A Visual Citation Search Engine

Tetsuya Nakatoh[1](✉), Hayato Nakanishi[2], Toshiro Minami[3,4], Kensuke Baba[3], and Sachio Hirokawa[1]

[1] Research Institute for Information Technology, Kyushu University,
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan
nakatoh@kyudai.jp

[2] Graduate School of Integrated Frontier Sciences, Kyushu University,
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan

[3] Kyushu University Library,
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan

[4] Kyushu Institute of Information Sciences, 6-3-1 Saifu,
Dazaifu, Fukuoka 818-0117, Japan

Abstract. Carrying out the survey of the related researches is an essential part in research activities, the aim of which is to have an overall view of the target field. Generally, we take two approaches toward this aim. One approach is paying attention to selected articles and deeply investigate them. The selection is performed according to some indicators for measuring importance. The other approach is considering the citation relations between articles. One problem is that these approaches cannot be combined straightforwardly. Another problem in carrying out the survey is that there are a huge amount of articles exist already. The aim of this paper is to propose a framework of a visualization system that assists us in surveying related researches. The system displays the important articles together with their key citation relations by displaying not only direct citations between important articles but also the indirect, or weak-tie, citation relations that connect them.

Keywords: Bibliometrics · Research investigation · Citation count · Visualization · Thread-Tree · Weak-tie

1 Introduction

Surveying the related researches is one of the indispensable activities in carrying out academic researches. For new research areas which have no standard textbooks nor survey articles, we have to search a massive amount of scholarly information for related researches. The simplest way to find related researches is searching research articles by some keywords. However, the result of a search often contains too many articles for humans to deal with due to the existence of a large number of articles have been published already, and the number is increasing very rapidly. Thus, novel technologies are needed in order to survey research articles from among the vast amount of scholarly information.

S. Yamamoto (Ed.): HIMI 2016, Part I, LNCS 9734, pp. 168–178, 2016.
DOI: 10.1007/978-3-319-40349-6_17

We generally take two approaches as we survey a research field and grasp their overall view. One of them is selecting articles by considering their *importance* and investigate deeply how they deal with the research topics. A specific feature of research articles is that they are related each other by citations (or references). The number of the citations from other articles, i.e., the number of the articles that cite the target article, is a simple but useful measure for evaluating the importance of the target article; which is called its *citation count*. Actually, the conventional databases, such as Scopus[1], use the citation count of articles in the function of sorting the result of a search. Impact factor [1,3], a popular measure for evaluating academic journals, is calculated using the citation count of the articles published in the target journal. The h-index [4] measure for evaluating researchers is calculated on the basis of the number of the articles written by the target researcher and the citation counts to the articles.

The other approach to get useful information for surveying a research field is investigating the structural features of citations between research articles. In this paper, we call the graph that represents the citation relations between the articles the *citation graph*.

We take the research of Garfield et al. [2] as an example for structural analysis of citation graph. They conducted an analysis on the history of a research area by using a figure of the citation relations between the concerned articles.

Structural analysis can be used as a complement to the approach using the citation count. Suppose there are two articles having the same citation count. It may happen that their citation graphs are contrastingly different; e.g., one may have a long chain of citations in a specific research topic, whereas the other may be cited by the papers relating to a wide variety of research topics. Thus, it is preferable to evaluate the papers on the basis of these two approaches as we perform a survey of a research field.

As we have pointed out, it is an important task to find out important articles and to capture the relations between them for surveying a research field. As a part of "research automation," a search engine system will play a very important role in pursuing our research activities more effectively and more efficiently. Also, since such activities are basically carried out in trial-and-error, such system should be designed as an interactive system, and thus HCI is a crucial part in designing such a system.

In this paper, we propose a system which visualizes the citation relations of important articles in terms of citation count and how they are connected, so that the user can interactively specify which graph to display. In this visualization, the user of the system chooses the nodes and edges of the displayed citation graph by considering the paths from particular nodes. We can interactively choose an article by clicking the node to see detailed information of the article. The graph can be changed by restricting the length of the path or the number of the edges of each node.

The ultimate goal of our study is to develop a search engine with a good HCI interface. As a very beginning toward this goal, we start with proposing a basic

[1] http://www.scopus.com/.

mechanism for visualization of articles in this paper. The main contribution of this paper is the proposal of the visualization method which shows the important articles together with the key relations between them. We call this visualization by “Thread-Tree.”

The rest of this paper is organized as follows. In Sect. 2, we describe the construction method of Thread-Tree of the citation. In Sect. 3, we show how a search engine can be designed by using the algorithm described in Sect. 2. We also show an example of application to real data. In Sect. 4, we discuss the differences of the features of our system from other related researches. In Sect. 5, we summarize the study in this paper and show some of the research topics for the future.

2 Construction of Thread-Tree

The system proposed in this paper starts with receiving an article as input and displays its Thread-Tree as output. The root of the Thread-Tree represents the article given as input and the other nodes represent the selected articles among the articles citing the root article either directly or indirectly by using the concept of weak-tie.

The whole process of the system consists of the following three small steps:

(1) Construct Thread-Tree from citation graph
(2) Filter important articles
(3) Recover Weak-Tie nodes and edges

In Step 1, the system starts with collecting the articles that cite the article which is given as input. Then the system constructs the citation graph of the articles. The “Thread-Tree” having the given article as the root is generated from the citation graph. In this process, the most important citation link is chosen for each article except the root, and other links are eliminated. Here, the importance is measured by the number of the citing articles, or the citation count, of the article under consideration. In this way, the tree structure (Thread-Tree) is constructed.

In Step 2, the system eliminates the nodes (or articles) which are judged as unimportant on the basis of the citation count; i.e., the citation count is less than the given threshold value. The edges relating to the eliminated nodes are also eliminated. After this process, the obtained Thread-Tree may not be connected even if the original citation graph is connected.

In Step 3, the system recovers some of the articles and their links that were once eliminated in Step 2. The recovered nodes are chosen if they are eliminated in Step 2 as unimportant but still they appear on the paths that connect some of the important nodes and the root node. The edges of them are also recovered accordingly.

2.1 Step 1: Citation Graph to Thread-Tree

Let $\mathcal{A}$ be the set of articles and let $\mathcal{C}$ be the citation relation between the articles in $\mathcal{A}$; i.e., $\mathcal{C} \subseteq \mathcal{A} \times \mathcal{A}$. We will use the notation $b \leftarrow a$ in two senses; as an alternative representation of the binary relation $(b, a) \in \mathcal{C}$ between b and a, and as the element (b, a) of $\mathcal{C}$. The citation graph G is the graph structure consisting of $\mathcal{A}$ and $\mathcal{C}$; i.e., $G = (\mathcal{A}, \mathcal{C})$. We denote $\mathcal{N}(G)$ for $\mathcal{A}$ and $\mathcal{E}(G)$ for $\mathcal{C}$. We assume there are no loops in the citation graph G, because, basically, an article a cites b only after b is published and thus the publication time of a becomes later than that of b. Further, self citation, $a \leftarrow a$, will not happen as well.

We denote the transitive closure of $\leftarrow$ by $\Leftarrow$; i.e., $b \Leftarrow a$ iff there exists a sequence of nodes $a_0, a_1, \ldots, a_{n+1}$ for some $n \geq 0$, and $a_0 = b, a_{n+1} = a$ and $a_i \leftarrow a_{i+1}$ for $i = 0, 1, \ldots, n$. In other words $b \Leftarrow a$ means that the article b is either directly or indirectly cited by the article a. The citation count of an article $a \in \mathcal{A}$ is defined as the number of the articles b that cite a; i.e., $cc(a) = \#\{b \in \mathcal{A} \mid a \leftarrow b\}$, where $\#S$ indicates the number of the elements of the set S.

We define the Thread Graph $TG(a)$ of the specified article a ($\in \mathcal{A}$) as a subgraph of G as follows:

$\mathcal{N}(TG(a)) = \{b \in \mathcal{A} \mid a \Leftarrow b \text{ or } b = a\}$; i.e., the nodes of $TG(a)$ consist of a and the node of $\mathcal{A}$ which directly or indirectly cites a.

$\mathcal{E}(TG(a)) = \{b \leftarrow c \in \mathcal{C} \mid b, c \in \mathcal{N}(TG(a))\}$ ($\subseteq \mathcal{C}$); i.e., the edges which connect the two nodes of $TG(a)$.

The Thread-Tree $TT(a)$ of a is defined as a subgraph of $TG(a)$, which has a as its root node and the edges are chosen so that each node except a has only one edge to another node. We define $TT(a)$ as follows:

$\mathcal{N}(TT(a)) = \mathcal{N}(TG(a))$; i.e., the nodes are the same as of $TG(a)$.

For each node b ($\neq a$) in $TT(a)$, most important node $\iota(b)$ is chosen so that it satisfies the following conditions: $\iota(b) \leftarrow b$ and $cc(\iota(b)) \geq cc(c)$ for any c such that $c \leftarrow b$. As we take the citation count as the measure for importance of an article, $\iota(b)$ is the, or one of the, most important article(s) among the articles cited by b.

The edges of $TT(a)$ is defined formally as: $\mathcal{E}(TT(a)) = \{b \leftarrow c \in \mathcal{E}(TG(a)) \mid b = \iota(c)\}$.

2.2 Step 2: Elimination of Non-important Articles

As we have already mentioned, we use the citation count as the measure for importance of articles. Thus we set up a threshold value m for judging an article if it is important or not. Now we define the subgraph of $TT(a)$ in which only the important nodes (articles) of $TT(a)$ are left and other nodes together with their edges are eliminated. The resulting graph structure is not necessarily a tree anymore in general, so we call it a "graph."

Let m (≥ 0) be a number. We define the Thread Graph $TG_m(a)$ of the article a with the threshold value m as follows:

$\mathcal{N}(TG_m(a)) = \{b \in \mathcal{N}(TT(a)) \mid cc(b) \geq m \text{ or } b = a\}$; i.e., the nodes consist of the article a and the important articles of $\mathcal{N}(TT(a))$. Where "important article" means its citation count is greater or equals to the threshold value m.

$\mathcal{E}(TG_m(a)) = \{b \leftarrow c \in \mathcal{E}(TT(a)) \mid b, c \in \mathcal{N}(TG_m(a))\}$. The edges of $\mathcal{E}(TG_m(a))$ consist of those in $\mathcal{E}(TT(a))$ so far as their both end nodes are important (i.e., belong to $\mathcal{N}(TG_m(a))$). From the definition, $TG_0(a) = TT(a)$.

Figure 1 shows an example of the Thread-Tree of the article by Newman [10] with the threshold value 10. Unfortunately, the graph is very complicated and it is difficult for us to understand and extract some kind of valuable findings from this graph.

Thus we need to find more sophisticated criteria for restricting the number of nodes so that we are able to have an image for surveying the research field. So, not only the Citation Count, but also a lot of other evaluation criteria have been investigated, so far; e.g., Focused Citation Count, Accumulated Citation Count, Journal Impact Factor, Focused Journal Impact Score, h-index, etc. The combined use of them and evaluation using the length of thread might be useful as well. In this paper, we use Citation Count only for measuring importance and the left possibilities are remained as one of our future work.

2.3 Step 3: Thread-Tree by Recovering the Weak-Ties

As the result of Step 2, the resulting graph $TG_m(a)$ becomes a subgraph of the Thread-Tree $TT(a)$ of a, thus it is a collection of subtrees of $TT(a)$ in general. A problem here is that even though there exist the connections, or indirect citation links between the existing articles, they may not be displayed in the Thread Graph $TG_m(a)$. In order to solve this problem, we rescue the once-eliminated nodes and edges to $TG_m(a)$ so that the resulting graph becomes a subtree of $TT(a)$ and only the important nodes as well as the necessary nodes together with their connecting edges appear in the graph.

The final Thread-Tree $TT_m(a)$ of the article a with the threshold value m is defined as follows:

$\mathcal{N}(TT_m(a)) = \{b \in \mathcal{N}(TT(a)) \mid$ there exists a sequence $a_0, a_1, \ldots, a_{n+1}$ for some $n \geq 0$ such that $a_0 = a, a_{n+1} \in \mathcal{N}(TG_m(a)), a_i = b$ for some $i = 0, 1, \ldots, n+1$, and $a_i \leftarrow a_{i+1} \in \mathcal{E}(TT(a))$ for all $i = 0, 1, \ldots, n\} \cup \mathcal{N}(TG_m(a))$; i.e., the nodes which are either contained in the Thread Graph $TG_m(a)$ or those that appear in the paths from a node in $TT(a)$ on the way to the root node a.

$\mathcal{E}(TT_m(a)) = \{b \leftarrow c \in \mathcal{E}(TT(a)) \mid b, c \in \mathcal{N}(TT_m(a))\}$; i.e., the paths that appear in the Thread-Tree $TT(a)$ and need to be used in the paths to connect an important node to the root a.

We call an edge (link) as a Weak-Tie if it appears in $TT_m(a)$ and does not appear in $TG_m(a)$. As has been pointed out of the importance of Weak-Ties between human beings in social relations, our concept of Weak-Ties for the articles are also important in capturing the overall citation relations among important articles.

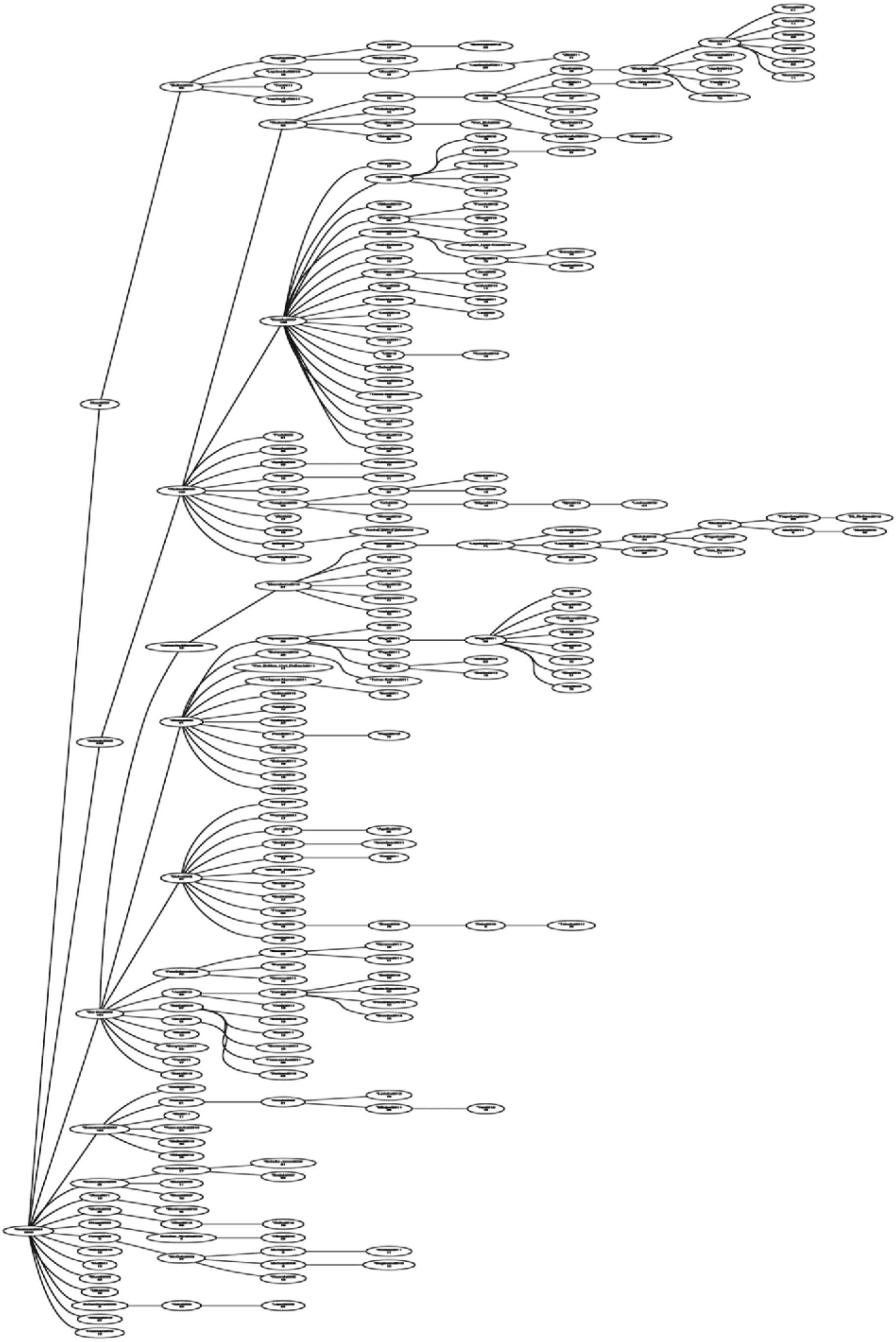


Fig. 1. Thread-Tree of Newman 2001

3 Experiment with Real Data

3.1 Gathering Article Data and Basic Analysis

In this section, we explain the collection method of articles for analysis, and conduct a basic analysis of the data.

The article data were gathered from Scopus. In this experiment, we chose the term "bibliometrics" as the query keyword to specify the research field. We got 10,186 articles with the publishing years range from 1976 to 2015 through the Scopus's search API.

These data are written in JSON format, where the items consist of: "Content Type," "Search identifier," "Complete author list," "Resource identifiers," "Abstract Text," "First author," "Page range," "SCOPUS Cited-by URI," "Result URL," "Document identifier," "Publication date," "Source title," "Article title," "Cited by count," "ISSN," "Issue number," and "Volume."

While some articles have no citation at all, an article has 2,977 citations. The total number of citations becomes 116,743.

Then we gathered the articles which cite other articles. The citation information of an article appears at the "link" item of JSON format in the form of URL. Since the Scopus API does not provide with the sufficient information of the articles at the URLs, we obtained the HTML files using the "wget" command. The number of citation articles obtained by the wget command is limited to 20 at maximum in one HTML file, we called the wget command repeatedly until we got all articles we needed.

Since 3,024 articles have no citations from other articles, we obtained the remaining 7,162 articles as those which have at least one citation counts. The wget command was called 10,719 times for these 7,162 articles and 116,743 citation data were obtained as the result. After eliminating the duplications, 62,265 articles were left.

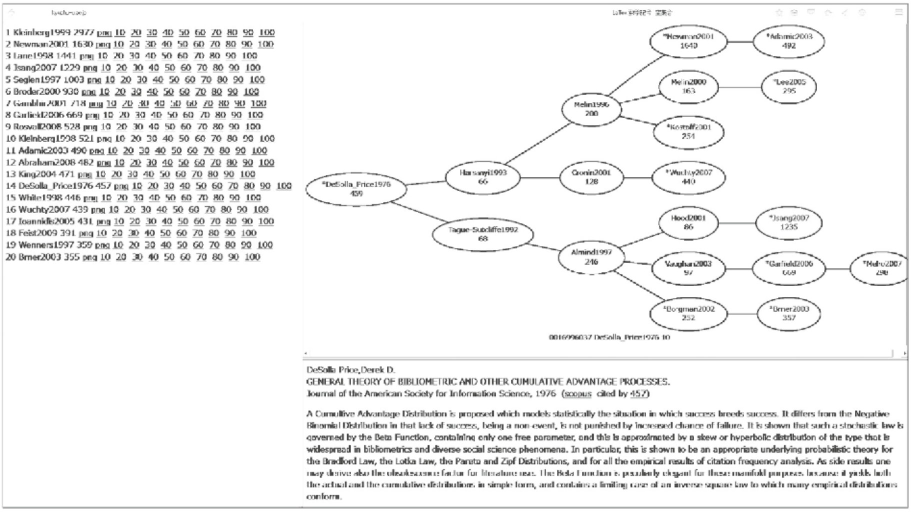
3.2 Visualized Citation Search Engine

Figure 2 shows a screenshot of the visual citation search engine we are developing. The system displays a citation graph which is explained in Sect. 2. It is built by using the data explained in Sect. 3.1.

The top window of the search system consists of three panes. Articles are obtained according to the given search keyword. Top 20 lists of articles with citation information are displayed on the left pane. Each line looks like as follows:

```
2 Newman2001 1630 png 10 20 30 40 50 60 70 80 90 100
```

The first numerical value shows the ranking in terms of the citation counts. The 2nd item shows the article name in an abbreviated format, which consists of the name of the first author and the publication year of the article. In this case, "`Newman2001`" represents the article [10] "The structure of scientific collaboration networks" written by Newman and published in 2001. The 3rd item `1630` shows the citation count of the article.



1 Kleinberg1999 2977 png 10 20 30 40 50 60 70 80 90 100
2 Newman2001 1630 png 10 20 30 40 50 60 70 80 90 100
3 Lane1998 1441 png 10 20 30 40 50 60 70 80 90 100
4 Jsang2007 1229 png 10 20 30 40 50 60 70 80 90 100
5 Seglen1997 1003 png 10 20 30 40 50 60 70 80 90 100
6 Broder2000 930 png 10 20 30 40 50 60 70 80 90 100
7 Gamblin2001 718 png 10 20 30 40 50 60 70 80 90 100
8 Garfield2006 669 png 10 20 30 40 50 60 70 80 90 100
9 Rosvall2008 528 png 10 20 30 40 50 60 70 80 90 100
10 Kleinberg1998 521 png 10 20 30 40 50 60 70 80 90 100
11 Adamic2003 490 png 10 20 30 40 50 60 70 80 90 100
12 Abraham2008 482 png 10 20 30 40 50 60 70 80 90 100
13 King2004 471 png 10 20 30 40 50 60 70 80 90 100
14 DeSolla_Price1976 457 png 10 20 30 40 50 60 70 80 90 100
15 White1998 446 png 10 20 30 40 50 60 70 80 90 100
16 Wuchty2007 439 png 10 20 30 40 50 60 70 80 90 100
17 Ioannidis2005 431 png 10 20 30 40 50 60 70 80 90 100
18 Feist2009 391 png 10 20 30 40 50 60 70 80 90 100
19 Wenners1997 359 png 10 20 30 40 50 60 70 80 90 100
20 Brner2003 355 png 10 20 30 40 50 60 70 80 90 100

DeSolla Price,Derek D.
GENERAL THEORY OF BIBLIOMETRIC AND OTHER CUMULATIVE ADVANTAGE PROCESSES.
Journal of the American Society for Information Science, 1976 (scopus cited by 457)

A Cumulative Advantage Distribution is proposed which models statistically the situation in which success breeds success. It differs from the Negative Binomial Distribution in that lack of success, being a non-event, is not punished by increased chance of failure. It is shown that such a stochastic law is governed by the Beta Function, containing only one free parameter, and this is approximated by a skew or hyperbolic distribution of the type that is widespread in bibliometrics and diverse social science phenomena. In particular, this is shown to be an appropriate underlying probabilistic theory for the Bradford Law, the Lotka Law, the Pareto and Zipf Distributions, and for all the empirical results of citation frequency analysis. As side results one may derive also the obsolescence factor for literature use. The Beta Function is peculiarly elegant for these manifold purposes because it yields both the actual and the cumulative distributions in simple form, and contains a limiting case of an inverse square law to which many empirical distributions conform.

Fig. 2. A screenshot of the visualized citation search engine

The 4th and the remaining items are clickable. The corresponding graph is displayed on the right pane as the user clicks one of them. The 4th item "`png`" is for displaying the Thread-Tree of the designated article. By default, top 50 articles in terms of the citation count will be displayed. In this case, the whole Thread-Tree is displayed using the whole area of the right-hand pane. It is useful for having the overall view of Thread-Tree.

The 5th and the following items are for specifying the number of articles which appear in the Thread-Tree. As the user clicks the numerical value N, the system will displays the Thread-Tree in the upper part of the right pane, which shows top N articles in terms of the citation count. The displayed image is generated as a Scalable Vector Graphics (svg). At the same time, the bibliographic information of the article, including the author, the title, the journal name, the publication year, the abstract, etc., will be displayed in the lower part of the right pane.

It becomes possible to understand both the global image of research, and the contents of each article because this system visualizes the transition of the articles that cites the specified article displayed as the root.

3.3 Drawing Thread-Tree

In this section, we introduce an example of search by the system, and the effect of Weak-Ties.

Figure 3 is the Thread Graph of the article "The increasing dominance of teams in production of knowledge" written by Wuchty et al. [11] in 2007 (`Wuchty2007` in the graph). The top 10 important articles according to the citation count measure appear in the graph. Although this graph is a subgraph of the Thread-Tree of the article and thus all the articles appearing in the graph cite `Wuchty2007` either directly or indirectly, we cannot see such relations in Fig. 3.

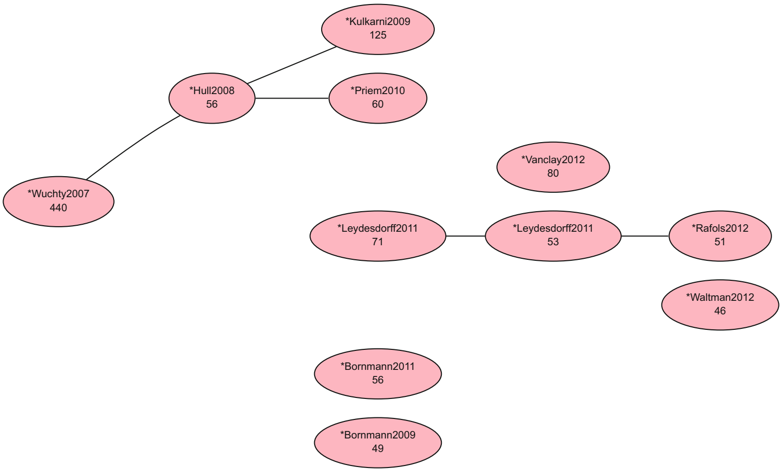


Fig. 3. Thread graph of Wuchty2007 without Weak-Tie

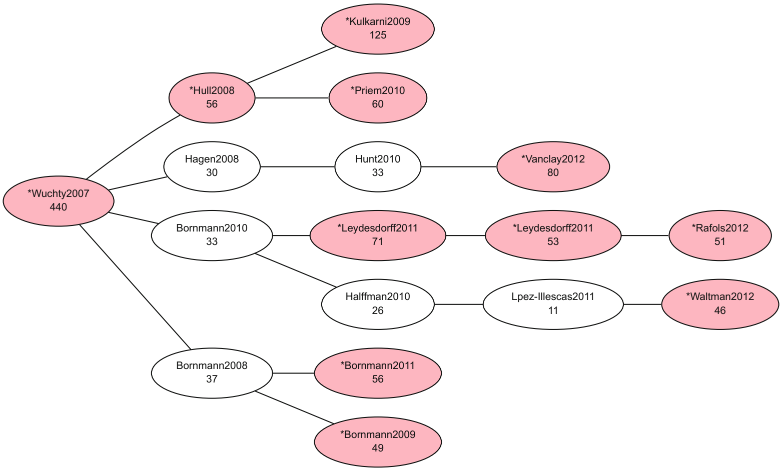


Fig. 4. Thread-Tree of Wuchty2007 with Weak-Tie

Figure 4 shows the graph (tree) obtained by adding the Weak-Tie connections to the graph shown in Fig. 3. See Sect. 2.3 for the definition of Weak-Tie. In this example, 6 articles and 11 citations hidden in the Thread Graph appear and it becomes much easier to capture how the important articles' mutual relations. As we compare these two figures, we easily understand the usefulness of the concept of Weak-Ties.

4 Related Work

There are many work aiming at research investigation. The citation count is useful evaluation for scientific research. Martin [7] has reported that the citation count gained many supports as criteria. Kostoff [5] showed that the citation count as a measure of evaluation has some problems.

It is more appropriate to find out the related articles if we restrict the articles in a specific research area. Nakatoh et al. [9] proposed FCC (Focused Citation Count) which restricts the research area of the cited articles by keywords, and showed that more appropriate articles could be extracted.

Even with such examples of good use of the citation count as a measure for the article's importance, it is not almighty. For example, it is not appropriate to use as a measure to evaluate a new article. Therefore it is common to use the following attribute instead; the evaluation of the scientific journal in which the journal was published or the researcher who wrote the article.

Journal Impact Factor [1,3,6] is one of the most popular evaluation measures of scientific journals. Thomson Reuters updates and provides the score of journals in Journal Citation Reports every year. Hirsch [4] defined the h-index of a researcher as the largest number h such that the researcher wrote h articles and each of the articles is cited from h articles or more. Scopus provides the h-index score of researchers.

For detection of appropriate journals, there are also researches that conduct an analysis focused on a research area. Nakatoh et al. [8] proposed the method of selecting appropriate journals by using the citation which focus on the specific field as evaluation of a journal.

It is one of the further work to combine these evaluation measures with this research.

Garfield et al. [2] visualized the transition of research. They built the network diagram of the articles of DNA research area with the citation index, and showed an analysis. Such research is called "Domain Visualization," "Domain analysis," etc. From the viewpoint of visualization, these researches seem to be close to our research. However, the purpose of these researches is the analysis of research itself, and is not related with the analytic methods. We show the general visualization method for research analysis.

5 Conclusion

It is essential for researchers to survey the related researches as a part of their research activities. They investigate important, or key, articles which have big influence to other articles, and how these articles are related each other. However, a huge amount of scholarly articles exist nowadays and it is practically impossible to find such important articles all by themselves.

Information visualization is used in the system which effectively relieves such time-consuming activity. In this paper, we proposed a new algorithm for displaying important articles and their relations based on the citation information between articles. An advantage of our method is that it solves a dilemma, which is as follows: If you display all the articles and their citation relations as a graph, it becomes too complicated and thus it is impossible to see where the important articles locate and how they are related. On the other hand, if you display important articles and their citation links only, it is hard to recognize how they are related. In our proposed methods, you can get the important articles only and how they are related in terms of their in-direct citation relationships; which we call "Weak-Tie."

We also demonstrated the usefulness of our visualization algorithm by applying it to the articles collected with the term "bibliometrics."

Our future topics include:

- Use and evaluation of other evaluation methods,
- Cooperative use with more complicated search systems,
- Quantitative evaluation and qualitative evaluation of the proposed visualization method,
- Integration to a research assistant system.

References

1. Garfield, E.: Citation indexes for science. Science **122**(3159), 108–111 (1955)
2. Garfield, E., Sher, I.H., Torpie, R.J.: The use of citation data in writing the history of science. Institute for Scientific Information, Institute for Scientific Information, Philadelphia, p. 71 (1964)
3. Garfield, E.: The history and meaning of the journal impact factor. J. Am. Med. Assoc. **295**(1), 90–93 (2006)
4. Hirsch, J.E.: An index to quantify an individual's scientific research output. Nat. Acad. Sci. U.S.A. **102**(46), 16569–16572 (2005)
5. Kostoff, R.N.: Performance measures for government-sponsored research: overview and background. Scientometrics **36**(3), 281–292 (1996)
6. Marshakova-Shaikevich, I.: The standard impact factor as an evaluation tool of science fields and scientific journals. Scientometrics **35**(2), 283–290 (1996)
7. Martin, B.R.: The use of multiple indicators in the assessment of basic research. Scientometrics **36**(3), 343–362 (1996)
8. Nakatoh, T., Nakanishi, H., Hirokawa, S.: Journal impact factor revised with focused view. In: Neves-Silva, R., Jain, L.C., Howlett, R.J. (eds.) (KES-IDT 2015). SIST, vol. 39, pp. 471–481. Springer International Publishing, Switzerland (2015)
9. Nakatoh, T., Nakanishi, H., Baba, K., Hirokawa, S.: Focused citation count: a combined measure of relevancy and quality. In: IIAI 4th International Congress on Advanced Applied Informatics (IIAI AAI 2015), pp. 166–170 (2015)
10. Newman, M.E.J.: The structure of scientific collaboration networks. Nat. Acad. Sci. U.S.A. **98**(2), 404–409 (2001)
11. Wuchty, S., Jones, B.F., Uzzi, B.: The increasing dominance of teams in production of knowledge. Science **316**(5827), 1036–1039 (2007)