graph structure, and extends to web pages or documents discussing similar, or semantic similar topics from multiple web sites.

3. Text Landscape

The terminology “information landscape” has appeared since 1990s. Russell [3] thought it is used to describe a way of presenting different views (personalised or community interest) of information resources to users, based on their interests and needs. An information landscape could be a set of web pages which link to certain resources. It also could be a view constructed dynamically according to user’s profile. In [5], “information landscape” is defined as the user’s personal view of the information universe, which is intended to represent a complete working environment. Information resources are not separated out as standalone items, but integrated into people’s working and learning environments. That means the more important issue of “information landscape” is to dynamically lead users to new things, perhaps their potential interests.

"Text landscape" is subtly different from the above-mentioned "information landscape". It is used to describe the whole knowledage environment about a query topic in a well-organized way. It is supposed to support the ambiguous exploratory search, especially those which has unknown goal before a query is issued, and user's learning process. Documents that share the same, similar, or semantic similar topics with a document, are called "surrounding information" of the certain document. Here the topics contain not only those of a whole document, but also those extracted from a part of a document, such as topic of a paragragh. As a result, documents of a query, together with their "surrounding information", build up the "text landscape" of the afore-mentioned query. Since the whole knowledge environment about a query topic is well organized, it is no longer a trial-and-error journey through the information space as the destination. Users can figure out their subconscious goal with the help of "text landscape".

For example, a problem we often encounter. Consider an academic paper discussing ranking algorithm in information retrieval architecture, such as [11]. When we read a paper, of course not all methods or terminologies can we understand. At this time, one choice is to read some references of this paper. Suppose we have little knowledge about PageRank algorithm mentioned in [11]. Then we may check one of [11]'s references "The Anatomy of a Large-Scale Hypertextual Web Search Engine" written by S. Brin and L. Page. Still, if we encounter other unknown things, we will read references of S. Brin and L. Page's above-mentioned paper. In a simple case, just treat references as surrounding information, then the relations among all related papers can show us a brief introduction of a certain research field. For another example, consider an event stated in a news web page. A simple case is that the causation and the result of the event will be also presented to users as well. The "text landscape" of the event is, but not limited to the detailed develpment, the causation and the result of the event.

4. Basic Idea

The problem described in this paper is as follows:

- **Input**: A web page of search result of a query, indicated by user
- **Output**: The whole "text landscape" of the query topic

As our notion is to present user "surrounding information" of an input query to completely show the knowledge environment, the following two relations between documents, text similarity and text adjacency, are considered as key factors to detect "surrounding information" of a specific query.

4.1 Text Similarity

This problem is as follows, given a web page as our input, other web pages that are similar to the input are expected to be flagged based on their similarity to the input one. Suppose there is a web page dataset P_n as follows, P_n=$\{p_1, p_2, ..., p_i, ..., p_n\}$, where n is the total number of pages in this dataset. When any one page p_i in the dataset is given as an input, the expected output is similarities between other pages in the dataset and the input page. The functional form representing this problem is as follows,

$$Sim(p_i, p_j)$$

where p_j is any other page in the dataset, $1 \leqq i \leqq n$, $1 \leqq j \leqq n$. It is desired that this function returns a real number, which has a high value when the content of p_j is highly similar to that of p_i, and vice versa.

4.2 Text Adjacency

As we discussed in Section 3., "surrounding information" of a document is those documents that share the same, similar, or semantic similar topics with the certain document. In this paper, the degree that two documents or web pages share the same, similar, or semantic similar topics is referred to as "text adjacency". In other words, "text adjacency" is a kind of partial similarity between two documents or web pages.

The problem is, then similar to that of text similarity. Given a web page as our input, it is expected that other pages adjacent to this page will be highlighted in terms of adjacency with the input one. When any one page p_i in the dataset is given as an input, the expected output is adjacencies between other pages in the dataset and the input page. The functional form representing this problem is as follows,

$$Adj(p_i, p_j)$$

$1 \leqq i \leqq n$, $1 \leqq j \leqq n$. A high value will be returned when p_j is highly adjacent to p_i, and vice versa.

5. Our approach

Here we explain our graph-based ranking method PanoramaRank for "text landscape" in detail. It is a revision of our previous work [14]. Roughly speaking, we employ PanoramaRank to discover a "text landscape" of a user-indicated web page. Text similarity and text adjacency is combined in our method to find "surrounding information" of a certain web page.

5.1 Calculating Text Similarity

Text similarity is obtained by calculating similarity between two web pages or documents. Here we employ the widely-used vector space model [1]. In this model, given a set of n documents, each document d_i is represented by a

t-dimensional vector,

$$V(d_i) = (w_{i1}, w_{i2}, ..., w_{it}),$$

w_{ij} representing the weight of the jth term, t representing the number of unique terms that occurs in any of $d_1, d_2, ..., d_n$. In the equation above,

$$w_{ij} = tf_{i,j} \cdot log\frac{W}{w},$$

where $tf_{i,j}$ is the term frequency of term j in document d_i, $log\frac{W}{w}$ is the inverse document frequency. Here W is the total number of documents in the document set, w is the number of documents containing the term j.

Thus, $Sim(p_i, p_j)$ which denotes the content similarity between web page p_i and p_j, is computed by cosine similarity of their feature vectors, defined as below,

$$Sim(p_i, p_j) = \frac{V(p_i) \cdot V(p_j)}{|V(p_i)||V(p_j)|}$$

$$= \frac{\sum_{m=0}^{t} w_{m,i} \cdot w_{m,j}}{\sqrt{\sum_{m=0}^{t} w_{m,i}^2} \cdot \sqrt{\sum_{m=0}^{t} w_{m,j}^2}}.$$

5.2 Calculating Text Adjacency

In our previous work [14], adjacency between two images is defined as the overlap degree between them. It can be also considered as an approach to find similarity of fragments of the two images. Similarly, in text architecture, text adjacency is partial similarity between two web pages. Before "text adjacency" is computed, each web page must be divided into several parts.

One way to get "text adjacency" between web page p_i and p_j is to take the maximum similarity among those of all part-pairs, shown as follows:

$$Adj(p_i, p_j) = \max_{x \in Part(p_i), y \in Part(p_j)} Sim(x, y)$$

where $Part(p_i)$ and $Part(p_j)$ are referred to as a part of web page p_i and p_j, respectively.

In order to avoid web pages with high text similairities gathering together, we also consider a way to give a penalty in terms of "text similarity" between two web pages, which is defined as follows:

$$Adj(p_i, p_j) = \frac{\max_{x \in Part(p_i), y \in Part(p_j)} Sim(x, y)}{Sim(p_i, p_j)}$$

where $Part(p_i)$ and $Part(p_j)$ are referred to as a part of web page p_i and p_j, respectively.

There are many algorithms with which a full-length document can be divided into several parts in terms of their different subtopics discussing. In [2], using orthographically marked segments supplied by the author to determine topic boundaries is confirmed consistent with human judgment. So in this paper, we just break the document into paragraphs and regard similarity of paragraphs between two documents as "text adjacency" between these two documents.

5.3 PanoramaRank for "Text Landscape"

In [11], VisualRank, which is an inferred visual similarity graph-based ranking model for image-ranking problems, was introduced. In this model, edge weights have also been considered when estimating the score associated with a vertex in the graph. The random walk algorithm is employed to rank images based on the visual hyperlinks among the images. It is assumed that if a user is viewing an image, other related(similar) images may also arouse the user's interest. Similar to PageRank algorithm, if image u is visually hyperlinked to image v, such hyperlink is treated as a vote of confidence, which means it is possible that the user will go from viewing u to viewing v. As a result, images related (similar) to the query image will have many other images pointing to them and will therefore be viewed often. In [11], given n images, VisualRank (VR) is iteratively defined as

$$VR = dS^* \times VR + (1 - d)p$$

where $p = [\frac{1}{n}]_{n \times 1}$. S^* is the column normalized adjacency matrix S, where $S_{u,v}$ denotes the visual similarity between image u and v. d is a damping factor between 0 and 1, which is usually set to be greater than 0.8.

Here, let $G_S = (V, E_S)$ be an undirected graph with a set of vertices V and a set of edges E_S, where E_S is a subset of $V \times V$, $E_S = \{e = (u, v) | \text{u is similar to v}\}$. Likewise, let $G_A = (V, E_A)$ be an undirected graph with a set of vertices V and a set of edges E_A, where E_A is a subset of $V \times V$, $E_A = \{e = (u, v) | \text{u is adjacent to v}\}$. Then the final similarity/adjacency graph(SA graph for short below) for ranking is defined as $G = G_S \cup G_A$, where $E_S \cap E_A \neq \phi$.

In our case, we apply the above-mentioned random walk algorithm to the defined SA graph G. Thus, after iterative calculation, web pages not only content similar but also discussing the same, or similar, or semantic similar topics, with specified page(s), will come to the top. For instance, if a web page about an event is assigned as the starting point, a simple case is that the causation and the result of this event will be also returned as well. The surrounding information of the assigned page is, but not limited to the causation and the result of this event.

Given n web pages, our proposed PanoramaRank (PR) is defined as follows:

$$PR = dS^* \times PR + (1 - d)p$$

where p_i is the initial value of V_i, and we refer to vertex V_i of a page p_i. d is a damping factor, and we set it to 0.85 in our evaluation experiments empirically. S^* is the column normalized adjacency matrix S, but here S_{p_i,p_j} denotes the combination of text similarity and text adjacency between web page p_i and p_j.

A simple way to set initial value of a page is whether this page is indicated by user as an input or not, defined as follows:

$$p_i = \begin{cases} 1, \text{web page } p_i \text{ is specified as an input} \\ 0, \text{otherwise} \end{cases}$$

We also consider another way to set initial value of a page based on semantic similarity among keywords of the page and those of input page. For computing sementic similarity between two terms, we employ normalized Google distance introduced by Cilibrasi and Vitanyi in [10]. It is derived from the number of hits returned by the Google search engine for words and phrases from the world wide web. Words or phrases with the same or similar meanings tend to be "close" when evaluating their semantic similarity by Google distance, while words or phrases with dissimilar meanings tend to be farther apart. Especially, the normalized Google distance (NCG) between two words x and y is defined as below:

$$NGD(x,y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - \min\{\log f(x), \log f(y)\}}$$

where $f(x)$ denotes the number of pages containing word x, and $f(x,y)$ denotes the number of pages containing both word x and word y, as searched by Google search engine. By increasing N, the NGD is decreased, and everything tends to get closer together, while by decreasing N, the NGD is increased, and everything tends to get further apart. In their experiments, the number of web pages indexed by Google at the time of their writing is used as normalizing factor N.

Thus, initial value of web page p_i in terms of semantic similarity among keywords of the page and those of input page p_Q is defined as:

$$p_i = \Theta_{x \in Keyword(p_i), y \in Keyword(p_Q)} NGD(x,y)$$

where Θ is considered as:

$$\Theta = \max, \text{or avg}, \text{or } \frac{\max - \min}{\max}.$$

$Keyword(p_i)$ and $Keyword(p_Q)$ are referred to as keyword set of web page p_i and p_Q, respectively. In the experiments we plan to do in the near future, just take words, or phrases in each news title as keywords of that news web page, since for newspaper article, news title is typically a good summary of its content. Of course, other keyword extraction algorithms such as TextRank [8], can be applied here to extract keywords of a web page, or document.

The weight between web page p_i and p_j in SA graph G is defined as the arithmetic mean of their text similarity $Sim(p_i, p_j)$ and their text adjacency $Adj(p_i, p_j)$.

$$S_{p_i,p_j} = \alpha \cdot Sim(p_i,p_j) + (1-\alpha) \cdot Adj(p_i,p_j)$$

where α is the weight factor, whose default setting is 0.5.

6. Conclusions

In this paper, we proposed a method that can be used to find the whole "text landscape" of the query topic according to a user-indicated document, which helps searcheres easily find or clear their target, or even awake to their ambiguous and unknown goal. Unlike the state-of-the-art search technolodies only focus on similarity of documents' contents to get extremely similar documents, we aimed to extend search result to "surrounding information" of specified document(s). For the sake of seeking "surrounding information", the concept of "text adjacency" was introduced into search framework. With the help of our proposed method, PanoramaRank for text landscape, documents not only content similar but also discussing the same, or similar, or semantic similar topics, with specified document(s), will come to the top. It means finally we can retrieve more "related" information, not only "relevant" ones.

Because of the lack of time, we did not do evaluations of our method. So in the near future, the first thing we need to do is to evaluate PanoramaRank for text landscape. We plan to do our evaluations based on a news web page dataset. Several text landscape about different query topics are prepared in advance, while web pages in each text landscape are viewed and judged by humans to make sure they are discussing the same, or similar, or semantic similar topics with a certain topic. Take last-minute retirement of civil servants for instance. Not only web pages about the current situation of the last-minute retirement, but also pages of the causation led to this incident, the result caused by this incident, and attitudes of multitude as well. Given any page from the dataset, such as a page about the last-minute retirement of civil servants, each returned page will be judged in terms of whether it is a page belonging to the text landscape of the last-minute retirement of civil servants indeed. In other words, we endeavor to evaluate the precision of discovering a text landscape by our proposed method.

Also, the time factor is an important factor for "text landscape" discovery, especially for newspaper articles. With the lapse of time, the progress of an event is changing beyond all doubt, sometimes even to the completely opposite direction. Therefore, our second step is to bring the time factor into "text landscape" discovery.

Acknowledgements

This work was supported in part by KAKENHI (Nos. 24240013, 24680008).

文　　献

[1] G. Salton, A. Wong, C. S. Yang, “A vector space model for automatic indexing”, Communications of the ACM, v.18, n.11, pp. 613–620, Nov. 1975

[2] Marti A. Hearst, Christian Plaunt, “Subtopic structuring for full-length document access”, Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 59–68, June 27-July 01, 1993, Pittsburgh, Pennsylvania, United States

[3] R. Russell, “Mining the information landscapes: the Agora Project”, The New Review of Information and Library Research. 4, pp 121-127, 1998

[4] Keishi Tajima , Yoshiaki Mizuuchi , Masatsugu Kitagawa , Katsumi Tanaka, “Cut as a querying unit for WWW, Netnews, e-mail”, Proceedings of the 9th ACM conference on Hypertext and hypermedia : links, objects, time and space —structure in hypermedia systems: links, objects, time and space—structure in hypermedia systems, p.235-244, June 20-24, 1998, Pittsburgh, Pennsylvania, United States

[5] R. Heseltine, “Alice through the looking glass: information spaces for a new learning generation”, Information Landscapes for a Learning Society: Networking and the future of libraries 3 ed. S. Criddle, L. Dempsey and R. Heseltine. London: Library Association publishing (in association with the UK Office for Library and Information Networking), pp. xv-xxiii, 1999

[6] Andrei Broder, “A taxonomy of web search”, ACM SIGIR Forum, v. 36, n. 2, pp. 3–10, Fall 2002

[7] Daniel E. Rose, Danny Levinson, “Understanding user goals in web search”, Proceedings of the 13th international conference on World Wide Web, pp. 13–19, 2004

[8] R. Mihalcea, P. Tarau, “Textrank: Bringing order into text”, Proceedings of the conference on Empirical Methods in Natural Language Processing, pp. 404–411, 2004

[9] Carlos Castillo, Debora Donato, Aristides Gionis, Vanessa Murdock, “Know your neighbors: web spam detection using the web topology”, Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 423–430, 2007

[10] Rudi L. Cilibrasi, Paul M. B. Vitanyi, “The Google Similarity Distance”, IEEE Transactions on Knowledge and Data Engineering, v.19 n.3, pp. 370–383, March 2007

[11] Yushi Jing, Shumeet Baluja, “VisualRank: Applying PageRank to Large-Scale Image Search”, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.30 n.11, pp. 1877–1890, November 2008

[12] Xiaojun Wan, Jianguo Xiao, “Single document keyphrase extraction using neighborhood knowledge”, Proceedings of the 23rd national conference on Artificial intelligence, v. 2, pp. 855–860, 2008

[13] Ryen W. White, Resa A. Roth, “Exploratory search : beyond the query-response paradigm”, 2009

[14] Meng Zhao, Hiroaki Ohshima, Katsumi Tanaka, “Panoramic image search by similarity and adjacency for similar landscape discovery”, Proceedings of the 13th International Conference on Web Information Systems Engineering, pp. 284–297, 2012