# GSB’15 Graph Search and Beyond Workshop

Omar Alonso, Marti A. Hearst, Jaap Kamps
(editors)

First International Workshop on Graph Search and Beyond
held at the 38th Annual ACM SIGIR Conference
on August 13, 2013 in Santiago, Chile
http://humanities.uva.nl/ kamps/gsb15/

*SIGIR Workshop on Graph Search and Beyond*'15 Santiago, Chile

Published on CEUR-WS: http://ceur-ws.org/Vol-1393/

## Preface

These proceedings contain the contributed papers of the First International Workshop on Graph Search and Beyond (GSB 2015), held at SIGIR 2015 in Santiago de Chile, on August 13, 2015.

Modern Web data is highly structured in terms of entities and relations from large knowledge resources, geo-temporal references and social network structure, resulting in a massive multidimensional graph. This graph essentially unifies both the searcher and the information resources that played a fundamentally different role in traditional IR, and "Graph Search" offers major new ways to access relevant information. Graph search affects both query formulation (complex queries about entities and relations building on the searcher's context) as well as result exploration and discovery (slicing and dicing the information using the graph structure) in a completely personalized way. This new graph based approach introduces great opportunities, but also great challenges, in terms of data quality and data integration, user interface design, and privacy. We view the notion of "graph search" as searching information from your personal point of view (you are the query) over a highly structured and curated information space. This goes beyond the traditional two-term queries and ten blue links results that users are familiar with, requiring a highly interactive session covering both query formulation and result exploration. The workshop attracted a range of researchers working on this and related topics, and made concrete progress working together on one of the greatest challenges in the years to come.

The workshop consisted of three main parts. First, four keynotes to help us frame the problem, and create a common understanding of the challenges: Rose Marie Philip (Facebook), Swee Lim (Linkedin), Doug Oard (Maryland), and Alex Wade (Microsoft Research). Second, a boaster and poster session with 6 papers selected by the program committee from 8 submissions (a 75% acceptance rate). Each paper was reviewed by at least two members of the program committee. Third, breakout groups on different aspects of exploiting graph search, with reports being discussed in the final session. The papers represent the ideas and opinions of the authors who are trying to stimulate debate. It is the combination of these papers and the debate that will make the workshop a success! We thank the ACM and SIGIR for hosting this workshop, and Diane Kelly, Fernando Diaz and Diego Arroyuelo for their outstanding support in the organization. Thanks also go to the program committee, the authors of the papers, and all participants without whom there would be no workshop.

July, 2015

Omar Alonso
Marti A. Hearst
Jaap Kamps

## Table of Contents

### Overview Paper

### Invited Papers

### Contributed Papers

# Overview of Graph Search and Beyond

Omar Alonso
Microsoft
Mountain View, CA
USA

Marti A. Hearst
UC Berkeley
Berkeley, CA
USA

Jaap Kamps
University of Amsterdam
Amsterdam
The Netherlands

## ABSTRACT

Modern Web data is highly structured in terms of entities and relations from large knowledge resources, geo-temporal references and social network structure, resulting in a massive multidimensional graph. This graph essentially unifies both the searcher and the information resources that played a fundamentally different role in traditional IR, and "Graph Search" offers major new ways to access relevant information. Graph search affects both query formulation (complex queries about entities and relations building on the searcher's context) as well as result exploration and discovery (slicing and dicing the information using the graph structure) in a completely personalized way. This new graph based approach introduces great opportunities, but also great challenges, in terms of data quality and data integration, user interface design, and privacy.

We view the notion of "graph search" as searching information from your personal point of view (you are the query) over a highly structured and curated information space. This goes beyond the traditional two-term queries and ten blue links results that users are familiar with, requiring a highly interactive session covering both query formulation and result exploration. The workshop attracted a range of researchers working on this and related topics, and made concrete progress working together on one of the greatest challenges in the years to come.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation, Search process, Selection process*

## Keywords

Graph search; Semantic search; Personalization; Exploration; Query suggest

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: `http://ceur-ws.org/Vol-1393/`.

## 1. INTRODUCTION

Information on the Web is increasingly structured in terms of entities and relations from large knowledge resources, geo-temporal references and social network structure, resulting in a massive multidimensional graph. This graph essentially unifies both the searcher and the information resources that played a fundamentally different role in traditional IR, and offers major new ways to access relevant information. In services that rely on personalized information like social networks, the graph plays an even more important role, in other words: *you are the query*.

Graph search affects both query formulation as well as result exploration and discovery. On the one hand, it allows for incrementally expressing complex information needs that triangulate information about multiple entities or entity types, relations between those entities, with various filters on geo-temporal constraints or the sources of information used (or ignored), and taking into account the rich profile and context information of the searcher (and his/her peers, and peers of peers, etc). On the other hand, it allows for more powerful ways to explore the results from various aspects and viewpoints, by slicing and dicing the information using the graph structure, and using the same structure for explaining why results are retrieved or recommended, and by whom.

This new graph based information seeking approach introduces great opportunities, but also great challenges, both technical ranging from data quality and data integration to user interface design, as well as ethical challenges in terms of privacy; transparency, bias and control; and avoiding the so-called filter bubbles. Graph search is already available today in many flavors with different levels of interactivity. Social network-based services like Facebook and LinkedIn provide flexibility to search their personal network form many diverse angles. Web search engines like Google and Bing rely more on using graphs to show related content as a mechanism to include other possible contexts for a given query. Clearly, it is not limited to web, and can be applied to other highly structured data. Just to give an example, the hansards or parliamentary proceedings are fully public data with a clear graph structure linking every speech to the respective speaker, their role in parliament and their political party. Graph search allows to explore politics from the viewpoint of individual members of parliament or government.

At a high level, graph search seems limited to familiar entity types (e.g., Facebook entities) and templates. How far can this scale? Will this work on truly open domains? There is a huge potential to use the graph to go beyond recommendations for new friends and contacts or semantically related content. Unlocking the potential of richer knowledge sources for new search strategies requires us to think outside the box, by combining different insights from IR, semantic search, data integration, query expansion and user interfaces to name a few.

The rest of this report is structured in the following way. Section 2 discusses the open research questions raised by searching highly structured data from a personal point of view. Next, in Section 3) we discuss the four keynotes who helped frame the problems and reach a shared understanding of the issues involved amongst all workshop attendees. Rose Marie Philip talked about personalized post search at Facebook, Swee Lim about graph search at Linkedin, Doug Oard about good uses for crummy knowledge graphs, and Alex Wade about Microsoft academic graph. Section 4 discusses the six contributed papers, which were presented in a boaster and poster session. Finally, Section 5 provides preliminary discussion of the results and progress made during the workshop.

## 2. OPEN RESEARCH QUESTIONS

We view the notion of "graph search" as searching information from your personal point of view (you are the query) over a highly structured and curated information space. This goes beyond the traditional two-term queries and ten blue links results that users are familiar with, requiring a highly interactive session covering both query formulation and result exploration.

This raises many open questions:

**IR Theory** What happens if search gets personal? Does this break the classic dichotomy between users and documents, as users are nodes in the social network data themselves? What is the consequence of ultimate personalization, as the local graph differs for all users? As the local graph structure is key, does this obviate the need for large central indexes? Do these types of requests fit in the classic paradigm (e.g., Broder's taxonomy)? How does this shift the balance between the control of the searcher and the ranker over the result set?

**Data Integration** Building a knowledge graph requires massive data integration at many levels: are there trade-offs in simplicity and level of detail (such as the classic knowledge representation trade-off)? What levels of granularity and comprehensiveness are needed for effective deployment? What quality is needed: is any noise acceptable? How to deal with near duplicate detection, conflation, or entity disambiguation?

**Use Cases and Applications** Rather than a universal solution, graph search is particularly useful for specific types of information needs and queries. What are the data and tasks that make graph search works? What kind of scenarios that would benefit from a graph model? In what context can switching perspectives by showing results from the vista of other persons useful?

**Query formulation** How to move from singular queries to highly interactive sessions with multiple variant queries? What new tools are needed to help a searcher construct the appropriate graph search query using refinements or filters to better articulate their needs, or explore further aspects? How can we augment query autocompletion to actively prompt user to interactively construct longer queries exploring different aspects?

**Result Exploration** There is a radical shift towards the control of the searcher—small changes in the query can lead to radically different result sets—how can we support active exploration of slices of the data to explore further aspects? Unlike traditional facetted search options, the result space is highly dynamic, how can we provide adaptive exploration options tailored to the context and searcher, at every stage of the process?

**Evaluation** How do we know the system is any good? How to evaluate the overall process, given its personalized and interactive nature? Can we rely on the direct evaluation of query suggestions and query recommendations? Are there suitable behavioral criteria for in the wild testing, such as longer queries, multiple filters, longer dwell-time, more active engagement, more structured-query templates? Can we use are standard experimental evaluation methods from HCI and UI/UX design?

**Privacy** Access to personal data is fraught ethical and privacy concerns, is there is similarly structured public data for scientific research? As an extreme form of personalization, how to avoid the uncanny cave, filter bubbles and echo chambers? How ethical is it to privilege a particular query refinement suggestion over the many other possible candidates?

Further discussion on the challenges of graph based search can be found in [1].

## 3. KEYNOTES

Four invited speakers helped frame the problems and reach a shared understanding of the issues involved amongst all workshop attendees.

### 3.1 Personalized Post Search at Facebook

The opening keynote was given by Rose Marie Philip (Facebook) on "personalized post search at Facebook" [7].

There are over a billion people and over a trillion posts on Facebook. Among these posts, there are uniquely personalized answers to many search queries. The goal of Facebook post search is to help people find the most personally relevant posts for each individual query, tailored to the content of people's networks. In this talk, I will present some of our work to build a search product that uses personalized graph signals in ranking. I will also give an overview of query modification, posts retrieval and ranking of results.

### 3.2 Graph Search at Linkedin

The second keynote in the morning was given by Swee Lim (Linkedin) on "graph search at Linkedin" [5].

Linkedin is the largest professional social network. Linkedin's graph and search systems help our users discover other users, jobs, companies, schools, and relevant professional information. I will present the evolution of these systems, how they support current use cases, their strengths and weaknesses, our next generation systems, and how we intend to leverage these systems to perform graph searches.

### 3.3 Good Uses for Crummy Knowledge Graphs

The first keynote in the afternoon was given by Doug Oard (University of Maryland) on "good uses for crummy knowledge graphs" [6].

In 1993, Ken Church and Ed Hovy suggested that before we ask how well some new technology meets the need we envision for it, we should pause and first reflect on the question of whether – now that we know something about what can be built – we are envisioning the right uses for what we have. They titled their paper "Good Applications for Crummy Machine Translation" [3]. At about that same time, information retrieval researchers obliged them by (generally without having read their paper) starting to work on cross-language information retrieval; arguably the best application for crummy machine translation ever invented. Now we have some crummy knowledge graphs – and this time we have read the Church and Hovy paper – so perhaps the time is right for us to ask whether we have yet envisioned good uses for crummy knowledge graphs. In this talk, I will seek to seed that discussion.

### 3.4 Overview of Microsoft Academic Graph

The second afternoon keynote was given by Alex Wade (Microsoft Research) with an "overview of Microsoft academic graph" [13]

The Microsoft Academic Graph is a heterogeneous graph containing scientific publication records, citation relationships between those publications, as well as authors, institutions, journals and conference "venues" and fields of study.

## 4. ACCEPTED PAPERS

We requested the submission of short, 4–5 page papers to be presented as boaster and poster. We accepted a total of 6 papers out of 8 submissions after peer review (a 75% acceptance rate).

Jadeja and Shah [4] investigate data driven ways to visualize and navigate graph or tree structured data. Navigating or traversing highly curated graph data is an understudied problem, and hierarchical or tree visualizations can help create order and overview. When visualizing the data from the viewpoint of a particular node makes any graph data (such as social network data) look like a tree with the starting node (a person with all context) as point of origin.

Sabetghadam et al. [8] investigates ways of "reranking" results based on a graph traversal approach for multimodal IR, that is hoped to be robust over different distributions of modalities. The use case of multimodal IR in a curated data space with rich context presents a challenge, as different features and scores on different modalities will be very differently distributed in very different probability spaces. An application of Metropolis-Hastings as sampling/estimation method is suggested as (partial) solution.

Sakamoto et al. [9] investigate captioning or summarizing results in highly curated graph data. Succint descriptions are essential for effective graph exploration, and requires to take the context and structure into account. The paper discusses a particular graph of words, sentences and documents, and also touches upon semantic annotations, which would move the document and text space to an entity space, with all documents and text linked to a particular category or entity.

Santisteban and Cárcamo [10] investigates a variant of the classic Tanimoto or Jaccard similarity measure able to deal with assymmetry in directed graphs and subsumption hierarchies. Similarity measures are central in IR, and related distance measures central in graph data. The discussion is motivated by a use case of "paradigmatic" structures.

Tong et al. [12] investigate category and word relation graphs for retrieving trouble shooting information/documents, addressing the classical IR problem of human assigned controlled terms versus document free text in the context of a curated data space and rich context (at least in principle). The paper offers an interesting graph approach is outlined, mapping terms to categories, for both requests and documents. Making this graph level explicitly available to users offers interesting new possibilities, and opens up ways to map the noisy term occurrence space to the curated, concept and entity based space of the category codes. Hence this paves the way to a semantic, entity based view.

Yu et al. [14] investigates the strength of connections in an entity graph, specifically a scholarly network with a rich entity graph available as public data. This is an interesing use case with a curated data space and rich context, plus an interesting dynamic structure over time. The paper proposes to take the strength (or weakness) of connections into account — here as simulated blind feedback — turning a network into a weighted network of the simple graph into a valued graph.

## 5. CONCLUSIONS

The workshop brought together researchers from a range of areas in information access, who worked together on searching information from your personal point of view over a highly structured and curated information space. One of the main lines of discussion was the considerable industrial activity around social graphs. The most famous example is Facebook Graph Search, a feature that allows users to perform more sophisticated searches on their social network [11]. Bing has been integrating Facebook into their web search results for the last couple of years. Similarly, Google has been annotating search results with Google+ profiles. And all the rest of the search industry is moving in the same direction.

There are also crucial links with work on searching structured data, and work on the appropriate query languages, in particular as part of semantic search. These branches of research in particular focus on complex querying of structured text or data, whereas the graph search addresses also, and perhaps primarily, the process of constructing series of complex queries interactively. This is directly related to exploratory search and sense making. The graph structure provides natural facets for exploring the data, from a local point of view, allowing for a more dynamic structure than traditional faceted search using rigid, global, hierarchical structure. This challenges our understanding of search user interfaces design and evaluation, with search results moving from the found links, to the HIT page as snippets, and now to query suggestion as previews of possible query extensions.

Graph Search has fundamental consequences for information access and offers tremendous opportunities for building new systems and tools that allow users to explore information from many different angles, shifting control back to the user. This is a radical departure from current systems where the machine learning dominate the interaction: the entire information space is determined by the user, and the user is in the driver's seat when expressing her needs and exploring the space of options interactive.

**Acknowledgments** This research is funded in part by the Netherlands Organization for Scientific Research (ExPoSe project, NWO CI # 314.99.108; DiLiPaD project, NWO Digging into Data # 600.006.014).

## References

[1] O. Alonso and J. Kamps. Beyond graph search: Exploring and exploiting rich connected data sets. In *ICWE'15: Engineering the Web in the Big Data Era*, volume 9114 of *LNCS*, pages 3–12. Springer, 2015. URL `http://dx.doi.org/10.1007/978-3-319-19890-3_1`.

[2] O. Alonso, M. A. Hearst, and J. Kamps, editors. *GSB'15: Proceedings of the SIGIR'15 Workshop on Graph Search and Beyond*, 2015. CEUR-WS. URL `http://ceur-ws.org/Vol-1393/`.

[3] K. W. Church and E. H. Hovy. Good applications for crummy machine translation. *Machine Translation*, 8:239–258, 1993. URL `http://dx.doi.org/10.1007/BF00981759`.

[4] M. Jadeja and K. Shah. Tree-map: A visualization tool for large data. In Alonso et al. [2], pages 9–13. URL `http://ceur-ws.org/Vol-1393/`.

[5] S. Lim. Graph search at linkedin. In Alonso et al. [2], page 5. URL `http://ceur-ws.org/Vol-1393/`.

[6] D. W. Oard. Good uses for crummy knowledge graphs. In Alonso et al. [2], page 6. URL `http://ceur-ws.org/Vol-1393/`.

[7] R. M. Philip. Personalized post search at facebook. In Alonso et al. [2], page 7. URL `http://ceur-ws.org/Vol-1393/`.

[8] S. Sabetghadam, M. Lupu, and A. Rauber. Leveraging metropolis-hastings algorithm on graph-based model for multimodal ir. In Alonso et al. [2], pages 14–18. URL `http://ceur-ws.org/Vol-1393/`.

[9] K. Sakamoto, H. Shibuki, T. Mori, and N. Kando. Fusion of heterogeneous information in graph-based ranking for query-biased summarization. In Alonso et al. [2], pages 19–22. URL `http://ceur-ws.org/Vol-1393/`.

[10] J. Santisteban and J. T. Cárcamo. Unilateral jaccard similarity coefficient. In Alonso et al. [2], pages 23–27. URL `http://ceur-ws.org/Vol-1393/`.

[11] N. V. Spirin, J. He, M. Develin, K. G. Karahalios, and M. Boucher. People search within an online social network: Large scale analysis of facebook graph search query logs. In *CIKM'14*, pages 1009–1018. ACM, 2014. URL `http://doi.acm.org/10.1145/2661829.2661967`.

[12] B. Tong, T. Yanase, H. Ozaki, and M. Iwayama. Information retrieval boosted by category for troubleshooting search system. In Alonso et al. [2], pages 28–32. URL `http://ceur-ws.org/Vol-1393/`.

[13] A. D. Wade. Overview of microsoft academic graph. In Alonso et al. [2], page 8. URL `http://ceur-ws.org/Vol-1393/`.

[14] Y. Yu, Z. Jiang, and X. Liu. Random walk and feedback on scholarly network. In Alonso et al. [2], pages 33–37. URL `http://ceur-ws.org/Vol-1393/`.

# Graph Search at Linkedin

## Keynote Abstract

Swee Lim
Linkedin
slim@linkedin.com

## ABSTRACT

Linkedin is the largest professional social network. Linkedin's graph and search systems help our users discover other users, jobs, companies, schools, and relevant professional information. I will present the evolution of these systems, how they support current use cases, their strengths and weaknesses, our next generation systems, and how we intend to leverage these systems to perform graph searches.

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: `http://ceur-ws.org/Vol-1393/`.

# Good Uses for Crummy Knowledge Graphs

## Keynote Abstract

Douglas W. Oard
University of Maryland
College Park, MD
USA
oard@umd.edu

## ABSTRACT

In 1993, Ken Church and Ed Hovy suggested that before we ask how well some new technology meets the need we envision for it, we should pause and first reflect on the question of whether – now that we know something about what can be built – we are envisioning the right uses for what we have. They titled their paper "Good Applications for Crummy Machine Translation" [1]. At about that same time, information retrieval researchers obliged them by (generally without having read their paper) starting to work on cross-language information retrieval; arguably the best application for crummy machine translation ever invented. Now we have some crummy knowledge graphs – and this time we have read the Church and Hovy paper – so perhaps the time is right for us to ask whether we have yet envisioned good uses for crummy knowledge graphs. In this talk, I will seek to seed that discussion.

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

## References

[1] K. W. Church and E. H. Hovy. Good applications for crummy machine translation. *Machine Translation*, 8:239–258, 1993. URL http://dx.doi.org/10.1007/BF00981759.

# Personalized Post Search at Facebook

## Keynote Abstract

Rose Marie Philip
Facebook Inc.
Menlo Park, CA
USA
rosephilip@fb.com

## ABSTRACT

There are over a billion people and over a trillion posts on Facebook. Among these posts, there are uniquely personalized answers to many search queries. The goal of Facebook post search is to help people find the most personally relevant posts for each individual query, tailored to the content of people's networks. In this talk, I will present some of our work to build a search product that uses personalized graph signals in ranking. I will also give an overview of query modification, posts retrieval and ranking of results.

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

# Overview of Microsoft Academic Graph

Keynote Abstract

Alex D. Wade
Microsoft Research
Redmond, WA
USA
Alex.Wade@microsoft.com

## ABSTRACT

The Microsoft Academic Graph is a heterogeneous graph containing scientific publication records, citation relationships between those publications, as well as authors, institutions, journals and conference "venues" and fields of study.

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

# TREE-MAP: A VISUALIZATION TOOL FOR LARGE DATA

Mahipal Jadeja
DA-IICT
Gandhinagar,Gujarat
India
Tel:+91-9173535506
mahipaljadeja5@gmail.com

Kesha Shah
DA-IICT
Gandhinagar,Gujarat
India
Tel:+91-7405217629
kesha.shah1106@gmail.com

## ABSTRACT

Traditional approach to represent hierarchical data is to use directed tree. But it is impractical to display large (in terms of size as well complexity) trees in limited amount of space. In order to render large trees consisting of millions of nodes efficiently, the Tree-Map algorithm was developed. Even file system of UNIX can be represented using Tree-Map. Definition of Tree-Maps is recursive: allocate one box for parent node and children of node are drawn as boxes within it. Practically, it is possible to render any tree within predefined space using this technique. It has applications in many fields including bio-informatics, visualization of stock portfolio etc. This paper supports Tree-Map method for data integration aspect of knowledge graph. Social customer relationship management (CRM) tree-map example is briefly used to explain how data integration is supported by tree-maps. In this paper, key features of Tree-map are discussed briefly including expressive power of tree-map and types of queries supported by it. As an example of social network visualization, how twitter tree-maps can be used to answer dynamic queries interactively is also discussed in detail.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval: Search Process.

## General Terms

Algorithms, Management, Measurement, Documentation, Performance, Design, Reliability, Experimentation, Human Factors,Theory

## Keywords

Tree-Map, Large Data Visualization, NewsMap, Dynamic Query in Tree-Map, Social CRM Tree-Map,Twitter Tree-Map

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/

## 1. INTRODUCTION

Tree-Maps are used to present hierarchical information on 2-D[1] (or 3-D [2]) displays. Tree-maps offer many features: based upon attribute values users can specify various categories, users can visualize as well as manipulate categorized information and saving of more than one hierarchy is also supported [3].

Various tiling algorithms are known for tree-maps namely: Binary tree, mixed treemaps, ordered, slice and dice, squarified and strip. Transition from traditional representation methods to Tree-Maps are shown below. In figure 1 given hierarchical data and equivalent tree representation of given data are shown. One can consider nodes as sets, children of nodes as subsets and therefore it is fairly easy to convert tree diagram into Venn diagram. Figure 2 represents Venn diagram and its equivalent representation as nested tree-map. Nested tree-map represents the nesting of rectangles. Finally in figure 3, tree-map representation of given hierarchical data is shown [4]. Tree-map is a comprehensive design in which a border is used to show nesting and it is more space efficient compared to nested version. Key advantages of tree-maps are easy identification of patterns and efficient usage of space.

Queries related to space can be answered easily with the use of tree-map visualization. Consider tree-map representation of operating system say UNIX. With the help of this representation, one can easily answer following queries: Identification of directory which is taking up most of space, how much amount of space is taken up by specific directories, types of files present in hierarchy etc. Tree-maps offer dynamic visualization. Key features of dynamic visualization are: immediate feedback mechanism, support for dynamic queries( queries which are incremental and reversible). In Section 5, types of queries supported by tree-map are explained in detail. Tree-maps can be used to represent complex social networks. Possible approaches to deal with dynamic queries during highly interactive sessions are discussed in subsection 6.1 for such types of tree-maps.

Intuitively, tree-map representation is better than simple manual list representation. Peet is a San Francisco Bay Area based famous coffee roaster as well as retailer since 1966. A marketing survey showed following result: For 92 out of 100 customers of peet, (who used the tree-map interface) online shopping was easy. Whereas for the manual lists users, this percentage was only 12. Tree-map interface of peet is shown in figure 4. Most of the other techniques of data visualization were invented in the absence of widely-available computational(computer) resources. Tree-maps were conceived as a result of computerization and therefore they have crucial benefits from this more modern scenario.

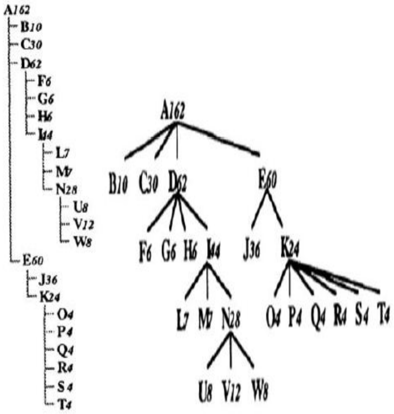


**Figure 1: Hierarchical data and Corresponding Tree Representation**

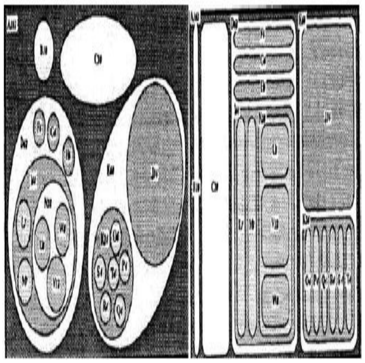


**Figure 2: Venn Diagram and Nested Tree-Map**

## 2. GUIDELINES FOR TREE MAP DESIGN

1. Every box of the tree-map can display two different measures namely size and color. Size should reflect quantity measure whereas color is used to display measure of performance and/or change. i.e. satisfaction of customer, growth rate etc.
2. In selection of tree-map layouts, extreme aspect ratios should be avoided [5].
3. Tree-maps are more suitable for high density data, for low density one can use bar charts.
4. Comparing non-leaf nodes is easier in tree-maps compared to bar charts.
5. Appropriate labels should be given and labels should be meaningful.
6. It is advisable to show labels only when user rolls over a tree-map box.

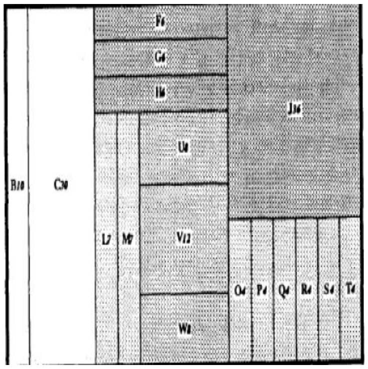
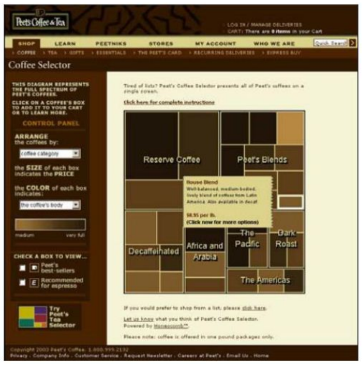


**Figure 3: Tree-Map**

7. Labels must be visible in multicolored background of tree-map.
8. Depending upon the nature of the color measure, one sided/two sided color range should be used.
9. In order to show correlation, highlighting should be used.
10. One can use animation in tree-maps to show change in the data.
11. Simple presentation method (Tooltip window/sidebar) can be used to show node detail.



CART: There are 0 items in your Cart

THIS DIAGRAM REPRESENTS THE FULL SPECTRUM OF PEET'S COFFEES.

CLICK ON A COFFEE'S BOX TO ADD IT TO YOUR CART OR TO LEARN MORE.

ARRANGE the coffees by: coffee category

the SIZE of each box indicates the PRICE

the COLOR of each box indicates: the coffee's body

CHECK A BOX TO VIEW...

□ Peet's best-sellers

□ Recommended for espresso

Tired of lists? Peet's Coffee Selector presents all of Peet's coffees on a single screen.

Click here for complete instructions

House Blend
Well-balanced, medium-bodied, lively blend of coffees from Latin America. Also available in decaf.

$8.95 per lb.
(Click now for more options)

If you would prefer to shop from a list, please click here.

Let us know what you think of Peet's Coffee Selector.
Powered by Honeycomb™.

Please note: coffee is offered in one pound packages only.

Copyright 2003 Peet's Coffee. 1.800.999.2132
Privacy . Company Info . Customer Service . Request Newsletter . Careers at Peet's . Email Us . Home

**Figure 4: www.peets.com**

## 3. EXPRESSIVE POWER OF TREE-MAP

Tree-Maps are used to express a variety of nested as well as hierarchical data and data structures. In general, type of tree-map representation depends upon application and type of data hierarchy.

"Tree-map visualization generator" are used to display tree-maps for arbitrary hierarchical data. Tree-Maps can be provided as images in static forms or they can be used to provide interactive features (like zooming into small area of hierarchy) in applications. Tree-maps support browser as well as rich client applications. In one of the applications, tree-maps are incorporated with Windows Forms- Microsoft Corporation.

Tree-Maps are also famous amongst news designers. Examples are listed below.

1. NewsMap[6] (Newsmap.jp is developed by Marcos Weskamp and it represents current items of Google News using interactive Tree-map which is shown in figure 5.)

2. London 2012 Olympics and Tree- maps [7]

3. BBC News- SuperPower: Visualising the internet

4. The New York Times- Obama's budget proposal (Year 2011)

5. CNN Twitter buzz of South Africa (Year 2010)

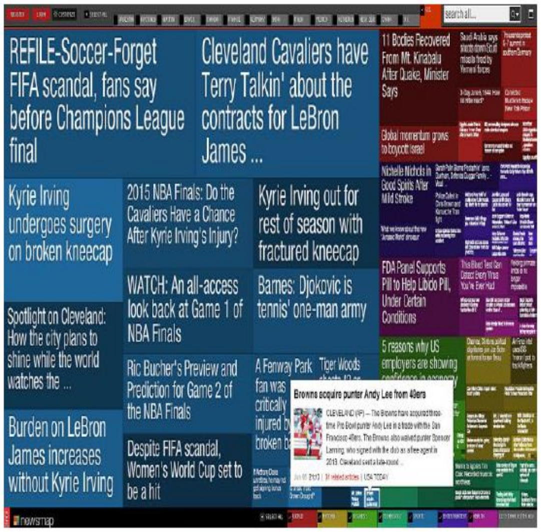

search all...
REFILE-Soccer-Forget FIFA scandal, fans say before Champions League final
Cleveland Cavaliers have Terry Talkin' about the contracts for LeBron James ...
11 Bodies Recovered From Mt. Kinabalu After Quake, Minister Says
Saudi Arabia says shoots down Scud missile fired by Yemeni forces
Global momentum grows to boycott Israel
Kyrie Irving undergoes surgery on broken kneecap
2015 NBA Finals: Do the Cavaliers Have a Chance After Kyrie Irving's Injury?
Kyrie Irving out for rest of season with fractured kneecap
Nichelle Nichols in Good Spirits After Mild Stroke
FDA Panel Supports Pill to Help Libido Pill, Under Certain Conditions
This Blood Test Can Detect Every Virus You've Ever Had
WATCH: An all-access look back at Game 1 of NBA Finals
Barnes: Djokovic is tennis' one-man army
Spotlight on Cleveland: How the city plans to shine while the world watches the ...
5 reasons why US employers are showing confidence in economy
Ric Bucher's Preview and Prediction for Game 2 of the NBA Finals
A Fenway Park fan was critically injured by broken b
Tiger Woods
Browns acquire punter Andy Lee from 49ers
CLEVELAND (AP) — The Browns have acquired three-time Pro Bowl punter Andy Lee in a trade with the San Francisco 49ers. The Browns also waived punter Spencer Lanning, who signed with the club as a free agent in 2013. Cleveland sent a late-round ...
31 related articles | USA TODAY
Burden on LeBron James increases without Kyrie Irving
Despite FIFA scandal, Women's World Cup set to be a hit
newsmap

**Figure 5: www.newsmap,jp**

## 4. SOCIAL NETWORK DATA AND TREE-MAP

For the promotion of brand, role of marketer is not significant in the modern era of social media. In the past, information was produced by marketers and consumed by customers. Currently more information is generated by customers about brands on social media including blogs, social media networks, online forums etc. Currently marketing teams are struggling in analysis of this online information, which is required for prediction of acceptance rates of products, patterns of purchase and level of satisfaction in customers. Marketers can use these new channels for promotion by developing customers as brand advocates.

For travel as well as hospitality industry, decisions related to purchase are mainly determined by online reviews as well as recommendations. Online customer data along with business functions information forms an integrated database. In order to study levels of customer loyalty, study of this integrated database is necessary.

It is possible to use customer tree-map for segmenting customers and generation of 'brand score' for customers and brand score depends upon 1) Brand engagement of customer-behavioral aspect and 2) Attitudes of customers.

Two different types of score namely spend value score and advocacy score are calculated using integrated database (traditional CRM and unstructured data). Social CRM tree-map can be created by plotting these scores (by integrating two data-sets) on a 2-D axis[9]. Example of social CRM tree-map is given in figure 6. Members without any spend value are defined as noncustomers. This tree-map is useful for calculation of overall "customer brand score".

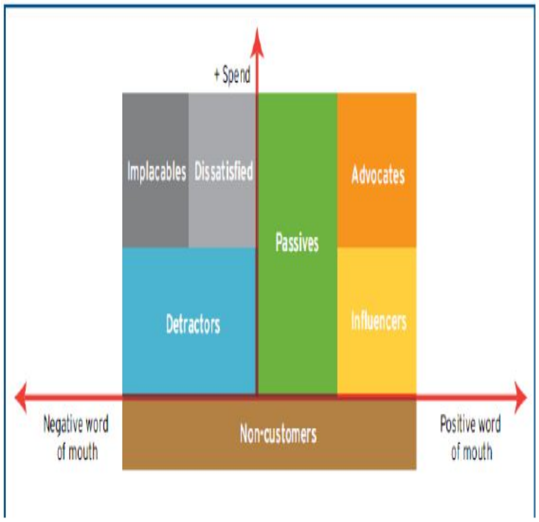


**Figure 6: Social CRM Tree-Map**

Advocates have following qualities: They have high values for spend value as advocacy score. They are brand evangelists and their behavior as well as attitude is very loyal to brand.

After successful development of Tree-map, organizations can take actions in order to cultivate advocates of brand.

## 5. TYPES OF QUERY SUPPORTED BY TREE-MAPS

Tree-Maps provide two important features by supporting dynamic queries:

1. Querying a large set of data.

2. To find out patterns in large data set. [10][11]

In tree-maps, dynamic queries are implemented using radio buttons, buttons and sliders. Tree-map follows principle of direct manipulation for searching in large database.
Key features of query processing of Tree-Map are listed below:

- Supports visual representation. (for components of query)

- Supports visual representation of query results

- Provides rapid, reversible and incremental control of query.

- Selection is done by just pointing, not by typing.
- Tree-map provides immediate as well as continuous mechanism of feedback

## 6. TREE-MAP FOR TWITTER DATA VISUALIZATION

Key requirements for visualization of any social network are listed below:

- Identification of the actors-members of the social network.
- Visualization should represent relationships of various types.
- Visualization should support aggregated as well as structured view of the complex social network.

Consider example of Twitter network with four sample actors namely Steve, John, Luke and Adam. Figure 7 represents this network as a Tree-map. Tree-map offers all the crucial features which are desirable for visualization tool. Here actors are represented by rectangles and size of each rectangle is proportional to the total number of tweets sent by that particular actor. The friendship relationship is represented by a common edge between two rectangles. In our example. rectangle corresponding to Luke has highest area which implies highest number of tweets amongst the four users. No common edge is present between Steve and Luke which implies that they are not friends in Twitter.

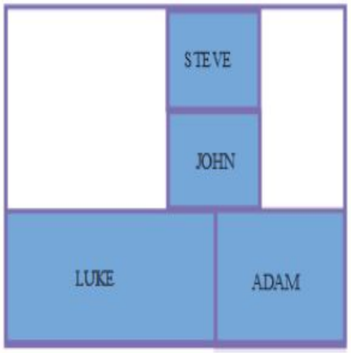


**Figure 7: Basic Twitter Treemap**

Other variants of Twitter tree-map are also shown in figure 8 and 9. Tree mapping is not as popular as other visualization techniques still recent survey results are encouraging for twitter tree-maps [12]. Better results are possible by improving current design of tree-maps as well as integration of tree-map with other visualization techniques.

### 6.1 Discussion on Interactivity of Tree-maps

Tree-map offers interactive feature which is distinctive. The main objective of this visualization tool is to provide interactive display on a computer screen. Because of this unique feature, one can explore the data hierarchy effortlessly and simultaneously decent level of estimation is also possible for quantitative aspects of the information. In order to provide element specific information in detail, various tree-map soft wares offers computer screen mouseovers using which the user can get specific information just by placing the computer mouse over the specific box. Because of these crucial interactive features, tree-map is emerging powerful visualization technique-also for large social datasets because real time feedback is essential in the case of complex social network. Due to this interactivity feature, the analyst has the ability to traverse the tree and he can also present categorical data view at every level.

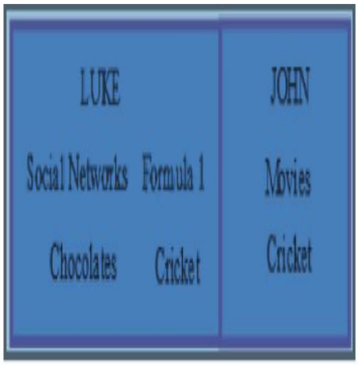


**Figure 8: Twitter Treemap with additional information (Actor's interests)**

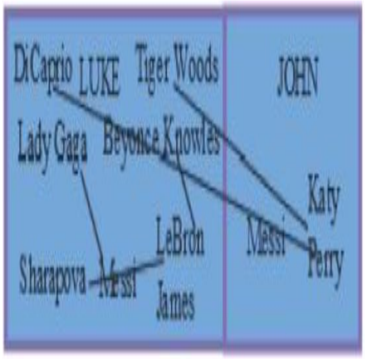


**Figure 9: Twitter Treemap integrated with Network diagram**

Generally, queries on social network data focus more on relationships between different groups and size of particular category is often very common type of query. For example, which country has highest number of tweeter users? Now consider one complex query: Do white males in the North America use the twitter more than white females in the South? In order to answer this question one has to consider sub-questions for all data points. i.e. whether a particular person is black/white, whether he has twitter account or not and so on.

In order to answer these queries interactively for categorical social data, we propose the use of CatTrees.(enhancement of tree-maps) [13] It is possible to answer these types of question easily if the data has hierarchy because then for each possible answer pattern one can allocate leaf node with counter and to get final answer, the analyst can follow two different paths (depending upon query) from root to leaf nodes and give final result depending upon the comparison of counters. So depending upon query, new hierarchy may be required every time. In short, dynamic hierarchies are required to support dynamic queries! Dynamic hierarchies are implemented by CatTrees.

All social data is not hierarchical in nature. Surprisingly tree-maps can be used to visualize non-hierarchical data too. In this case, imaginary hierarchy is provided as an input by the analyst [14].

## 7. CONCLUSION AND FUTURE WORK

Speed of data accessing is very crucial parameter for any visualization tool. Tree-maps should support hardware or parallel processing or grid computing approach for better results.

Overall design of tree-maps should be modified for offering better understandings of the data. Data accuracy is also equally important along with decent data accessing speed. Tree-maps should offer better meaningful results for various queries. Nowadays tree-maps are famous at the academy but they are not accepted as a general hierarchical tool. Tree-maps have various drawbacks: 1) Specific use 2) Lack of cognitive plausibility 3) Poor performance (Task-driven) 4) Average aesthetic qualities. Currently ongoing research in this area is trying to solve these issues.

Tree-maps are very useful tools for identification of extreme values in large database as well as primary trends. They are not meant for comparison of values precisely mainly because of two dimensional limited area and color encoding. Tree-maps are successful and can be understood easily by public.

## 8. ACKNOWLEDGMENTS

We would like to specially thank and acknowledge Dr. Jaap Kamps for motivating us and providing quality inputs. We would also like to convey our regard to SIGIR team for organizing GSB'15-the first international workshop for graph search and beyond, enabling us to participate and give us a chance to contribute to the community to the best of our abilities.

## 9. REFERENCES

1. Shneiderman, B. (1992). Tree visualization with tree-maps:2-d space-filling approach. ACM Transactions on graphics (TOG), 11(1),92-99.

2. Bladh, T., Carr, D. A., & Scholl, J. (2004, January). Extending tree-maps to three dimensions: A comparative study. In Computer Human Interaction (pp. 50-59). Springer Berlin Heidelberg.

3. Bederson, B. B., Shneiderman, B., & Wattenberg, M. (2002).Ordered and quantum tree-maps: Making effective use of 2D space to display hierarchies.AcM Transactions on Graphics (TOG), 21(4), 833-854.

4. Johnson, B., & Shneiderman, B. (1991, October). Tree-maps:A space-filling approach to the visualization of hierarchical information structures. InVisualization, 1991.Visualization'91, Proceedings., IEEE Conference on (pp.$284-291$). IEEE.

5. Kong, N., Heer, J., & Agrawala, M. (2010). Perceptual guidelines for creating rectangular tree-maps. Visualization and Computer Graphics, IEEE Transactions on, 16(6), $990-998$.

6. Ong, T. H., Chen, H., Sung, W. K., & Zhu, B. (2005). Newsmap: a knowledge map for online news. Decision Support Systems, 39(4), 583-597.

7. Field, K. (2012). Mapping the London 2012 Olympics. The Cartographic Journal, 49(3), 281-296.

8. Cao, N., Lin, Y. R., Sun, X., Lazer, D., Liu, S., & Qu, H. (2012). Whisper: Tracing the spatiotemporal process of information diffusion in real time.Visualization and Computer Graphics, IEEE Transactions on, 18(12), 2649-2658.

9. Vijay Raghunathan(2014) Moving Beyond Social CRM with Customer Brand Score Available: http://www.cognizant.com/InsightsWhitepapers/Moving-Beyond-Social-CRM-with-the-Customer-Brand-Score.pdf

10. Deussen, O., Hansen, C., Keim, D. A., & Saupe, D. Interactive Tree-maps With Detail on Demand to Support Information Search in Documents.

11. Schlechtweg, S., Schulze-Wollgast, P., & Schumann, H.(2004, May). Interactive tree-mapswith detail on demand to support information search in documents. In Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization (pp. 121-128) .Eurographics Association.

12. Sathiyanarayanan, M., & Burlutskiy, N. Design and Evaluation of Euler Diagram and Treemap for Social Network Visualisation.

13. Kolatch, E., & Weinstein, B. (2001). Cattrees: Dynamic visualization of categorical data using treemaps. Project report.

14. Sirin, E., & Yaman, F. (2002). Visualizing dynamic hierarchies in treemaps.

# Leveraging Metropolis-Hastings Algorithm on Graph-based Model for Multimodal IR

Serwah Sabetghadam, Mihai Lupu, Andreas Rauber
Institute of Software Technology and Interactive Systems
Vienna University of Technology
Vienna, Austria

## ABSTRACT

The velocity of multimodal information shared on web has increased significantly. Many reranking approaches try to improve the performance of multimodal retrieval, however not in the direction of true relevancy of a multimodal object. Metropolis-Hastings (MH) is a method based on Monte Carlo Markov Chain (MCMC) for sampling from a distribution when traditional sampling methods such as transformation or inversion fail. If we assume this probability distribution as true relevancy of documents for an information need, in this paper we explore how leveraging our model with Metropolis-Hastings algorithm may help towards true relevancy in multimodal IR.

## Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: General; H.3.3 [**Information Search and Retrieval**]: Metrics—*Retrieval models, Search process*

## General Terms

Theory, Algorithm

## Keywords

IR, Multimodal, Graph, Metropolis-Hastings

## 1. INTRODUCTION

There are many challenges in multimodal information retrieval. Mei et al. [8] have performed a survey on reranking models of multimodal information retrieval. They divide the related work in four categories: 1) *Self-reranking*: includes reranking methods that include data from the original ranking result such as Pseudo-Relevance Feedback or learning a ranking model by giving top ranked documents as positive. 2) *Example-based reranking*: methods to understand the query using accompanying examples. 3) *Crowd reranking*: leverages crowd-sourced knowledge on the web to mine relevant patterns for a query. 4) *Interactive Reranking*: in this case a user can edit a part of the search results (to delete or to emphasize).

Graph-based methods for reranking are a subset of Self-reranking category, in which a graph oriented search is performed based on relations between objects. Mostly, related work in this area is performed on images/videos with similarity links between them [11, 5]. The use of results from independent modality indexing neglect that data objects are interlinked through different relations. The problem becomes more challenging when the graph is multimodal. During traversal, we may see information objects from different modalities (text, audio, video or image). We propose a model to utilize probabilistic model of IR in multimodal retrieval, with the goal of approaching true relevancy rather than just a reranking. This means that a query may have null result because of lack of any relevant data. According to probability ranking principle in IR, the relevancy of a document to a query is defined as $p(d|q) = \frac{p(q|d)p(d)}{p(q)}$. This requires the probabilities of $p(q)$ and $p(d)$ which are not available. Different ranking models like TF.IDF, BM25 or LM aim to probe the true ranking through different models on $p(q|d)$.

In this paper, we explore the capability of our model to approach probabilistic IR for multimodal retrieval with the help of the MH algorithm. MH is based on MCMC and is used in cases where it is hard to sample from a probability distribution. Assuming the true probability distribution of relevancy of documents to the query as stationary distribution, utilizing MH we make a Markov-chain of documents which results in the same stationary distribution of probabilities. We conduct the experiments on ImageCLEF2011 Wikipedia collection as a multimodal collection.

## 2. RELATED WORK

There are many efforts in multimodal retrieval in combining textual and visual modalities. Martinent et al. [7] propose to generate automatic document annotations from inter-modal analysis. They consider visual feature vectors and annotation keywords as binary random variables. Jing et al. [6] employ the PageRank to rerank image search. The hyperlinks between images are based on visual similarity of search results. Yao et al. [11] make a similarity graph of images and find authority nodes as result for image queries. Through this model, both visual content and textual information of the images is explored. Hsu et .al [5] leverage context reranking as a random walk over a graph of video stories. The links are based on similarities between different

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: `http://ceur-ws.org/Vol-1393/`.

video stories. The final scoring value is a combination of initial text and stationary distribution scores.

The application of MH method in information retrieval, is limited to search in peer-2-peer networks [3, 1]. Ferreira et al. [3] have designed a protocol for locating a specific object regardless of the topology of the network through uniform sampling from peer-to-peer networks. Zhong et al. [12] use random walks and focus on convergence time for different network sizes. They investigate the probability distribution of visiting nodes. In order to go beyond peer-2-peer networks and apply MH in IR, we need a jumping distribution, i.e. weighted links between nodes. Such links may be similarity/semantic or a mixture of the two. The difficulty, as we will see, is ensuring the stochastic and ergodic nature of the chain.

## 3. MH ALGORITHM

MH is one of the algorithms based on MCMC to obtain samples from a complex probability distribution $\pi(x)$. The goal is to draw samples form $\pi(x)$ where $\pi(x) = \frac{\widetilde{\pi}(x)}{K}$. The normalizing variable $K$ is unknown and hard to compute. Based on the jumping distribution matrix of $W$, MH algorithm generates a sequence from this distribution as follows:

1. Start with initial value $x$ that $\pi(x) > 0$
2. Using the current $x$ value, sample a candidate point $y$ from $W(x, y)$.
3. The transition probability then is made according to

$$Pr(x, y) = W(x, y)\lambda(x, y) \tag{1}$$

$$\lambda(x, y) = min\left[\frac{\widetilde{\pi}(y).W(y, x)}{\widetilde{\pi}(x).W(x, y)}, 1\right] \tag{2}$$

Note that $\lambda(x, y)$ does not require knowledge of the normalizing constant because $\pi(y)/\pi(x)$ drops it out. If it increases the density ($\lambda > 1$), accept y and set the next sample $x_t = y$. Repeat step 3. If it decreases the density, sample $u$ from uniform (0,1). Accept if $\lambda > u$, else reject it.

In order to reach a unique equilibrium state for a Markov-chain, it should be ergodic, satisfying irreducibility (for any state, the probability of getting there given any starting point is more than zero) and aperiodicity (there is no rhythm in which states can be reached given a starting point). There may be different proposal distributions for MH. Two general approaches are [10]: 1) Random walks - the new state y is dependent to the current state x. 2) Independent sample finding - the probability of jumping to point y is chosen from a distribution of interest, independent of the current value. This method is usually used in asymmetric MH. We use the first approach in our work.

## 4. MODEL REPRESENTATION

We define a graph-based model $G = (V, E)$, in which $V$ is the set of information objects and their facets, and $E$ is the set of edges. By facet we mean inherent feature or representation of an object (e.g., tf.idf facet of a document or edge histogram of an image). Each object may have a number of facets. We define four types of relations. Their characteristics are discussed in detail in [9]. We formally define the relation types and their weights as follows:

- **Semantic** ($\alpha$): any semantic relation between two objects in the collection (e.g. the link between lyrics and a music file). The edge weight $w_{xy}$ is made inversely proportional to the $\alpha$-out-degree of the source node $u$ and $w_{xy} = 1/N_x^{(\alpha)}$.

- **Part-of** ($\beta$): a specific type of semantic relation, indicating an object as part of another object, e.g. an image in a document. The weight is 1 because of containment relation as an object part of another one.

- **Similarity** ($\gamma$): relation between the facets of two information objects. The weight is the similarity value between the facets.

- **Facet** ($\delta$): linking an object to its representation(s). It is a unidirectional relation from facet to the parent object. Weights are given by perceived information content of features, with respect to the query type.

Our scoring method consists of two steps: 1) In the first step, we perform an initial search with Lucene and/or Lire result based on the facets. This provides us a set of activation nodes. 2) In the second step, using the initial result set of data objects (with normalized scores) as seeds, we exploit the graph structure and traverse it.

The model can perform both partial/whole facet retrieval. We may decide to search e.g. only based on query textual or visual facets, or based on all query facets. In practice, we make a form of late facet fusion by combination of different scores and giving one score to the parent information object. However, it is not in the traditional way of late fusion. Since we do not make the result rank list out of top ranked nodes. We initiate their scores in graph nodes and then start propagation. In our model, facet fusion is implicitly calculated by matrix multiplication and final vector computation.

## 5. MH MAPPED TO IR

We want to achieve a query dependent stationary distribution such that the probability in node $x$ is proportional to the probability that this node is relevant to the query, and at any other node (non-relevant) the probability is zero. This is the $\pi(x)$ distribution from which we cannot directly sample. Instead, we have the $\widetilde{\pi}(x)$ which could be a relevance scoring function (e.g. a BM25 score between the data object $x_i$ and the query). MH would formally provide us with a method to sample from the probability distribution, if the approximate probability $\widetilde{\pi}$ is properly chosen.

We have the graph of different relations in the adjacency matrix $W$. Assuming the true relevancy of nodes to the query as $\pi(x)$, we define the $\tilde{\pi}(x)$ as relevance score value function ($RSV$). A node (M) in the graph may be of any modality: Text (T), Image (I), Audio (A) or Video (V), and the query (Q) may be combination of different modalities. We define the relevance score value function ($RSV$), as follows:

$M \in \{T, I, V, A\}$
$M = \cup_{i=1}^{n} M_{f_i}$
$Q = \cup_{j=1}^{m} Q_{f_j}$
$l = |\{Q_f | Q_{f_i} = M_{f_i}\}|$

$$RSV(Q,M) = \sum_{i=1}^{l} norm(sim(\overline{Q_{f_i}}, \overline{M_{f_i}})).w_{f_i} \qquad (3)$$

where $n$ is the number of facet types of the information object node, $m$ is the number of facet types of the query, $sim$ is the similarity function between two facets, $norm$ is the normalizing function and $w_{f_i}$ is the weight of facet $f_i$ for this query. We compute the similarity ($sim$) between $l$ number of the same facets of this information object and the query, in which $\overline{Q_{f_i}}$ and $\overline{M_{f_i}}$ are the value of corresponding facets. Usually the value of a facet is in the form of a feature vector. In case of no common facet, the $sim$ function output is zero. Relevancy of an information object to a query should be calculated in accordance to other information objects. For this purpose we compute the similarity of all objects for each query facet and normalize. As we have a multimodal graph and in each step may visit a node with different modality, we require a normalized value to be able to compare the relevancy values.

Different modalities have different facets. Reaching nodes with the same modality of query examples, we have all the facets in common (e.g. an image query and an image node). Visiting nodes with different modality than query examples, we perform similarity for common facets. For instance, if we have an audio object and an image query, we can compare their textual facets (the tf.idf facet of image metadata and tf.idf facet of the audio tags or lyrics).

## 5.1 MH Constraints in Astera

**Irreducibility**: To check irreducibility we should prove that our graph is connected. By adding different relations of $\beta$, $\gamma$ and $\alpha$, we have a connected graph. For this purpose, starting from top ranked results for a sample query we traverse the graph. In each step we visit new neighbours and continue until we see no more new nodes. The number of nodes seen in this traversal was the whole graph size. This observation, even for one query, indicates the connectivity of our graph.

**Aperiodicity**: Finding nodes from a starting point is not multiple of a number in our graph. We satisfy this constraint by construction.

**Stochastic property**: According to the weight definition in Astera for $\beta$ links, the sum of weights on a row may be more than one. However, semantic ($\alpha$) and/or similarity ($\gamma$) links can be used in a normalized form, complying with stochastic property.

**Transition Function in Astera** According to Metropolis-Hasitngs algorithm, and Eq. 2 we sample from $W(x,y)$ and accept the move with probability $\lambda(x,y)$. This implies on how we define high-order transition probabilities after $t$ steps: $Pr_q^{t+1}(x,y) = \sum_{i=1}^{k} Pr_q^t(x,z_i)(z_i,y)$ where q is the query, k is the number of common nodes z between x and y, and $Pr^t$ is the transition probability of starting from $x$ and moving $t$ steps further.

**Mixing** Walsh divides the mixing chains in two categories of **poorly mixing** and **well mixing** chains [10]. To prevent poorly mixing, one usual way is to use Simulated Annealing method with high jumps. Second option is to start with several chains to cover the space to find nodes. Our model follows the second option, as we start from different starting points according to standard search result for each facet.

## 5.2 Role of MH in Adjusting the Weights

In principle, MH either accepts a jumping density of $W(x,y)$ (when $\lambda > 1$) and keeps the value and moves forward, or modifies the weight with the factor of $\lambda$. The new value of this link for next step is $W(x,y) \cdot \lambda$. According to stochastic property, the sum of the weights of links of an edge is 1. In each step, when weights are adjusted by MH, the sum may get lower than 1. In this case the link is accepted with probability of $\lambda < 1$. The decreased value is given as self-transitivity value to the node, indicating staying in this state is preferred than choosing that specific link. Performing this for many steps, loosens the links with less relevant neighbours and keeps the links with increasing relevancy neighbours. This way, MH may modify the weights in the direction of making a Markov chain which reaches to the true probability distribution.

To prevent poorly mixing, we start from different starting points according to standard search result for each facet. These points satisfy the condition of $\widetilde{\pi}(x) > 0$ as it is the scored ranked result.

# 6. EXPERIMENT DESIGN

We applied the ImageCLEF 2011 Wikipedia collection for imgae retrieval task. Each image has one metadata file that provides information about name, location, one or more associated parent documents in up to three languages (EN, DE and FR), and textual image annotations (i.e. caption, description and comment). The collection consists of 125,828 documents and 237,434 images. We parsed the image metadata and created nodes for all parent documents, images and corresponding facets. We created different relation types: the $\beta$ relation between parent documents and images (as part of the document), and $\delta$ relation between information objects and their facets. We use the 50 English query topics.

## 6.1 Document and Image Facets

In the first phase of our hybrid search, we use standard indexing results both for documents and images. The computed scores in both modalities are normalized per topic between (0,1) based on min-max method. Different indexings based on different facets are:

- **Text tf.idf facet**: We utilize default Lucene indexer, based on tf.idf, as text facet.
- **Image textual annotation tf.idf facet (Metadata)**: We use metadata information of the images caption, comment and description), as image textual facets.
- **CEDD facet**: For image facets, we selected the Color and Edge Directivity Descriptor (CEDD) feature since it is considered the best method to extract purely visual results [2].

In the second phase, starting from standard indexed results, we conduct the graph search based on MH. In this instantiation of Astera, we use only $\beta$ links between the documents and images. We investigate adding $\alpha$ and $\delta$ link types are in our future works.

## 6.2 Transition Matrix in Astera

To compute the transition matrix $Pr$, we need to compute the $\lambda(x,y)$ for each two neighbour nodes to update the weights. In this instantiation of Astera with ImageCLEF

2011 Wikipedia collection, we have images and documents node types. The query topic in this collection is multimodal. It is a combination of keywords and image examples with facet set of $\{tf.idf, CEDD\}$.

Based on any of these facets, we can start traversal in the graph. For example, if we start from similarity with metadata tf.idf results, we will have a set of images as starting points to make the traversal. In this instantiation of Astera, an image object (I) has two facets of $\{tf.idf, CEDD\}$. The common set of facets of $l$ between the query and image is $l = \{tf.idf, CEDD\}$. Each image is connected to at least one parent document (D) through $\beta$ link. To compute the $Pr(I, D) = W(I, D) \cdot \lambda(I, D)$, we need the $\lambda$ value, which is:

$$\lambda(I, D) = \left[\frac{RSV(Q, D)}{RSV(Q, I)} . \frac{W(D, I)}{W(I, D)}, 1\right] \quad (4)$$

where

$$\begin{aligned} RSV(Q, I) = norm(sim(\overline{Q_{tf.idf}}, \overline{I_{tf.idf}})).w_{tf.idf} + \\ norm(sim(\overline{Q_{CEDD}}, \overline{I_{CEDD}})).w_{CEDD} \end{aligned} \quad (5)$$

and

$$RSV(Q, D) = norm(sim(\overline{Q_{tf.idf}}, \overline{D_{tf.idf}})).w_{tf.idf} \quad (6)$$

The $RSV$ value is computed based on normalized Lucene and LIRE similarity score for tf.idf and CEDD facet respectively. The $w_{CEDD}$ and $w_{tf.idf}$ are facet weights for this query. For each query, we perform this similarity computation in all three languages, separately for image metadata and documents. We take this value as relevancy value of each image/document for a specific query.

## 6.3 Experiment Result

We included text tf.idf and metadata tf.idf facets in this experiment. We start with top 20 similar documents and images (as activated nodes) based on these facets for each query, and traverse the graph from these 40 touch points, step by step in parallel. In each step, for node $x$ and its neighbour $y$, we compute the $\lambda(x, y)$, update the weight and continue to the next neighbour. This is performed in the form of matrix multiplications.

In Markov chain random walks, without MH algorithm, we utilize matrix multiplication to simulate the walk in the graph. The probability distribution after $t$ steps is calculated as $a^t = a^0 \cdot W^t$, where $a^0$ is the starting scores and $a^t$ is the scores after $t$ steps. However, leveraging MH, the edge weights are affected by $\lambda$ (Eq. 1). This is a potential problem for computing the updated transition matrix. The reason is that, in each iteration, the matrix $W$ is affected by $\lambda$ which is a min function - $W \cdot \lambda$ in first iteration and $W \cdot \lambda \cdot \lambda$ in the second iteration. However, Hlynka et al. [4] observed that the transition matrix $Pr$ does not change in further steps. Therefore, we need to compute only once the matrix of $Pr(x, y) = W(x, y) \cdot \lambda(x, y)$ for all nodes, and use this matrix in further multiplications. This makes the MH steps simulation feasible in implementation.

We compute the final score as $a^t = a^0 \cdot Pr^t$ after $t$ steps. This computation is needed for middle steps, since in ideal case the multiplication is performed many times until the matrix converges and in stationary distribution the nodes' probability are independent of starting scores in the graph.

We compare the results with/without using MH algorithm (Tables 1, 2). We did not get better result in our preliminary experiment with MH. The reason is dependency of a jump to the value of $RSV(y)/RSV(x)$. The implemented RSV function for images is based on metadata facet. A large number of images are not retrieved in Lucene result for Metadata facet- we retrieve in the scale of 1000 images for each query, compared to having 274,000 images. We set the minimum value of retrieved scores (0.0001), as RSV value of visited images not in the Lucene results. We have observed that this approach biases a large number of images to very low score, which we assume to be the cause of low precision. Though, further experiments in this direction are needed [1].

# 7. CONCLUSION AND DISCUSSION

We presented a graph-based model for multimodal IR leveraging MH algorithm. The graph is enriched by extracted facets of information objects. Different modalities are treated equally thanks to faceted search. We proposed a generic relevancy function based on facet similarity of objects to the query. Leveraging this model, we have a platform, potential to investigate the affect of different facets on performance, and burning in the matrix. We have the opportunity to examine query dependent traversal, as weights in the graph are affected by relevancy of source and target nodes to the query. The preliminary results with MH did not improve the result. Many steps in the graph should be taken until the matrix burns in to the stationary distribution, which is in our future work. However, this experiment brings some issues to discuss: 1) How much the final probability distribution is dependent on the chosen $\widetilde{\pi}(x)$? 2) Is MH algorithm on graph-based collections an opportunity to compare the effect of different ranking models? 3) How much expensive is this approach regarding the need of high number of transitions until the matrix burns in? 4) How do we satisfy stochastic property in multimodal graph with heterogeneous relation types? In principle, this property is beyond mathematically summing the weights to 1, but it goes back to the utility of different modalities as neighbours to the user. The difficulty is whether these neighbours are equally useful to the user?

# 8. REFERENCES

[1] A. Awan, R. A. Ferreira, S. Jagannathan, and A. Grama. Distributed uniform sampling in unstructured peer-to-peer networks. In *HICSS*, 2006.

[2] T. Berber, A. H. Vahid, O. Ozturkmenoglu, R. G. Hamed, and A. Alpkocak. Demir at imageclefwiki 2011: Evaluating different weighting schemes in information retrieval. In *CLEF*, 2011.

[3] R. A. Ferreira, M. Krishna Ramanathan, A. Awan, A. Grama, and S. Jagannathan. Search with probabilistic guarantees in unstructured peer-to-peer networks. In *P2P*, 2005.

[4] M. Hlynka and M. Cylwa. *Observations on the Metropolis-Hastings Algorithm.* University of Windsor, Department of Mathematics and Statistics, 2009.

[5] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking through random walk over document-level context graph. MULTIMEDIA, 2007.

[1] The code of Astera is open-source and available at link: http://www.ifs.tuwien.ac.at/∼sabetghadam/Astera.html

| *steps* | *st* | *p@10* | *r@10* | *p@20* | *r@20* |
|---|---|---|---|---|---|
| 1 | 0.297 | 0.135 | 0.229 | 0.158 | |
| 2 | 0.297 | 0.135 | 0.229 | 0.158 | |
| 3 | 0.252 | 0.123 | 0.188 | 0.138 | |
| 4 | 0.224 | 0.120 | 0.184 | 0.134 | |
| 5 | 0.206 | 0.1148 | 0.173 | 0.124 | |
| 6 | 0.182 | 0.1104 | 0.156 | 0.113 | |
| 7 | 0.142 | 0.106 | 0.13 | 0.115 | |

Table 1: Result for documents without image facets, self-transitivity: 0.9, links: $\delta, \beta$

| *steps* | *st* | *p@10* | *r@10* | *p@20* | *r@20* |
|---|---|---|---|---|---|
| 1 | 0.27 | 0.125 | 0.151 | 0.135 | |
| 2 | 0.27 | 0.125 | 0.151 | 0.135 | |
| 3 | 0.23 | 0.113 | 0.148 | 0.1295 | |
| 4 | 0.22 | 0.1097 | 0.133 | 0.1163 | |
| 5 | 0.18 | 0.1091 | 0.113 | 0.1163 | |
| 6 | 0.17 | 0.107 | 0.111 | 0.109 | |
| 7 | 0.14 | 0.08 | 0.108 | 0.087 | |

Table 2: Result for documents without image facets, self-transitivity: 0.9, links: $\delta, \beta$

[6] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2008.

[7] J. Martinet and S. Satoh. An information theoretic approach for automatic document annotation from intermodal analysis. In *Workshop on Multimodal Information Retrieval*, 2007.

[8] T. Mei, Y. Rui, S. Li, and Q. Tian. Multimedia search reranking: A literature survey. *ACM Computing Surveys (CSUR)*, 2014.

[9] S. Sabetghadam, M. Lupu, and A. Rauber. Astera - a generic model for multimodal information retrieval. In *Proc. of Integrating IR Technologies for Professional Search Workshop*, 2013.

[10] B. Walsh. Markov chain monte carlo and gibbs sampling. 2004.

[11] T. Yao, T. Mei, and C.-W. Ngo. Co-reranking by mutual reinforcement for image search. CIVR, 2010.

[12] M. Zhong, K. Shen, and J. Seiferas. The convergence-guaranteed random walk and its applications in peer-to-peer networks. *Computers, IEEE Transactions on*, 2008.

# Fusion of Heterogeneous Information in Graph-Based Ranking for Query-Biased Summarization

Kotaro Sakamoto
Yokohama National University
sakamoto
@forest.eis.ynu.ac.jp

Hideyuki Shibuki
Yokohama National University
shib@forest.eis.ynu.ac.jp

Tatsunori Mori
Yokohama National University
mori@forest.eis.ynu.ac.jp

Noriko Kando
National Institute of
Informatics
kando@nii.ac.jp

## ABSTRACT

We propose a graph-based ranking method for query-biased summarization in a three-layer graph model consisting of document, sentence and word-layers. The model has a representation that fuses three kinds of heterogeneous information: part-whole relationships between different linguistic units, similarity using the overlap of the Basic Elements (BEs) in the statements, and semantic similarity between words. In an experiment using the text summarization test collection of Nakano et al., our proposed method achieved the best result of the five considered methods, which were based on other graph models with an average R-Precision of 0.338.

## Keywords

graph-based ranking, multi-layer graph model, query-biased summarization

## 1. INTRODUCTION

Query-biased summarization, which is a multi-document summarization method customized to reflect the information need expressed in a query[10], has increased in importance for accessing user-preferred information. Following TextRank[5] and LexRank[1], which use graph-based ranking algorithms for sentence selection in summarization, several versions of graph-based ranking algorithms have been proposed for query-biased summarization[3, 4, 7, 11]. Graph-based ranking algorithms are advantageous because they do not only rely on the local context of a text unit, but rather they consider information recursively drawn from the entire text[5]. Hu et al.[3] proposed an extension of the Co-HITS-Ranking algorithm by naturally fusing sentence-level and document-level information in a graph model to take into account the strength of document-to-document and sentence-to-document correlation. Their graph model has document and sentence layers with links between two homogeneous nodes and links between two heterogeneous nodes. The homogeneous nodes are defined as nodes in the same layer, and the heterogeneous nodes are defined as ones in different layers. The link weight for homogeneous nodes is similarity based on the degree of word-overlap between two sentences or two documents, and the link weight for heterogeneous nodes is similarity based on the degree of word-overlap between a sentence and a document. Note that the link weights are homogeneous in nature (based on word overlap) even if the nodes are heterogeneous.

Here, we are interested in the behavior when link weights of different natures and different layers such as the word layer are introduced into the graph model in addition to the sentence and document layers used in the Hu et al. model. Kaneko et al.[4] proposed a four-layer graph model that consists of document, passage, sentence and word layers to comprehensively select adequate passages for summaries. In their model, two nodes from different layers are linked in accordance with part-whole relationships. For example, if a sentence contains a word, the corresponding sentence layer node is linked to the corresponding word-layer node. If another sentence contains the same word, the corresponding sentence-layer node is also linked to the same word-layer node. This is another representation of word overlap between sentences, which is distinct from word overlap using link weight. In this paper, we use a three-layer graph model, which consists of document, sentence, and word layers, based on part-whole relationships. Because we are not interested in passage selection, we do not use the passage layer. We use the Basic Elements (BEs), which are minimal semantic units and represent dependencies between the words in a sentence originally proposed by Hovy et al.[2], as units for calculating the meaning of a statement in the proposed three-layer model although Hovy et al. was not graph-based. Because BEs can more exactly represent the meaning of a statement than words, we use similarity based on the degree of BE overlap instead of word overlap as link weights in the sentence and document layers. Moreover, as link weights in the word layer, we use semantic similarity based on a thesaurus. We attempt to improve graph-based ranking by fusing the above three heterogeneous natures, which are part-whole relationships between different linguistic units, BE-overlap similarity between sentences or documents, and semantic similarity between words.

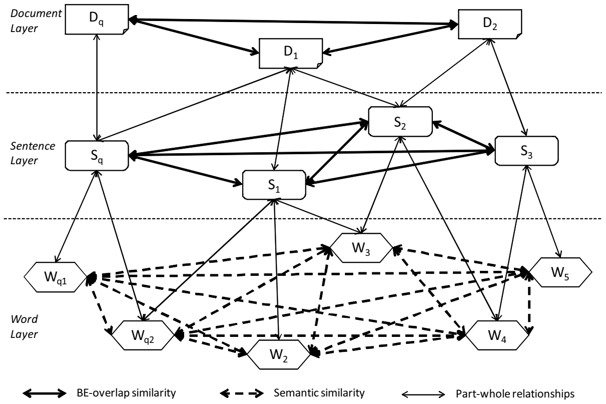


**Figure 1: The graph model for fusing heterogeneous information.**

In this paper, we propose a graph-based ranking method for query-biased summarization by extending the Co-HITS-Ranking algorithm to a three-layer graph model that has a representation fusing three kinds of heterogeneous information. Although we used Japanese texts in the experiment, the proposed graph model and algorithm are language independent. We suppose that a query is given as a sentence.

## 2. GRAPH MODEL

Figure 1 shows the graph model for fusing heterogeneous information. The model consists of three layers for representing the different linguistic units of a given document set, namely the document, sentence, and word layers.

Two nodes in the document or sentence layers are linked with each other using BE-overlap similarity. The BE-overlap similarity link is represented by a solid bold arrow in Figure 1. The BE-overlap similarity $sim_{BE}(n_i, n_j)$ between two nodes $n_i$ and $n_j$ is defined as

$$sim_{BE}(n_i, n_j) = \frac{|set_{BE}(n_i) \cap set_{BE}(n_j)|}{|set_{BE}(n_i) \cup set_{BE}(n_j)|}, \tag{1}$$

where $set_{BE}(n)$ is the set of BEs used in a linguistic unit, which is a document or sentence, corresponding to $n$. Moreover, $sim_{BE}$ $(n_i, n_j)$ is a value in the interval [0,1]. As the rate of BEs commonly used in $n_i$ and $n_j$ increases, the value of $sim_{BE}$ $(n_i, n_j)$ becomes higher.

Two nodes in the word layer are linked with each other using semantic similarity based on a thesaurus. The semantic similarity link is represented by a dashed arrow in Figure 1. The semantic similarity $sim_{sem}(n_i, n_j)$ between two word-layer nodes $n_i$ and $n_j$ is defined as

$$sim_{sem}(n_i, n_j) = \frac{MD - \max_{c \in hyper(c_k, c_l)} depth(c)}{MD}, \tag{2}$$

where $MD$ is the maximum depth of the thesaurus, $c_k$ and $c_l$ are the concepts in the thesaurus corresponding to $n_i$ and $n_j$, respectively, $hyper(c_k, c_l)$ is a set of thesaurus concepts that subsume both $c_k$ and $c_l$, and $depth(c)$ is the depth of concept $c$ in the thesaurus. Here, $sim_{sem}(n_i, n_j)$ is a value in the interval [0,1]. When the distance between nodes $n_i$ and $n_j$ decreases, the value of $sim_{sem}(n_i, n_j)$ becomes higher.

Two nodes in neighboring layers, namely between the document and sentence layers or between the sentence and word layers, are linked with each other using part-whole relationships. The part-whole relationship link is represented as a solid thin arrow in Figure 1. If a linguistic unit in the upper layer contains a unit in the lower layer, a part-whole relationship link can be drawn. For example, if a sentence in the sentence layer contains word $w$, a part-whole relationship link is drawn between the node for the sentence in the sentence layer and the node for word $w$ in the word layer. The link weight of part-whole relationships is fixed to 1. Note that a part-whole relationship link between the document and sentence layers indicates that the document contains words used in the sentence. Therefore, two nodes in the document and word layers are not directly linked.

## 3. ALGORITHM

The proposed method takes a query sentence and a set of documents as input and ranks all sentences in the documents, in order of relevance to the query, using the extended Co-HITS-Ranking algorithm. The proposed method is performed in four stages. The first stage makes a graph representing the query and documents. The second stage assigns initial ranking scores $R^q$ to all nodes in the graph. The third stage calculates homogeneous ranking scores $R^o$ according to recommendations among the neighboring homogeneous nodes. The fourth stage calculates heterogeneous ranking scores $R^e$, which are the final ranking scores, according to recommendations among the neighboring heterogeneous nodes.

### 3.1 Constructing the Graph

The graphical representation of query sentence is given as follows. The node for the query is added to the sentence layer. Another node corresponding to the query, which is regarded as a pseudo-document, is added to the document layer. Nodes of words used in the query are added to the word layer. The above-mentioned nodes are defined as query nodes in the lump. The graphical representation of the input documents is given as follows. One node per document is added to the document layer. Nodes corresponding to sentences or words used in the document are added to the sentence or word layers, respectively. Finally, two nodes in neighboring layers are linked based on part-whole relationships, two nodes in the document or sentence layers are linked using BE-overlap similarity, and two nodes in the word layer are linked using semantic similarity.

### 3.2 Assigning Initial Scores

The initial ranking score $R^q(n)$ of node $n$ is defined as

$$R^q(n) = \begin{cases} 1 & \text{(if } n \text{ is a query node)} \\ 0 & \text{(otherwise)} \end{cases} . \tag{3}$$

This is a simple criterion that $R^q(n)$ is 1 if $n$ is a query node; otherwise, $R^q(n)$ is 0.

### 3.3 Ranking Homogeneous Nodes

The ranking of homogeneous nodes in a layer is performed separately from ranking in other layers. When we define a link weight $sim^o(n_i, n_j)$ between homogeneous nodes $n_i$ and

$n_j$ as

$$sim^o(n_i, n_j) = \begin{cases} sim_{sem}(n_i, n_j) & \text{(if they are word-layer nodes)} \\ sim_{BE}(n_i, n_j) & \text{(otherwise)} \end{cases}$$

the homogeneous ranking score $R^o(n_i)$ of $n_i$ is repeatedly calculated until the value converges according to the following expression:

$$\begin{aligned} R^o(n_i) &= d^o \sum_{n_j \in In(n_i)} \frac{sim^o(n_i, n_j)}{\sum_{n_k \in Out(n_j)} sim^o(n_j, n_k)} R^o(n_j) \\ &\quad + (1 - d^o) R^q(n_i), \end{aligned} \tag{5}$$

where $In(n_i)$ is a set of nodes linked to $n_i$, $Out(n_j)$ is a set of nodes linked from $n_j$, and $d^o$ is a trade-off parameter in the interval [0,1]. As the value of $d^o$ increases, more importance is given to ranking scores from the neighborhood homogeneous nodes compared to the initial score.

### 3.4 Ranking Heterogeneous Nodes

The ranking of heterogeneous nodes in neighboring layers is performed as follows. When a link weight $sim_{PW}(n_i, n_j)$ between heterogeneous nodes $n_i$ and $n_j$ is defined as the same value as the link weight of part-whole relationships, the heterogeneous ranking score $R^e(n_i)$ of $n_i$ is repeatedly calculated until the value converges according to the following expression:

$$\begin{aligned} R^e(n_i) &= d^e \sum_{n_j \in In(n_i)} \frac{sim_{PW}(n_i, n_j)}{\sum_{n_k \in Out(n_j)} sim_{PW}(n_j, n_k)} R^e(n_j) \\ &\quad + (1 - d^e) R^o(n_i), \end{aligned} \tag{6}$$

where $d^e$ is a trade-off parameter in the interval [0,1]. As the value of $d^e$ increases, more importance is given to ranking scores from the neighborhood heterogeneous nodes compared to the initial score. Finally, all sentences are ranked and returned in the order of the $R^e$ values of the sentence-layer nodes, with the exception of the query node.

## 4. EXPERIMENT

### 4.1 Experimental Setup

To research effects of fusing this heterogeneous information, we perform experimental comparisons using the following four graph models. The first model has only sentence layer like TextRank or LexRank and is referred to as "Only S-layer." The second model has sentence and document layers similar to the original Co-HITS-Ranking and is referred to as "With D-layer." The third model has sentence and word layers and is referred to as "With W-layer." The forth model is the proposed model that has document, sentence and word layers and is referred to as "Three layers." Note that links of part-whole relationships are not included in the first model and that links of semantic similarity are not included in the first and second models.

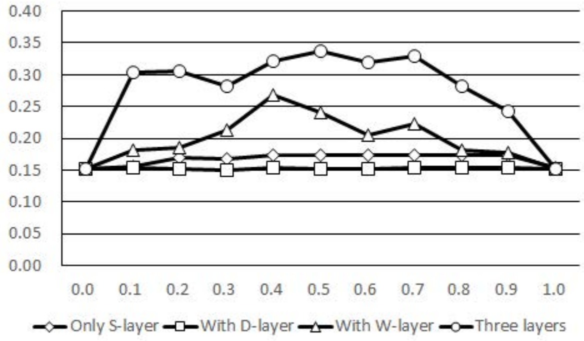


**Figure 2: Changes of the average R-Precision values .**

**Table 1: The average R-Precision values in the condition of $d^o = d^e = 0.5$.**

| | |
|---|---|
| Only S-layer | 0.173 |
| With D-layer | 0.151 |
| With W-layer | 0.240 |
| Three layers | **0.338** |

For the experimental data, we use the text summarization test collection[6] annotating sentence importance as summary materials for the credibility of information on the Web. The test collection has six query sentences, six sets of Web source documents, 24 extractive summaries, and 24 free descriptive summaries. The Web source documents are retrieved via the search engine TSUBAKI[9] using query sentences. Note that the documents are already biased to a query sentence, in that they include many common words, which will influence the word- or BE-overlap similarity, such as the words used in the query sentence. All sentences in the Web documents are annotated by four human annotators with binary labels regardless of whether the sentence seems to be useful for generating the extractive summary. Note that the annotators exhaustively applied the "useful" label to sentences even if the sentences were not used as part of the extractive summary. Therefore, we evaluate ranking methods using the "useful" label. If a method can rank more "useful" sentences above "useless" sentences, the method is considered more effective than other methods.

For the evaluation measure, we use the average R-Precision[1] $ARP$, which is the mean of the R-Precision values over a set of $Q$ queries. The R-Precision $RP(q)$ is the precision at the R-th position in the results ranking for query $q$ that has R "useful" sentences in the Web document set. The values $ARP$ and $RP$ are calculated as follows:

$$ARP = \frac{1}{Q} \sum_{q \in Q} RP(q), \tag{7}$$

$$RP(q) = \frac{r}{R}, \tag{8}$$

where $r$ is the number of sentences among the top R sentences that contains at least one "useful" label.

### 4.2 Result and Discussion

Figure 2 shows the changes in the average R-Precision values when the trade-off parameters $d^o$ and $d^e$ change by 0.1 between 0.0 and 1.0. Table 1 shows the average R-Precision values of the four methods at the condition that

[1] http://trec.nist.gov/pubs/trec15/appendices /CE.MEASURES06.pdf

$d^o = d^e = 0.5$. The proposed method achieved the best result. The results are improved as the number of layers in the models except for "With D-layer" increases. Therefore, we believe that fusing heterogeneous information improves the graph-based ranking algorithm and that the proposed model is effective.

Here, we describe why the result of "With D-layer" is worse than the result of "Only S-layer." The first reason is that the retrieved Web source documents are already biased to a query sentence. The second reason is that the same nature of links are used in both document and sentence layers. Therefore, the information in the document layer is very similar to the information in the sentence layer. Because the fusion of similar information cannot provide comprehensive judgment, if there is wrong information in a layer, it cannot be easily corrected by information in another layer. In the case of "With D-layer," we believe that the information of the sentence layer is deteriorated by its similar nature of the document layer. On the other hand, the proposed method was improved by fusing the word-layer information more heterogeneously than the document-layer information.

## 5. CONCLUSION

In this paper, we proposed a graph-based ranking method for query-biased summarization in a three-layer graph model that consists of document, sentence, and word layers. The model fuses part-whole relationships between different linguistic units, BE-overlap similarity between statements, and semantic similarity between words. In the experiment, the proposed method achieved the best average R-Precision of 0.338. We confirmed that fusing heterogeneous information improved the graph-based ranking algorithm when Web documents retrieved by a query sentence were given as source documents.

In our future work, we will investigate the optimal expressions for calculating the link weights and other kinds of links and layers. Moreover, we will apply this method to answer questions involving various context information. For example, at the NTCIR-11 QA-Lab task[8], a challenge to make QA systems answer questions of "world history" in real-world university entrance exams was conducted. Because such QA requires comprehensive judgment that considers various context information, we believe that the proposed method is well suited for the task.

## 6. REFERENCES

[1] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479, Dec. 2004.

[2] E. Hovy, C. yew Lin, and L. Zhou. A be-based multidocument summarizer with query interpretation. In *Proc. of DUC 2005*, 2005.

[3] P. Hu, D. Ji, and C. Teng. Co-hits-ranking based query-focused multi-document summarization. In *Information Retrieval Technology - 6th Asia Information Retrieval Societies Conference, AIRS 2010, Taipei, Taiwan, December 1-3, 2010. Proceedings*, pages 121–130, 2010.

[4] K. Kaneko, H. Shibuki, M. Nakano, R. Miyazaki, M. Ishioroshi, and T. Mori. Mediatory summary generation: Summary-passage extraction for information credibility on the web. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, PACLIC 23, Hong Kong, China, December 3-5, 2009*, pages 240–249, 2009.

[5] R. Mihalcea and P. Tarau. Textrank: Bringing order into texts. In D. Lin and D. Wu, editors, *Proceedings of EMNLP 2004*, pages 404–411, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[6] M. Nakano, H. Shibuki, R. Miyazaki, M. Ishioroshi, K. Kaneko, and T. Mori. Construction of text summarization corpus for the credibility of information on the web. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[7] J. Otterbacher, G. Erkan, and D. R. Radev. Using random walks for question-focused sentence retrieval. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 915–922, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[8] H. Shibuki, K. Sakamoto, Y. Kano, T. Mitamura, M. Ishioroshi, K. Y. Itakura, D. Wang, T. Mori, and N. Kando. Overview of the ntcir-11 qa-lab task. In *Proceedings of the 11th NTCIR Conference*, 2014.

[9] K. Shinzato, T. Shibata, D. Kawahara, and S. Kurohashi. Tsubaki: An open search engine infrastructure for developing information access methodology. *Journal of Information Processing*, 20(1):216–227, 2012.

[10] A. Tombros and M. Sanderson. Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 2–10, New York, NY, USA, 1998. ACM.

[11] X. Wan, J. Yang, and J. Xiao. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2903–2908, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

# Unilateral Jaccard Similarity Coefficient

Julio Santisteban
Universidad Católica San Pablo
Campus Campiña Paisajista s/n Quinta Vivanco,
Barrio de San Lázaro
Arequipa, Peru
jsantisteban@ucsp.edu.pe

Javier L. Tejada Carcamo
Universidad Católica San Pablo
Campus Campiña Paisajista s/n Quinta Vivanco,
Barrio de San Lázaro
Arequipa, Peru
jtejadac@ucsp.edu.pe

## ABSTRACT

Similarity measures are essential to solve many pattern recognition problems such as classification, clustering, and retrieval problems. Various similarity measures are categorized in both syntactic and semantic relationships. In this paper we present a novel similarity, Unilateral Jaccard Similarity Coefficient (uJaccard), which doesn't only take into consideration the space among two points but also the semantics among them.

## Categories and Subject Descriptors

E.1 [**Data Structures**]: Graphs and networks; G.2.2 [**Graph Theory**]: Graph algorithms

## General Terms

Theory

## Keywords

Jaccard, distance, similarity

## 1. INTRODUCTION

Since Euclid to today many similarity measures have been developed to consider many scenarios in different areas, particularly in the last century. Similarity measures are used to compare different kind of data which is fundamentally important for pattern classification, clustering, and information retrieval problems [3]. Similarity relations have generally been dominated by geometric models in which objects are represented by points in a Euclidean space [12]. Similarity is defined as "Having the same or nearly the same characteristics" [4], while the metric distance is defined as "The property created by the space between two objects or points". All metric distance functions must satisfy three basic axioms: minimality and equal self-similarity, symmetry, and triangle inequality.

$$d(i,i) = d(j,j) \leq d(i,j) \tag{1}$$

$$d(i,j) = d(j,i) \tag{2}$$

$$d(i,j) + d(j,k) \geq d(i,k) \tag{3}$$

Here for objects i, j and k, where d() is the distance between objects i and j. Bridge [1] argues that there exists empirical evidence of violations against each of the three axioms. Yet, there also exists geometric models of similarity which take asymmetry into account [10]. Nosofsky points out that a number of well-known models for asymmetric proximity data are closely related to the additive similarity and bias model [5]. Tversky [13] has proposed a different model in order to overcome the metric assumption of geometric models. One of the strengths of contrast models is its capability to explain asymmetric similarity judgments. Tversky's asymmetry may often be characterized in terms of stimulus bias and determined by the relative prominence of the stimuli.

$$sim(a,b) = \frac{|A \cap B|}{|A \cap B| + \alpha|A - B| + \beta|B - A|}, \quad \alpha, \beta \geq 0 \tag{4}$$

Here A and B represent feature sets for the objects a and b respectively; the term in the numerator is a function of the set of shared features, a measure of similarity, and the last two terms in the denominator measure dissimilarity: $\alpha$ and $\beta$ are real-number weights; when $\alpha$ != $\beta$. Jimenez et al. [6], Weeds and Weir [14] and Lee [7] also propose an asymmetric similarity measure based on Tversky's work. However all proposals include a stimulus bias, asymmetric similarity judgments, which Tversky refers to as human judgment. Today, similarity measure is deeply embedded into many of the algorithms used for graph classification, clustering and other tasks. Those techniques are leaving aside the semantic of each vertex and it's relation among other vertices and edges.

In a direct graph, the similarity from U to Z is not the same as the distance from Z to U, this due to the intrinsic features of a direct graph. The similarities are different because the channels are dissimilar. According to Shannon's information theory we could argue that each vertex is a source of energy with an average entropy which is shared among it's channels, and while that information flow among the vertex's channels, we need to be consider it in the similarity. A similarity does not fit all tasks or cases.

In Natural Langue Processing, where the similarity between two words is not symmetric sim(word a,word b) != sim(word b,word a). WordNet [4] presents 28 different types of relations; those relations have direction but are not symmetrical, they are not even synonyms because each synonym word has

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

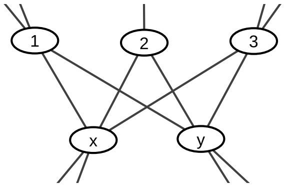


**Figure 1: Structural Equivalence.**

a particular semantic, meaning and usage, but are similar. Hence if two words have symmetric distance or similarity, those two words are the same. Paradigmatic is an intrinsic feature in language, It lets the utterer exchange words with other words, words with similar semantics [11]. In this paper we focus on paradigmatic analysis to support our unilateral Jaccard Similarity coefficient (uJaccard).
The rest of the paper is organized as follows. In section 2 we will show the unilateral Jaccard Similarity coefficient (uJaccard). In section 3 we will consider some cases; finally in section 4 we conclude this work.

## 2. PARADIGMATIC SIMILARITY DEFINITION

### 2.1 Basics Of Paradigmatic Structures

Paradigmatic analysis is a process that identifies entities which are not related directly but are related by their properties, relatedness among other entities and interchangeability [2]. In language the reason why we tend to use morphologically unrelated forms in comparative oppositions is to emphasize the semantics, this is done by substitution and transposition of words with a similar signifier. Similarity is not defined by a syntactic set of rules but rather by the use of the language. In some cases this use is not grammatically or syntactically correct but it is commonly used. We defined the signifier as being the degree of relation among entities of the same group, where not all members of the group have the same degree of relatedness. This is due to the fact that a member of a group might belong to more than one group.

### 2.2 Extended Paradigmatic

Two vertices in a graph are structurally equivalent if they share many of the same network neighbours. Figure 1 depicts a structural equivalence between two vertices y and x who have the same neighbours. Regular equivalence is more subtle, two regularly equivalent vertices do not necessarily share the same neighbours, but they do have neighbours who are themselves similar [8] [15]. We will use structural equivalence as the bases of uJaccard.

#### *2.2.1 Unilateral Jaccard Similarity*

To calculate a paradigmatic similarity we start with a question, is the similarity coefficient from vertex Va to Vc the same to the similarity coefficient from vertex Vc to Va ?. If we argue that both similarity coefficients are the same, we are arguing that the edges from the vertices Va and Vc are the same, and it is clear that that is not usually the case. Thus both vertices have different sets of edges. One problem with Tversky [13] similarity is the estimation for $\alpha$ and $\beta$ which are stimulus bias, generally a human factor. Similarly, other similarities which are based on Tversky idea, have the same problem. On the other hand we propose a measure that does not include this bias. We propose a modified version of Jaccard Similarity coefficient (1), unilateral Jaccard Similarity coefficient (uJaccard) (2)(3), used to identify the similarity coefficient of Va to Vc With respect to vertex Va, and to also identify the similarity coefficient of Vc to Va With respect to vertex Vc.

$$Jaccard(V_a, V_c) = \frac{|a \cap c|}{|a \cup c|} \tag{5}$$

$$uJaccard(V_a, V_c) = \frac{|a \cap c|}{|edges(a)|} \tag{6}$$

$$uJaccard(V_c, V_a) = \frac{|c \cap a|}{|edges(c)|} \tag{7}$$

Here Va and Vc are the number of edges in vertex a and c, likewise the edges(Vc) are the number of edges in vertex c. if uJaccard is close to 0, it means that they are not similar at all. The objective of using uJaccard is to identify how similar a vertex is to other vertices in relation to itself. uJaccard could be calculated among two connected vertices, uJaccard could also be calculated among vertices that are not connected directly, but which are connected by in-between vertices. The number of in-between vertices could be from 1 to n, we do not recommend a deep comparison since the semantics of the vertex loosest its meaning. Hence max(n)=3, it is suggested for NLP. For the calculations we do not consider the number of in-between vertices since we focus on the information flow and not the information transformation carried out on the intermediate vertices.

## 3. EXPERIMENTAL EVALUATION

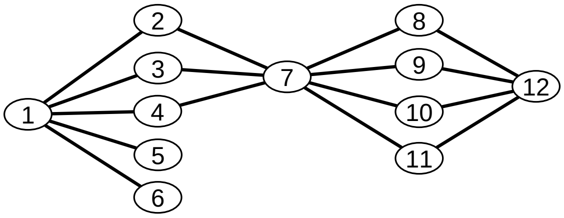


**Figure 2: Toy graph.**

### 3.1 Toy Testing

Using similarity uJaccard (6),(7) we can build a paradigmatic approach to group vertices. Figure 2 shows a toy graph with 12 vertices and 16 edges, following the paradigmatic analysis, we can determine that vertex 12 and 7 belong to group P because they have the same number of edges to a same set of vertices. Vertex 1 also belongs to group P because vertex 1 has 3 of the 5 edges, the same as vertex 7, the degree of membership of vertex 1 is lower than vertices 7 and 12 because vertex 1 has other edges that are not shared by vertices 7 or 12. In the same manner we can determine that vertex 8, 9, 10 and 11 belong to group Q because they have an equal number of edges to the same set of vertices. Similarly vertices 2, 3 and 4 belong to group R, and vertices 5 and 6 belong to group O. In this example we can easily identify the paradigmatic approach, where two or more vertices belong to the same group if they have the same or similar neighbours, but the neighbours in turn belong to another group.

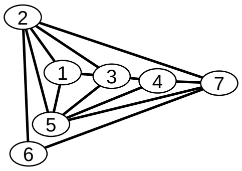
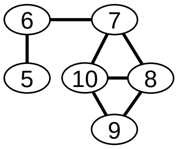
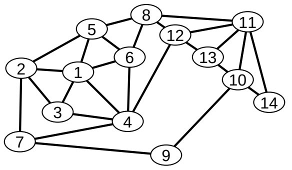
Following the uJaccard similarity and the paradigmatic approach, the results of the graph in figure 3 are shown in table 3.1. we notice that uJaccard similarity provides better information of similarity than Jaccard, this is because uJaccard considers the notion of unilateral similarity. Table 3.1 shows three toy graphs, in which we present a comparison between Jaccard and uJaccard. As show in table 3.1 uJaccard provides a unilateral similarity improving the symmetric similarity Jaccard.

**Table 1: uJaccard calculation from figure 3**

| |
|---|
| uJaccard (V1,V7) = 3/5 = 0.600 |
| uJaccard (V7,V1) = 3/7 = 0.428 |
| uJaccard (V7,V12) = 4/7 = 0.571 |
| uJaccard (V12,V7) = 4/4 = 1.000 |
| Jaccard (V1,V7) = 3/9 = 0.333 |
| Jaccard (V7,V1) = 3/9 = 0.333 |
| Jaccard (V7,V12) = 4/7 = 0.571 |
| Jaccard (V12,V7) = 4/7 = 0.571 |

Table 3.1 shows three toy graphs, in which we present a comparison among Jaccard and uJaccard. As show in table 3.1 uJaccard provide an unilateral similarity improving the symmetric similarity Jaccard.

**Table 2: Test uJaccard in toy graphs**

| | |
|---|---|
|  | uJaccard<br>sim(2,5)=4/4<br>sim(5,2)=4/6 |
| | Jaccard<br>sim(2,5)=4/6<br>sim(5,2)=4/6 |
|  | uJaccard<br>sim(7,5)=1/3<br>sim(5,7)=1/1 |
| | Jaccard<br>sim(7,5)=1/3<br>sim(5,7)=1/3 |
|  | uJaccard<br>sim(2,4)=3/4<br>sim(4,2)=3/5 |
| | Jaccard<br>sim(2,4)=3/6<br>sim(4,2)=3/6 |

## 3.2 Cut a graph

In graph theory, a cut is a partition of the vertices of a graph into two disjoint subsets. There are many techniques and algorithms to cut a graph, but in some cases there are graphs that are difficult to cut, due to their symmetric distribution of vertices.

It is shown in figure 3.2 that node 1 might belong to cluster {2,3,4} or cluster {5,6}; to resolve this problem we use uJaccard similarity measure to find the similarity of node 1 to other nodes. Table 3 shows that similarities from node 1 to other nodes 1 level deep are the same, so we could not allocate node 1 to a particular cluster. Table 3 also shows that similarities from node 1 to other nodes 2 levels deep, in which uJaccard(1,3) has a strong similarity over the rest. We could conclude that node 1 belong to cluster {2,3,4}.

In figure 3.2 also node 1 might belong to cluster {2,3,4} or cluster {5,6,7} or cluster {8,9,10,11}; this is where uJaccard comes in, being able to solve this problem. Table 4 shows result of similarities from node 1 to all other nodes on the network in different levels deep. cluster {8,9,10,11} presents the highest number of strong similarities, therefor we can conclude that node 1 belongs to cluster {8,9,10,11}.

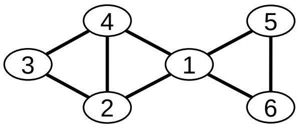


**Figure 3: Toy graph.**

**Table 3: Cut a graph 3.2 using uJaccard**

| 1 level deep | | 2 levels deep | |
|---|---|---|---|
| uJaccard(1,4) | 1/4 | uJaccard(1,2) | 1/4 |
| uJaccard(1,2) | 1/4 | uJaccard(1,3) | 2/4 |
| uJaccard(1,5) | 1/4 | uJaccard(1,4) | 1/4 |
| uJaccard(1,6) | 1/4 | uJaccard(1,5) | 1/4 |
| - | - | uJaccard(1,6) | 1/4 |

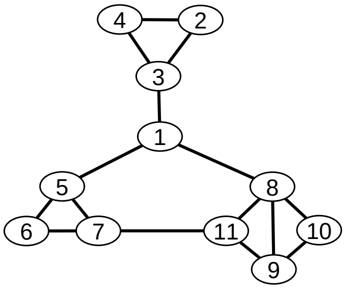


**Figure 4: Toy graph.**

## 3.3 Social Network

We tested uJaccard against two social network graphs; the first is the coauthorship network of scientists [9] the second is the network of Hollywood's actors[1].

The first network is the coauthorship network of scientists working on network theory and experiments, compiled by M. Newman [9]. We want to find the top scientists that Newman is similar to or that have paradigmatic similarity. As shown in table 5 the 3 most of Newman's paradigmatic similar scientists are Callaway, Strogatz and Holme. On the

[1] The Internet Movie Database: ftp://ftp.fu-berlin.de/pub/misc/movies/database/

**Table 4: Cut a graph 3.2 using uJaccard**

| 2 levels deep | | 3 levels deep | |
|---|---|---|---|
| uJaccard(1,4) | 1/3 | uJaccard(1,4) | 1/3 |
| uJaccard(1,2) | 1/3 | uJaccard(1,2) | 1/3 |
| uJaccard(1,6) | 1/3 | uJaccard(1,6) | 1/3 |
| uJaccard(1,7) | 1/3 | uJaccard(1,7) | 2/3 |
| uJaccard(1,9) | 1/3 | uJaccard(1,9) | 2/3 |
| uJaccard(1,11) | 1/3 | uJaccard(1,11) | 2/3 |
| uJaccard(1,10) | 1/3 | uJaccard(1,10) | 1/3 |

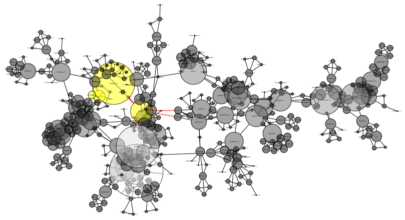


**Figure 5: Coauthorship network of scientists, selected nodes belong to scientists Newman, Callaway, Strogatz, Holme.**

other hand the top 3 scientists that are similar to Newman are Adler, Aberg and Aharony. uJaccard has been calculated in 2,3 and 4 levels deep away from Newman. Newman is more similar to Strogatz but the most similar scientist to new Newman is Adler and not Strogatz, even that Strogatz most similarity is toward Newman.

For the second network, we created the second social network of Hollywood's actors, we based on The Internet Movie Database (note). We download actors and actresses data, which includes title of movies in which they worked, we also download a list of top 1000 (nota) and top 250 (nota) actors and actresses. The network is composed of nodes representing actors and actresses, and vertices are the movies in which those actors worked together. A node is created for every person, with their names as the key, when two people are in the same movie; a vertex is created between their nodes. The first network presents 1000 top actors and actresses who also work in 41,719 movies with a total 113,478 edges. The second network presents 250 actors and actresses who work in 15,831 movies with a total of 14,096 edges. For this test we remove duplicated edges.

- From a given *actor A*
- We search for actors that *actor A* is similar to
- From the *actor A*'s similar actor list we get the most similar *actor B*
- We search for actors that *actor B* is similar to
- This is done to analyse if *actor A* and *actor B* are reciprocally similar
- Then we look for actors that are most similar to *actor A*
- We do this on the network top 250 actors and top 1000 actors.

**Table 5: uJaccard calculation from 3.3, in search paradigmatic scientists to Newman**

| 2 levels deep | | 3 levels deep | | 4 levels deep | |
|---|---|---|---|---|---|
| Scientist Newman is similar to: | | | | | |
| Callaway | 0.15 | Strogatz | 1.63 | Strogatz | 8.25 |
| Strogatz | 0.15 | Holme | 1.59 | Callaway | 7.85 |
| Watts | 0.15 | Kleinberg | 1.59 | Watts | 7.81 |
| Hopcroft | 0.11 | Sole | 1.59 | Kleinberg | 7.18 |
| Scientists that are similar to Newman: | | | | | |
| Adler | 0.33 | Aberg | 0.50 | Aberg | 2.50 |
| Aharony | 0.33 | Adler | 0.66 | Adler | 14.0 |
| Aleksiejuk | 0.50 | Aharony | 0.66 | Aharony | 14.0 |
| Ancelmeyers | 0.66 | Alava | 0.50 | Alava | 1.00 |
| Araujo | 0.33 | Albert | 0.10 | Albert | 0.50 |

The results of the search on the network of top 250 actors and top 1000 actors, using uJaccard and the paradigmatic approach are presented in tables 6 and 7. In table 6 we focus in *Tom Cruise*, we found that *Tom Cruise* is most similar to *Julia Roberts* but *Julia Roberts* is most similar to *John Travolta*, *Tom Cruise* is third in *Julia Roberts*' similarity list. clearly there is not a symmetric similarity among *Julia Roberts* and *Tom Cruise*. Moreover *Julia Roberts* is not the most similar toward *Tom Cruise*, the most similar towards *Tom Cruise* is *Heath Ledger*. Hence this confirm that uJaccard helps to identify similarities, particularly asymmetric similarities. Table 6 also shows similar scenario among *Tom Cruise*, *Tom Hanks* and *Joan Allen* in the network of top 250 actors and actresses, this confirm the usability of uJaccard.

In Table 7 we use the network of top 250 actors and actresses and we focus on *Anthony Quinn* and *Jack Nicholson*. We start by searching for actors that *Anthony Quinn* is sim-

**Table 6: uJaccard similarity among top 1000 and top 250 actor and actresses, searching paradigmatic similar actor**

| Top 1000 actors, cruise tom is similar to: | | | |
|---|---|---|---|
| | | roberts julia | |
| roberts julia | 0.405 | travolta john | 0.418 |
| hanks tom | 0.401 | hanks tom | 0.412 |
| jackson samuel | 0.399 | jackson samuel | 0.407 |
| douglas michael | 0.397 | cruise tom | 0.399 |
| eastwood clint | 0.393 | spacey kevin | 0.399 |
| Top 250 actors, cruise tom is similar to: | | | |
| | | hanks tom | |
| hanks tom | 0.430 | douglas michael | 0.425 |
| douglas michael | 0.420 | cruise tom | 0.421 |
| eastwood clint | 0.420 | jackson samuel | 0.418 |
| spacey kevin | 0.413 | travolta john | 0.414 |
| jackson samuel | 0.410 | spacey kevin | 0.411 |
| Who is similar to cruise tom: | | | |
| top 1000 actors | | top 250 actors | |
| ledger heath | 0.490 | allen joan | 0.496 |
| bacon kevin | 0.488 | balk fairuza | 0.495 |
| crowe russell | 0.482 | bello maria | 0.488 |
| gibson mel | 0.482 | collins pauline | 0.487 |
| benigni roberto | 0.482 | aiello danny | 0.487 |

ilar to, then we search for actors that are most similar to *Anthony Quinn* in 1 and 2 levels deep. We notice that *Anthony Quinn* is most similar to *Tom Hanks* but most similar actor to *Anthony Quinn* is *Antonio Banderas*, while *Anthony Quinn* is the 153th most similar for *Tom Hanks*. *Antonio Banderas* is most similar to *Samuel Jackson* and not to *Anthony Quinn*, while *Anthony Quinn* is the 53th most similar for *Antonio Banderas*. Therefore we could conclude that *Anthony Quinn* and *Tom Hanks* are not symmetric similar rather they are asymmetric similar. Table 7 also shows similar scenario for *Jack Nicholson*.

**Table 7: uJaccard similarity among top 250 actor and actresses, searching paradigmatic actor, in 2 and 3 levels deep**

| Top 250 actors, are similar to: | | | |
|---|---|---|---|
| quinn anthony | | nicholson jack | |
| hanks tom | 0.451 | hanks tom | 0.436 |
| jackson samuel | 0.443 | eastwood clint | 0.429 |
| lemmon jack | 0.443 | travolta john | 0.417 |
| cruise tom | 0.435 | williams robin | 0.417 |
| de niro robert | 0.435 | douglas michael | 0.414 |
| Who are similar to quinn anthony: | | | |
| 1 level deep | | 2 levels deep | |
| banderas antonio | 0.366 | banderas antonio | 21.866 |
| bardem javier | 0.350 | benigni roberto | 20.758 |
| martin steve | 0.333 | burns george | 20.565 |
| goodman john | 0.320 | baldwin alec | 20.413 |
| allen woody | 0.285 | mcqueen steve | 20.244 |
| Who are similar to nicholson jack: | | | |
| 1 level deep | | 2 levels deep | |
| greene graham | 0.484 | banderas antonio | 41.477 |
| bronson charles | 0.482 | bacon kevin | 41.133 |
| brody adrien | 0.480 | baldwin alec | 41.0862 |
| bale christian | 0.476 | bronson charles | 40.982 |
| baldwin alec | 0.474 | benigni roberto | 40.948 |

# 4. CONCLUSION

A key assumption of most models of similarity is that a similarity relation is symmetric. The symmetry assumption is not universal, and it is not essential to all applications of similarity. The need for asymmetric similarity is important and central in Information Retrieval and Graph Data Networks. It can improve current methods and provide an alternative point of view.

We present a novel asymmetric similarity, Unilateral Jaccard Similarity (uJaccard), where the similarity among A and B is not same to the similarity among B and C, *uJaccard(A,B) != uJaccard(B,A)*; this is based on the idea of paradigmatic association. In comparison to Tversky [13] our approach uJaccard does not need a stimulus bias, whereas in the case of Tversky human judgement is needed.

We present a series of cases in which we confirmed its usefulness and we validated uJaccard. We could extend uJaccard to include weights to improve the asymmetry, we could also use uJaccard and the paradigmatic approach to cluster Graph data Networks. These are tasks in which we are working on.

In conclusion, the proposed uJaccard similarity proved to be useful despite its simplicity and the few resources used.

# 5. ACKNOWLEDGMENTS

This research was funded in part by ”Fondo para la Innovación, Ciencia y Tecnología - Peru” (FINCyT). We thank the Universidad Católica San Pablo for supporting this research.

# 6. REFERENCES

[1] D. Bridge. Defining and combining symmetric and asymmetric similarity measures. In B. Smyth and P. Cunningham, editors, *Advances in Case-Based Reasoning (Procs. of the 4th European Workshop on Case-Based Reasoning)*, LNAI 1488, pages 52–63. Springer, 1998.

[2] F. De Saussure and W. Baskin. *Course in general linguistics*. Columbia University Press, 2011.

[3] R. Duda, P. Hart, and D. Stork. Pattern classification 2nd ed., 2001.

[4] C. Fellbaum. Wordnet and wordnets. In A. Barber, editor, *Encyclopedia of Language and Linguistics*, pages 2–665. Elsevier, 2005.

[5] E. W. Holman. Monotonic models for asymmetric proximities. *Journal of Mathematical Psychology*, 20(1):1–15, 1979.

[6] S. Jimenez, C. Becerra, and A. Gelbukh. Soft cardinality: A parameterized similarity function for text comparison. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 449–453. Association for Computational Linguistics, 2012.

[7] L. Lee, F. C. Pereira, C. Cardie, and R. Mooney. Similarity-based models of word cooccurrence probabilities. In *Machine Learning*. Citeseer, 1999.

[8] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of mathematical sociology*, 1(1):49–80, 1971.

[9] M. E. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, 74(3):036104, 2006.

[10] R. M. Nosofsky. Stimulus bias, asymmetric similarity, and classification. *Cognitive Psychology*, 23(1):94–140, 1991.

[11] M. Sahlgren. The word-space model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces. 2006.

[12] R. N. Shepard. Representation of structure in similarity data: Problems and prospects. *Psychometrika*, 39(4):373–421, 1974.

[13] A. Tversky. Features of Similarity. In *Psychological Review*, volume 84, pages 327–352, 1977.

[14] J. Weeds and D. Weir. Co-occurrence retrieval: A flexible framework for lexical distributional similarity. *Computational Linguistics*, 31(4):439–475, 2005.

[15] D. R. White and K. P. Reitz. Graph and semigroup homomorphisms on networks of relations. *Social Networks*, 5(2):193–234, 1983.

# Information Retrieval Boosted by Category for Troubleshooting Search System

Bin Tong Toshihiko Yanase Hiroaki Ozaki Makoto Iwayama
Research & Development Group, Hitachi, Ltd.
1-280, Higashi-koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
{bin.tong.hh, toshihiko.yanase.gm}@hitachi.com
{hiroaki.ozaki.yu, makoto.iwayama.nw}@hitachi.com

## ABSTRACT

Troubleshooting search system aims at extracting relevant information to solve the problem at hand. It is often the case that documents in troubleshooting system includes an abundant amount of domain-specific categories. However, the useful information about the domain-specific categories, such as relationship between words and categories and relationship between categories, is not fully utilized in simple query search and faceted search. In this paper, we propose an information retrieval method boosted by the domain-specific categories. Given a problem query and categories, the troubleshooting search system is able to retrieve the relevant information of interest with respect to the selected categories. The experiment results show our proposal improves the recall.

## Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Retrieval Models

## Keywords

Troubleshooting, Category, Co-occurrence Graph

## 1. INTRODUCTION

Troubleshooting [13] is a form of problem solving, and is often applied to repair malfunctioned facilities or equipments. Maintenance log [10, 5, 2] is one of important documents for troubleshooting, which is generated during conversations between customers and engineers in equipment maintenance. The maintenance log often includes the entries for problem titles and documents that relate to the problem description in details and instructions to solving the problem. To ease the management of the huge amount of the maintenance logs, domain-specific categories, such as machine code, trouble code, and countermeasure code, are used to tag for both the problem titles and the documents.

The target of information retrieval for troubleshooting [11] based on the maintenance logs is to help engineers to examine the similar situations of a problem within a certain period of time, which facilitates an appropriate solution. The troubleshooting search system requires the engineers to input a short problem query to search the relevant information from the documents. It may cause the lexical gap problem [8, 9], because it is difficult for the engineers to compose a succinct and precise problem query to represent their information needs.

Moreover, information about the domain-specific categories is not fully utilized in the problem query search. One way to use the category information is to make faceted search, in which the selected categories are used to filter the ranking results. For example, if a machine code is selected, all search results are restricted to the selected machine code.

However, the faceted search might have two problems in the troubleshooting search system. First, the information related to the selected categories can not be retrieved, since the retrieved information is limited to the selected categories. However, it is natural that system engineers tend to check relevant problems to facilitate their decision making. For example, given a selected machine code, the information about another machine code might be informative to solutions if two machine codes belong to the same machine series and have similar problems. Second, the ranking of search results is only dependent on the problem query but not on the selected categories. For example, a trouble code corresponds to a number of specific countermeasure codes. Given a selected trouble code, the information about its frequent countermeasures is expected to place higher in the ranked list of results.

To mitigate the lexical gap problem and the above mentioned retrieval problems, we propose an information retrieval method using a scoring technique. Our proposal is extended from a word co-occurrence graph in the QSB method [9] that aims at solving the lexical gap problem. In our proposal, besides using the word co-occurrence to score words in documents, the word's score is also weighted by a boosting term about the domain-specific categories. More specifically, the boosting term considers the relationship between categories and words and the relationship between categories. They are utilized to alleviate the above two retrieval problems with respect to the categories.

## 2. RELATED WORK

The information retrieval for troubleshooting is related to question answering [15] and query biased document summarization document summarization [14, 1, 3]. The most of work about question answering has been focusing on factoid

*SIGIR Workshop on Graph Search and Beyond* '15 Santiago, Chile
Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

question answering [6, 12, 7]. However, in the troubleshooting system, the answer of the question is a set of relevant sentences or phrases. As to query biased document summarization, there seems to be no work that leverages other auxiliary information, such as categories. In addition, it is worth to mention the work [12] about non-factoid question answering. In this work, Surdeanu et al. proposed a framework for answer ranking by exploiting various linguistic features generated from popular techniques, such as syntactic parsing, Name Entity Recognition (NER), and Semantic Role Labeling (SRL). However, except for regular sentences, a large number of typos and short phrases exist in maintenance logs. In such a case, those techniques might not perform well due to the irregularities in texts and the lack of training data in the troubleshooting domain.

Query Snowball (QSB) is a method for multi-document summarization that extracts the relevant sentences from multiple documents with respect to the given query. The basic idea of this method is to fill up the lexical gap between the query and relevant sentences by enriching the information need representation. In order to achieve it, a co-occurrence graph for the words in the queries and the documents is built. The words in the co-occurrence graph consist of three layers, which are $Q$ words, $R_1$ words, and $R_2$ words. $Q$ is the set of query terms. $R_1$ is the set of words that co-occur with a query term in the same sentence. $R_2$ is the set of words that co-occur with a word from $R_1$, excluding those that are already in $R_1$.

## 3. QUERY SNOWBALL WITH CATEGORY INFORMATION

To extract relevant information with respect to the selected categories, we extend the co-occurrence graph in QSB by integrating two types of relations, including the relationship between words and categories and the relationship between categories. The reason to extend QSB is that the co-occurrence graph is flexible to integrate the two relations.

The relationship between words and categories represents the distribution on words with respect to categories. If probabilities of words with respect to a given category are high, the information about these words is more likely to be retrieved given that category. The fundamental idea is that the distributions on co-occurrence probabilities of words with respect to different categories might be different. It is assumed that the words of higher probabilities with respect to a category are treated more important in that category. For example, a word appears more often in documents of a category than other categories. The word is therefore more important for that category than the others. Similarly, the relationship between categories represents the occurrences of categories. The information about categories, whose occurrence frequencies with respect to a specific category are high, is more likely to be retrieved. For example, a given category appears more often with specific categories than the others. The information about those specific categories with respect to the given category is treated more important than other categories.

### 3.1 Co-occurrence Graph Extension

In the first step, we build two co-occurrence graphs for the words and category values in the problem queries and the documents, respectively. If a category value is associated with a document, the words in the document have edges with the category value. In other words, this category value is associated with all the words in the sentences of documents. Similarly, if a category value is associated with a problem query, the words in the problem query have edges with the category value. In other words, this category value is associated with all the words in the problem query.

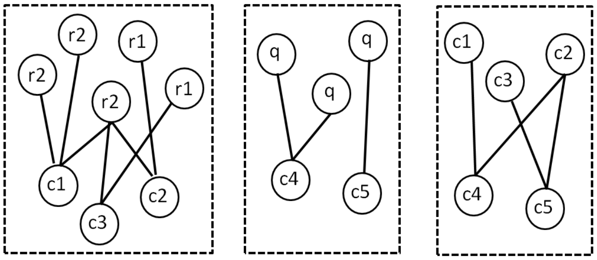


Figure 1: The co-occurrence graph among the words and the categories

In the second step, we build the co-occurrence graph for the category values in both problem queries and the documents. Suppose that a problem query corresponds to a document. In this graph, the category values associated with the problem query have edges with the category values associated with the document. As illustrated in Figure 1, the left side represents the co-occurrence graph for the $R_1$ and $R_2$ words and the category values associated with the documents; the middle part represents the co-occurrence graph for the $Q$ words and the category values associated with the queries; the right side represents the co-occurrence graph for the category values in the queries and the category values in the documents. We define $C_M$ as the set of the category values in the query set that are selected by the end-user, and $C_N$ as the set of the category values in the query set that are not selected by the end-user. Similarly, we define $C_J$ as the set of the category values in the document set that are selected by the end-user, and $C_K$ as the set of the category values in the document set that are not selected by the end-user. We also define $C_{MN} = C_M \cup C_N$ as the set of the category values for the queries and $C_{JK} = C_J \cup C_K$ as the set of the category values for the documents.

### 3.2 Score Boosted by Category Information

In order to integrate two new relationships about categories into the QSB method, we invent a score with respect to a query, $C_M$ and $C_J$, which is boosted by category information. The new score, which we call cqsb, can be formulated as follows:

$$cqsb(w) = qsb(w)\exp(\lambda \cdot s_{ctg}(w)) \quad (1)$$

where $qsb(w)$ is the score for a word $w$ in the QSB method. $s_{ctg}(w)$ is the boosting term for the word $w$, which includes two new relationships about categories. More specifically, the probabilities between words and categories and the probabilities between categories, which are calculated through the co-occurrence graph, are used to boost the score $qsb(w)$. $\lambda$ is a weight for the term $s_{ctg}(w)$. It can be seen from Eq. (1) that when $s_{ctg}(w)$ is larger than 0, $exp(\cdot)$ will be larger than 1. When multiplied with $qsb(w)$, it will give the word $w$ a higher degree of importance. If the value of $\lambda$ is set to be 0, $s_{ctg}(w)$ does not take any effect. Note that $s_{ctg}(w)$ is always larger than or equal to zero. The score of a sentence is a summation of the scores for any combinations of two words, which is simply calculated by multiplying the *cqsb* scores of the two words.

### 3.3 Score for $R_1$ Words

The relevant score of a word $r_1$ ($r_1 \in R_1$) with respect to the category values can be formulated as follows:

$$s_{ctg}(r_1) = s_{wc}(r_1, Q^{r_1}_{C_M}) + s_{cc}(r_1, Q^{r_1}_{C_M}) \qquad (2)$$

where $s_{wc}(r_1, Q^{r_1}_{C_M})$ measures the relationship between the words and the categories. $s_{cc}(r_1, Q^{r_1}_{C_M})$ measures the relationship between the categories. $Q^{r_1}_{C_M}$ is a set of top $k$ query terms that co-occur most frequently with the word $r1$. In addition, the words in $Q^{r_1}_{C_M}$ follow a constraint that they should have edges with both the word $r_1$ and the category values in $C_M$.

The term $s_{wc}(r_1, Q^{r_1}_{C_M})$ in Eq. (2) can be calculated as:

$$s_{wc}(r_1, Q^{r_1}_{C_M}) = \sum_{q \in Q^{r_1}_{C_M}} \sum_{i=1}^{|C^q_{MN}|} \theta \frac{freq(c_i, q)}{freq(c_i)} \qquad (3)$$

where $C^q_{MN}$ is the set of category values for the queries that also have edges with $q$. Let $\theta$ be $\beta$ if $c_i \in C_M$ and $c_i \in C^q_{MN}$, and be $1-\beta$ if $c_i \notin C_M$ otherwise. $freq(c_i, q)$ is the frequency of sentences that include both $c_i$ and $q$, which can be also represented by the distribution on $c_i$ with respect to $q$. It can be seen that Eq. (3) measures the closeness degree between the word $r_1$ and the category values in $C_M$ through the words in $Q^{r_1}_{C_M}$, since the word $r_1$ does not directly have edges with the category values in $C_M$ in the co-occurrence graph.

The term $s_{cc}(r_1, Q^{r_1}_{C_M})$ in Eq. (2) can be calculated as:

$$s_{cc}(r_1, Q^{r_1}_{C_M}) = \sum_{q \in Q^{r_1}_{C_M}} \sum_{\theta \in \Gamma} \sum_{c_i, c_j} \theta \frac{freq(c_i, c_j)}{freq(c_i)} \qquad (4)$$

where $\Gamma = \{\beta, 1-\beta\}$. $c_i \in C^{r_1}_J$ and $c_j \in C^q_M$ when $\theta = \beta$. Note that $C^{r_1}_J$ is a set of categories in $C_J$ that also have edges with the word $r_1$, and $C^q_M$ is a set of categories in $C_M$ that also have edges with the word $q$ ($q \in Q^{r_1}_{C_M}$). Let $c_i \in C^{-r_1}_{JK}$ and $c_j \in C^{-q}_{MN}$ when $\theta = 1-\beta$. Note that $C^{-r_1}_{JK} = C^{r_1}_{JK} - C^{r_1}_J$ and $C^{-q}_{MN} = C^q_{MN} - C^q_M$. $C^{r_1}_{JK}$ is a set of categories in $C_{JK}$ that also have edges with the word $r_1$, and $C^q_{JK}$ is a set of categories in $C_{MN}$ that also have edges with the word $q$ ($q \in Q^{r_1}_{C_M}$). It can be seen that Eq. (4) measures the closeness degree of the category values in $C^{r_1}_J$ and $C^q_M$.

### 3.4 Score for $R_2$ Words

Similarly, the relevant score of a word $r_2$ ($r_1 \in R_2$) with respect to the category values can be formulated as follows:

$$s_{ctg}(r_2) = s_{wc}(r_2, Q^{r_2}_{C_M}) + s_{cc}(r_2, Q^{r_2}_{C_M}) \qquad (5)$$

where $s_{wc}(r_2, Q^{r_2}_{C_M})$ measures the closeness degree between the word $r_1$ and the category values which have edges with $q$ words. $s_{cc}(r_2, Q^{r_2}_{C_M})$ measures the closeness degree between the category values in the query set and the category values in the document set, which are respect to the word $r_2$. $Q^{r_2}_{C_M}$ represents a set of query terms $q$ ($q \in Q$) that have close relationship with the word $r_2$. Since the word $r_2$ does not have edges with the $Q$ words in the co-occurrence graph of the word-word relation, the measurement of the relation could be done through the $R_1$ words by using the frequency, such as $freq(r_1, r_2)$ and $freq(r_1, q)$. An intuitive example of the measurement is a multiplication of $freq(r_1, r_2)$ and $freq(r_1, q)$ for the word $r_2$ and the word $q$. The word $q$ ($q \in Q^{r_2}_{C_M}$) also holds two constraints that the word $q$ is able to reach the word $r_2$ in the co-occurrence graph through a specific word $r_1$ and the word $q$ should have edges with the category values in $C_M$.

The term $s_{wc}(r_2, Q^{r_2}_{C_M})$ in Eq. (5) is calculated as:

$$s_{wc}(r_2, Q^{r_2}_{C_M}) = \sum_{r_1 \in R^1_{r_2}} \frac{freq(r_1, r_2)}{sum_{R^1_{r_2}}} s_{wc}(r_1, Q^{r_2}_{C_M}) \qquad (6)$$

where $R^1_{r_2}$ represents a set of $R_1$ words which have the top $k$ highest frequencies with the word $r_2$, and $sum_{R^1_{r_2}} = \sum_{r_1 \in R^1_{r_2}} freq(r_1, r_2)$. The term $s_{wc}(r_1, Q^{r_2}_{C_M})$ can be calculated by Eq. (3).

The term $s_{cc}(r_2, Q^{r_2}_{C_M})$ in Eq. (5) can be calculated through Eq. (4) by substituting $Q^{r_1}_{C_M}$ with $Q^{r_2}_{C_M}$.

## 4. EXPERIMENTS

In this experiment, we use the maintenance reports from a leading construction machinery company in Japan. We collect a part of the maintenance reports of 4 dominated troubles from total 19 troubles. Note that equipment code is the category for the problem queries. Phenomenon code and the countermeasure code are the categories for the documents. In each data set, one query consists of a problem query, model code, phenomenon code, and countermeasure code. We also manually label the important sentences from the documents. For each query, we search the sentences in the documents, and evaluate the performance by comparing if the sentences in the top rank are matched with the labelled sentences. Note that precision, recall and F-score are used as criteria. Among the three metrics, recall is the most important criterion. The reason is that, in troubleshooting system, system engineers prefer to examine all similar cases until they feel confident to solve the problem at hand. In this experiment, the training data and the test data are the same, since system engineers find out similar cases in the past from the troubleshooting system. Note that building and updating the co-occurrence graph can be done periodically in an unsupervised way.

For the comparisons, we select four baseline methods, which are *cqsb*, *qsb*, *lexsim*, and *lexsim*+*qsb*. We name our proposal by *lexsim*+*cqsb*. *lexsim* represents the lexical text similarity between the problem query and the sentence in the comments, which can be simply calculated by the cosine similarity between two vectors with bag-of-words features. One example of the bag-of-words features is the counts of frequent words in documents. Note that stop words are removed when counting the frequencies of words. *lexsim*+*qsb* aggregates *lexsim* and *qsb* to obtain the final ranking list, which belongs to the rank aggregation problem [4]. As a preliminary step, a simple rank aggregation method is used. Due to the different scales of *lexsim* and *qsb* scores, we simply sum up the orders of two ranking lists that use *lexsim* and *qsb*, respectively. In other words, the smaller the summation of two orders is, the higher rank the sentence gets.

Similarly, the orders of *lexsim* score and *cqsb* score is aggregated in $lexsim + cqsb$. In this experiment, we set two weights. One is for the weight between the *qsb* score or *cqsb* score and the *lexsim* score. The other is $\lambda$, which is for the category relation term in the *cqsb* score. The tuning space of the two weights is $[0, 0.01, 0.1, 1, 10]$. We use Macro Recall, Mean Average Precision (MAP), and $F_3$ score as recall, precision, and F-score, respectively.

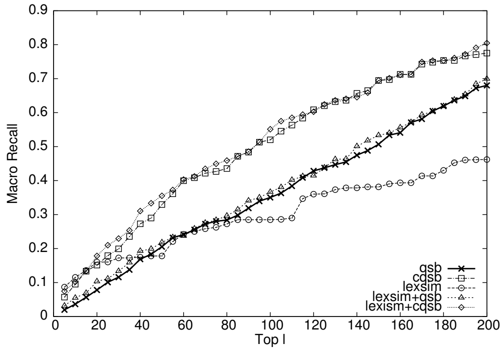


Figure 2: Recall for trouble code 03

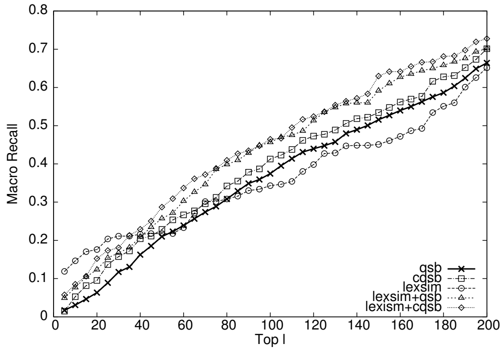


Figure 3: Recall for trouble code 05

As recall is the important criterion in troubleshooting search system, we calculate Macro Recall for each data set by setting the top $l$ ($l \in \{5, 10, \ldots, 195, 200\}$) ranking sentences. Figure 2, Figure 3, Figure 4, and Figure 5 show the Macro Recalls of all the methods, when the number of the top $l$ sentences changes from 5 to 200. It can be seen that the performances of $lexsim + cqsb$ are better than other baseline methods. It is also noticed from Figure 5 that the performance differences between *qsb* and *cqsb* are not obvious when the number of top sentences is increasing. The reason might be that the probabilities of words in informative sentences with respect to categories and the co-occurrence probabilities of categories are evenly distributed. Therefore, the second term on the right side of Eq. (1) does not differ over words to a large extent.

We also investigate MAP and $F_3$ score. For simplicity, we show their results in cases in which a best result and a worst result of Macro Recall for $lexsim + cqsb$ are achieved. The results are illustrated in Table 1 and Table 2, which are from the trouble code 03 data set and the trouble code 05 data set, respectively. Note that $\beta$ and the number of the top $l$ sentences are set to be 1 and 100, respectively. It is shown that $lexsim + cqsb$ outperforms $lexsim + qsb$, *cqsb*,

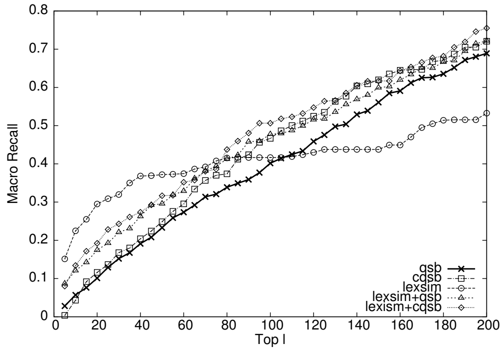


Figure 4: Recall for trouble code 14

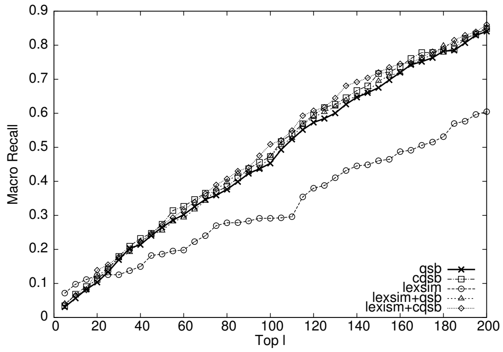


Figure 5: Recall for trouble code 18

Table 1: $F_3$ scores in trouble code 03 data set (best case)

| Methods | Macro Recall | MAP | $F_3$ score |
|---|---|---|---|
| lexsim+cqsb | .5513 | .0690 | .3244 |
| lexsim+qsb | .3665 | .0407 | .2036 |
| lexsim | .2849 | .0957 | .2379 |
| cqsb | .5200 | .0609 | .2964 |
| qsb | .3503 | .0312 | .1731 |

Table 2: $F_3$ scores in trouble code 05 data set (worst case)

| Methods | Macro Recall | MAP | $F_3$ score |
|---|---|---|---|
| lexsim+cqsb | .4645 | .0557 | .2679 |
| lexsim+qsb | .3893 | .0383 | .2031 |
| lexsim | .3431 | .1346 | .2971 |
| cqsb | .4128 | .0370 | .2049 |
| qsb | .3746 | .0299 | .1739 |

Table 3: The recalls at different values of $\beta$

| $\beta$ | $tr03_{100}$ | $tr03_{200}$ | $tr05_{100}$ | $tr05_{200}$ |
|---|---|---|---|---|
| 1 | .5200 | .7752 | .4128 | .7013 |
| 0.8 | .5225 | .7865 | .4133 | .7057 |
| 0.6 | .5560 | .8020 | .4276 | .6935 |
| 0.4 | .5456 | .7809 | .3901 | .7045 |

and *qsb* in both cases. It is also noticed that, in the worst case, even if Macro Recall of $lexsim + cqsb$ is better than the others, its $\mathrm{F}_3$ score is lower than that of *lexsim*. Note that $F_1$ has the same trend as $F_3$ in this experiment.

We also check the effect of the parameter $\beta$, as it influences the score about the relationship between the categories. Table 3 shows the macro recalls of *cqsb* at different values of $\beta$ and $l$ ($l \in \{100, 200\}$) in the data sets of trouble code 3 and trouble code 05. It is implied that 0.8 and 0.6 might be good values for *cqsb* to improve the recalls.

## 5. CONCLUSION

An information retrieval method using the scoring technique boosted by the domain-specific categories is proposed for troubleshooting search system. The knowledge about category information, which includes the relationship between words and categories the relationship between categories, is well integrated into a co-occurrence graph. The experiments on the maintenance logs proved the improvement of recalls, showing the effectiveness of using the category information.

## 6. REFERENCES

[1] A. Celikyilmaz and D. Hakkani-Tur. A Hybrid Hierarchical Model for Multi-document Summarization. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL), pages 815–824, 2010.

[2] A. Chandramouli, G. Subramanian, and D. Bal. Unsupervised Extraction of Part Names from Service Logs. In Proceedings of the World Congress on Engineering and Computer Science (WCECS), pages 826–828, 2013.

[3] H. Daumé, III and D. Marcu. Bayesian Query-focused Summarization. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (ACL), pages 305–312, 2006.

[4] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank Aggregation Methods for the Web. In Proceedings of the 10th International Conference on World Wide Web (WWW), pages 613–622, 2001.

[5] B. Edwards, M. Zatorsky, and R. Nayak. Clustering and Classification of Maintenance Logs using Text Data Mining. In Data Mining and Analytics 2008, Proceedings of the Seventh Australasian Data Mining Conference (AusDM), pages 193–199, 2008.

[6] Z. Ji, F. Xu, B. Wang, and B. He. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM), pages 2471–2474, 2012.

[7] J. Ko, L. Si, and E. Nyberg. A Probabilistic Framework for Answer Selection in Question Answering. In Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings of the Conference (NAACL-HLT), pages 524–531, 2007.

[8] J.-T. Lee, S.-B. Kim, Y.-I. Song, and H.-C. Rim. Bridging Lexical Gaps Between Queries and Questions on Large Online Q&A Collections with Compact Translation Models. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 410–418, 2008.

[9] H. Morita, T. Sakai, and M. Okumura. Query Snowball: A Co-occurrence-based Approach to Multi-document Summarization for Question Answering. In The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference (NAACL-HLT), pages 223–229, 2011.

[10] E. Mustafaraj, M. Hoof, and B. Freisleben. Mining Diagnostic Text Reports by Learning to Annotate Knowledge Roles. In Natural Language Processing and Text Mining (NLPT), pages 46–67, 2007.

[11] F. Roulland, S. Castellani, A. N. Kaplan, M. A. Grasso, and J. O'Neill. Real-time Query Suggestion in a Troubleshooting Context, Xerox, US8510306 B2, 4, 2014.

[12] M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to Rank Answers to Non-factoid Questions from Web Collections. Computational Linguistics, 37(2):351–383, June 2011.

[13] I. Sutton. Process Risk and Reliability Management: Operational Integrity Management. Elsevier, 2010.

[14] A. Tombros and M. Sanderson. Advantages of Query Biased Summaries in Information Retrieval. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pages 2–10, 1998.

[15] E. M. Voorhees. Overview of the TREC 2003 Question Answering Track. In The Text Retrieval Conference Proceedings (TREC), pages 54–68, 2003.

# Random Walk and Feedback on Scholarly Network

Yingying Yu
College of Transportation
Management
Dalian Maritime University
Dalian, China, 116026
uee870927@126.com

Zhuoren Jiang
College of Transportation
Management
Dalian Maritime University
Dalian, China, 116026
jzr1986@gmail.com

Xiaozhong Liu
School of Informatics and
Computing
Indiana University
Bloomington
Bloomington, IN, USA, 47405
liu237@indiana.edu

## ABSTRACT

The approach of random walk on heterogeneous bibliographic graph has been proven effective in the previous studies. In this study, by using various kinds of positive and negative feedbacks, we propose the novel method to enhance the performance of meta-path-based random walk for scholarly recommendation. We hypothesize that the nodes on the heterogeneous graph should play different roles in terms of different queries or various kinds implicit/explicit feedbacks. Meanwhile, we prove that the node usefulness probability has significant impact for the path importance. When positive and negative feedback information is available, we can calculate each node's proximity to the feedback nodes, and use the proximity to infer the usefulness probability of each node via the sigmoid function. By combining the transition probability and the usefulness probability of nodes on the path instance, we propose the new random walk function to compute the importance of each path instance. Experimental results with ACM full-text corpus show that the proposed method (considering the node usefulness) significantly outperforms the previous approaches.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

## Keywords

Meta-path-based Random Walk, Feedback, Heterogeneous Graph

## 1. INTRODUCTION

The volume of scientific publications has increased dramatically in the past couple of decades, which challenges existing systems and methods to retrieve and access scientific resources. Classical text-based information retrieval algorithms can recommend the candidate publications for scholars. However, most of them ignored the complex and heterogeneous relations among the scholarly objects. Not until recently, some studies proved that adopting the mining approaches on heterogeneous information networks could significantly improve the scholarly recommendation performance [3,7,9,12]. For instance, Liu et al., [2,3] constructed the heterogeneous scholarly graph and proposed a novel ranking method based on pseudo relevance feedback (PRF), which can effectively recommend candidate citation papers via different kinds of meta-paths on the graph.

In this paper, we intend to further investigate feedback information and enhance the meta-path-based random walk performance. Intuitively, for different information needs, when user feedbacks are available, the nodes on the graph should play different roles in the final measure. For example, given two different queries "*Content-based Citation Recommendation*" and "*Heterogeneous Information Network*", the same paper "*ClusCite: effective citation recommendation by information network-based clustering*" may be retrieved by scholarly search engines, e.g., Google Scholar. But the target paper can be more useful (positive) for the second query than the first one. As another example, for user X, if she prefers to cite influential scholars' work, the highly cited authors will be useful for her. While for user Y, if she tends to cite the frontiers, she will mark the newest publications and the newly topics as the useful feedback information. Therefore, the same node may perform significantly different based on different information needs and feedback information. Furthermore, by using (implicit/explicit positive/negative) feedbacks, it is possible to infer the usefulness probability of other nodes on the graph. So that, the importance of path instance will vary in terms of the probability of node usefulness.

**The main contribution of this paper is threefold**. First, in this paper, the feedback is not limited to documents. In scholarly network, user could provide feedback judgments for authors, keywords and venues, either useful or not useful. If the explicit user feedback is unavailable, we propose an approach to automatically generate the feedback nodes based on user queries and the relationships among the entities on the heterogeneous graph. Second, we infer the usefulness of the nodes in terms of feedback information. For instance, a node is less useful when it is close to the negative node(s). We make a conjecture that the usefulness probability of each node depends on its average proximity to the feedback set and can be estimated via sigmoid function. Third, we emphasize the node usefulness has a great impact on the path importance. Our approach about computing the random walk probability differs from the previous study in that, not only the transition probability, but also the usefulness probability of the node should be taken into account for random walk. To verify these hypotheses, we adopt a number of meta-paths on the graph (Figure 1) and make a comparison between the classical random walk function and the novel method. Experimental results on ACM corpus show that the proposed method significantly outperforms the original one.

The remainder of this paper is structured as follows. We 1) re-

Published on CEUR-WS: http://ceur-ws.org/Vol-1393/.

view relevant methodologies for pseudo relevance feedback, 2) introduce the preliminaries, 3) propose the improved methods, 4) describe the experiment setting and evaluation results, and 5) conclude with a discussion and outlook.

## 2. RELATED WORK

Pseudo relevance feedback, also known as blind relevance feedback, provides a way for automatic local analysis. When the user judgments or interactions are not available, it turns out to be an effective method to improve the retrieval performance. Traditional pseudo relevance feedback tends to treat the top ranked documents as relevant feedback, and then expand the initial queries. However, some of the top retrieved documents may be irrelevant, which could result in noisy feedback into the process. So that, there are various efforts to improve the traditional pseudo feedback. [11] exploited the possible utility of Wikipedia for query dependent expansion. From the perspective of each query and each set of feedback documents, [4] proposed how to dynamically predict an optimal balance coefficient query expansion rather than using a fixed value. [1] suggested to use evolutionary techniques along with semantic similarity notion for query expansion. [6] introduced an approach to expand the queries for passage retrieval, not based on the top ranked documents, but via a new term weighting function, which gives a score to terms of corpus according to their relatedness to the query, and identify the most relevant ones. Instead of using term expansion, graph-based feedback provides a new ranking assumption based on topology expansion. [2] used the pseudo relevant papers as the seed nodes, and then explored the potential relevant nodes via specific restricted/combined meta-paths on the heterogeneous graph. Our study is motivated by this approach and mainly focused on updating the random walk algorithm by investigating both the positive and negative feedbacks. In fact, positive and negative feedback approach has been studied in image retrieval [5]. With several steps of positive and negative feedback, the retrieval performance could be increasingly enhanced. From the view of negative feedback, [10] studied and compared different kinds of methods, it addressed that negative feedback is important especially when the target topic is difficult and initial results are poor. Besides, using multiple negative feedback methods could be more effective.

## 3. PRELIMINARIES

Following the work [2,8], an information network can be defined as follows.

DEFINITION 1. ***(Information network)*** *An* information network *is defined as a directed graph $G = (\mathcal{V}, \mathcal{E})$ with an object type mapping function $\tau : \mathcal{V} \to \mathcal{A}$ and a link type mapping function $\phi : \mathcal{E} \to \mathcal{R}$, where each object $v \in \mathcal{V}$ belongs to one particular object type $\tau(v) \in \mathcal{A}$, each link $e \in \mathcal{E}$ belongs to a particular relation $\phi(e) \in \mathcal{R}$, and if two links belong to the same relation type, the two links share the same starting object type as well as the ending object type.*

When there are more than one type of node or link in the information network, it is called *heterogeneous information network*. In [8], Sun further defined meta-path as follows.

DEFINITION 2. ***(Meta-path)*** *A* meta-path *$\mathcal{P}$ is a path defined on the graph of network schema $T_G = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of $\dot{A}_1 \xrightarrow{R_1} \dot{A}_2 \xrightarrow{R_2} \ldots \xrightarrow{R_l} \dot{A}_{l+1}$, which defines a composite relation $R = R_1 \circ R_2 \circ \ldots \circ R_l$ between types $\dot{A}_1$ and $\dot{A}_{l+1}$, where $\circ$ denotes the composition operator on relations.*

Given a specific scholarly network, there can be many kinds of meta-paths. For example, $P^* \overset{w}{\to} A \overset{w}{\leftarrow} P^?$ is a simple meta-path on the scholarly network, denoting all the papers published by the seed paper' author. $P^*$ is the starting paper node (seed node) in this path. $P^?$ denotes the candidate publication node. More examples can be found in Table 1.

## 4. RESEARCH METHODS

### 4.1 Generate the Feedback Nodes

Generally, given user initial queries, a list of ranking publications would be found via text retrieval. Based on the top ranked documents, user would probably give explicit judgments on whether the related keywords, authors or venues are useful or not. However, explicit feedback is not easy to get. In this study, we propose methods to infer the implicit feedback nodes on the heterogeneous graph according to the given information.

The feedback is a collection of multiple nodes marked with useful (positive) or unuseful (negative) on the heterogeneous graph. We represent this collection as $NF$. $NF_P$ and $NF_N$ denote the positive and negative nodes set respectively. The kinds of feedback nodes in discussion include keyword (K), author (A) and venue (V).

#### *4.1.1 Generate the Positive Feedback Nodes*

Since we know the initial queries (i.e., author provided paper keywords) that the users should be most concerned with, it is reasonable to take the explicit keywords $K_P$ as the positive feedback nodes. Next, we will infer the positive authors and venues based on $K_P$. We deem that the authors or venues that are highly likely related to $K_P$ are positive as well. So we rank authors via meta-paths $K_P \xrightarrow{con} A^?$ and $K_P \overset{r}{\leftarrow} P \overset{w}{\to} A^?$, and take the top ranked $K_{pos}$ authors as the pseudo positive authors $A_P$. Similarly, we locate the positive venues via $K_P \xrightarrow{con} V^?$ and $K_P \overset{r}{\leftarrow} P \overset{p}{\to} V^?$, and select the top ranked $K_{pos}$ venues as the positive nodes $V_P$.

#### *4.1.2 Generate the Negative Feedback Nodes*

Intuitively, to generate the negative feedbacks, our basic assumption is that the negative nodes should be directly related to the searched results, but least relevant to the explicit positive keywords. First, based on text retrieval results, we define the top ranked $topK$ papers as $P_r$, and then we locate the keywords, authors and venues that are directly connected to $P_r$ via different meta-paths, $P_r \overset{r}{\to} K_r$, $P_r \overset{w}{\to} A_r$ and $P_r \overset{p}{\to} V_r$.

Next, we filter collections of $K_r$, $A_r$ and $V_r$. 1. Rank the keywords $K_r$ via the transition probability of meta-path $K_P \overset{con}{\to} P \overset{r}{\to} K_r$. Use the last ranked $K_{neg}$ keywords as the pseudo negative nodes $K_N$. 2. Similar to keywords, rank the authors $A_r$ via the transition probability of meta-path $K_P \overset{con}{\to} P \overset{w}{\to} A_r$, and use the last ranked $K_{neg}$ authors as the pseudo negative nodes $A_N$. 3. Rank the venues $V_r$ via $K_P \overset{con}{\to} P \overset{p}{\to} V_r$, and use the last ranked $K_{neg}$ venues as the negative nodes $V_N$. Here we use $K_P \overset{con}{\to} P$ instead of $K_P \overset{r}{\leftarrow} P$ because the "contribution" characterizes the importance of each paper, given a topic. It does not necessarily means paper is relevant to topic [2]. Even if one paper is not explicit relevant to some topic, it might also be important. The "contribute" conveys more information.

Thus, we obtain all the positive and negative feedback nodes. $NF_P$ includes $K_P$, $A_P$ and $V_P$. $NF_N$ contains $K_N$, $A_N$ and $V_N$.

### 4.2 Infer the Usefulness Probability of Node

Unlike previous studies, in this paper, the importance of nodes on scholarly network is not even. The usefulness probability of node $N_i$ is determined by the feedback nodes. Intuitively, if node $N_i$ is more closely related to the positive nodes, it could be more useful. Conversely, if $N_i$ is much closer to the negative nodes, and further away from the positive nodes, it indicates that $N_i$ may be not very useful. Therefore, the proximity between given node and feedback node set is very crucial. We should note that the usefulness probability of each node varies from different feedback node sets.

To infer the usefulness probability of node $N_i$, we adopt the sigmoid function $P_u(N_i) = \frac{1}{1+e^{-\alpha D(N_i)}}$ to convert the proximity into probability, where $\alpha$ controls the convergent rate (default is 1). In our assumption, if $N_j$ is positive node, $P_u(N_j) = 1$, otherwise $P(N_j) = 0$. $D(N_i)$ denotes the proximity between $N_i$ and the feedback node set $NF$. It can be derived from the following formula.

$D(N_i) = \frac{\sum_{N_k \in NF_N} d(N_i, N_k)}{|NF_N|} - \frac{\sum_{N_j \in NF_P} d(N_i, N_j))}{|NF_P|}$, where $|NF_N|$ and $|NF_P|$ represents the size of collection $NF_N$ and $NF_P$ respectively. $d(N_i, N_j)$ indicates the proximity between node $N_i$ and node $N_j$. In this paper, we will estimate the proximity $d(N_i, N_j)$ based on the paths $N_i \rightsquigarrow N_j$ on the graph. There could be lots of path instances connected node $N_i$ and $N_j$. If the length of path is too long, the influence would be too small to be considered. We assume the maximum of path length is 10. Then we select the shortest path and define its length as the proximity $d(N_i, N_j)$.

If $D(N_i)$ is negative, it reflects node $N_i$ is closer to negative nodes than positive ones, which means node $N_i$ could be less important, and vice versa. Particularly, if $D(N_j) \to +\infty$, it indicates that $N_j$ is far away from negative feedback nodes, so the importance of this node approach to 1; If $D(N_j) = 0$, it indicates that $N_j$ has the same distance to negative and positive nodes, then $P_u(N_j) = 0.5$ ; If $D(N_j) \to -\infty$, it indicates that $N_j$ is closest to negative feedback node, then $P_u(N_j) \to 0$.

## 4.3 Compute the Random Walk Probability Based on Meta-path

Meta-path illustrates how the nodes are connected in the heterogeneous graph. Once a meta-path is specified, a meta-path-based ranking function is defined, so that relevant papers determined by the ranking function can be recommended [3]. It turns out that meta-path based feedback on heterogeneous graph performs better than other methods (PageRank) based PRF [2]. Random walk on heterogenous network can explore more global information, combining multiple feedback nodes, which might be very important for the recommendation tasks.

In order to quantify the ranking score of candidates relevant to the seeds following one given meta-path, a random walk based approach was proposed in [2]. The relevance between $P^*$ and $P^?$ can be estimated via $s(a_i^{(1)}, a_j^{(l+1)}) = \sum_{t=a_i^{(1)} \rightsquigarrow a_j^{(l+1)}} RW(t)$, where $t$ is a path instance from node $a_i^{(1)}$ to $a_j^{(l+1)}$ following the specified meta-path, and $RW(t)$ is the random walk probability of the instance $t$.

Suppose $t = (a_{i1}^{(1)}, a_{i2}^{(2)}, \ldots, a_{il+1}^{(l+1)})$, the random walk probability can be computed via $RW(t) = \prod_j w(a_{ij}^{(j)}, a_{i,j+1}^{(j+1)})$. While this formula only considers the weight of link on the path instance. Based on our hypothesis, the node usefulness probability has a great effect on the path importance. So in this study, we propose a novel random walk function as follows.

$RW(t) = \prod_j (\beta \cdot w(a_{ij}^{(j)}, a_{i,j+1}^{(j+1)}) + (1-\beta) \cdot P_u(a_{i,j+1}^{(j+1)}))$, where $P_u(a_{i,j+1}^{(j+1)})$ is the usefulness probability of the node $a_{i,j+1}^{(j+1)}$ on the path (derived from section 4.2), and $\beta$ determines which factor is more important. Theoretically, we need to tune $\beta$ for each meta-path to optimize the weight of each sub-meta-path. For this study, we set $\beta = 0.6$.

Then, the random walk probability will be decided by the transition probability and the usefulness probability of the node on the path instance. In this paper, we use eight meta-paths to investigate the novel random walk method with node feedback information for citation recommendation. All the meta-paths are listed in Table 1.

# 5. EXPERIMENT

## 5.1 Data Preprocessing

We used 41,370 publications (as candidate citation collection), published between 1951 and 2011, on computer science for the experiment (mainly from the ACM digital library). As [2] introduced, we constructed the heterogeneous graph shown in Figure 1 and Table 2.

For the evaluation part, we used a test collection with 274 papers. The selected papers have more than 15 citations from the candidate citation collection.

## 5.2 Generate Feedback Nodes

Attaining different types of feedback information is the most important part in this research. Since it is not available to get the user judgments right away. We used the method introduced in section 4.1 to create positive and negative feedback nodes. As aforementioned, the collection $K_P$ is the set of user given keywords. It is explicit positive feedbacks. While $A_P$ and $V_P$ can be derived by their connectivity to set $K_P$ based on the heterogeneous graph. Here we set $K_{pos} = 10$, and take the top 10 ranked authors/ venues as the implicit positive feedbacks.

Next, we produced the implicit negative feedback nodes. Through the text retrieved results, we grabbed the top ranked papers as $P_r$ ($topK = 20$). Then we located the list of keywords/ authors/ venues which have direct correlations to $P_r$, but the least relevance to $K_P$. Find the last ranked $K_{neg} = 10$ and used them as $K_N$, $A_N$ and $V_N$ respectively.

## 5.3 Experiment Result

In the evaluation part, we experimented with 8 different meta-paths. For each meta-path, two sets of results were shown on row 'N' and 'Y' in Table 3. The 'N/Y' column in Table 3 indicates whether we use the positive and negative feedback nodes or not for computing the path importance. 'N' indicates that the result was from the baseline in [2], while 'Y' means multiple feedback nodes were employed and the node influence was appended into the final random walk function. MAP and NDCG are used as the ranking function training and evaluation metrics. For MAP, binary judgment is provided for each candidate cited paper (cited or not cited). NDCG estimates the cumulative relevance gain a user receives by examining recommendation results up to a given rank on the list. We used an importance score, 0-4, as the candidate cited paper importance to calculate NDCG scores. Apparently, in most cases, row 'Y' significantly outperforms row 'N' , which shows that the positive/negative feedbacks enhance the random walk performance quite well. We also used t-test to verify this improvement and most meta-paths are significantly refined.

# 6. CONCLUSION AND LIMITIONS

In this study we use multiple kinds of feedback nodes and propose a new method to enhance the meta-path-based random walk performance. The new random walk function considers both transition probability and node usefulness probability on the path instance. We find that the node influence varies from the set of feedback nodes, which could be inferred based on the explicit user queries via a series of steps. Experimental results with ACM data illustrate that the new approach with positive/negative feedback information helps to improve the performance of meta-path-based recommendation.

For further study, we will continue this approach based on real user explicit feedbacks and design the personalized recommendation model to improve user experience. Not only the node usefulness is related to the feedback nodes, but also the weight of each relation type may be affected by the feedback nodes or retrieval task. If the retrieval task is to search the relevant papers based on given authors, the author feedback nodes will be more useful for "writtenby" relation, "writtenby" and "co-author" relation might be more important. This hypothesis will be discussed in the next step. Besides, more sophisticated inference models will be adopted which may enhance the ranking performance.

## 7. FIGURES AND TABLES

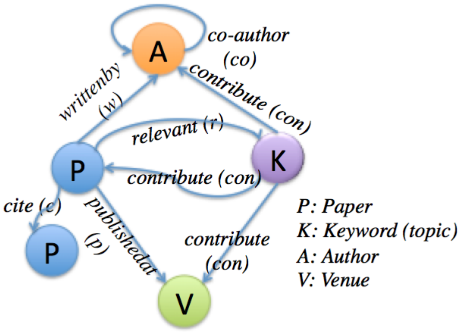


**Figure 1: Heterogeneous Bibliographic Graph**

**Table 1: All the meta-paths used in this study**

| NO. | Meta-path | Feedback ranking hypothesis |
|---|---|---|
| 1 | $P^* \xrightarrow{w} A \xleftarrow{w} P^?$ | Relevant paper's author's other papers can be relevant |
| 2 | $P^* \xrightarrow{c} P^?$ | Relevant paper's cited papers can be relevant |
| 3 | $P^* \xrightarrow{c} P \xrightarrow{c} P^?$ | Relevant paper's cited paper's cited paper can be relevant |
| 4 | $P^* \xrightarrow{c} P \xrightarrow{w} A \xleftarrow{w} P^?$ | Relevant paper's cited papers' authors' papers can be relevant |
| 5 | $P^* \xrightarrow{w} A \xrightarrow{co} A \xleftarrow{w} P^?$ | Relevant paper's author's co-author's papers can be relevant |
| 6 | $P^* \xrightarrow{w} A \xleftarrow{w} P \xrightarrow{c} P^?$ | Relevant paper's author's cited papers can be relevant |
| 7 | $P^* \xrightarrow{p} V \xleftarrow{p} P \xrightarrow{c} P^?$ | Paper can be relevant if it is cited by the ones published at the same venue as the relevant paper |
| 8 | $P^* \xrightarrow{p} V \xleftarrow{p} P \xrightarrow{w} A \xleftarrow{w} P^?$ | Paper can be relevant if its authors' papers are published at the same venue as the relevant paper |

**Table 2: Graph statistics**

| Node/Edge | Number | Description |
|---|---|---|
| P | 41,370 | Paper |
| A | 63,323 | Author |
| V | 369 | Venue |
| K | 3,911 | Keyword |
| $P \xrightarrow{c} P$ | 168,554 | Paper cites another paper |
| $P \xrightarrow{w} A$ | 105,992 | Paper is written by an author |
| $P \xrightarrow{p} V$ | 41,013 | Paper is published at venue |
| $A \xrightarrow{co} A$ | 239,744 | Co-author relationship |
| $P \xrightarrow{r} K$ | 587,252 | Paper is relevant to keyword(topic) |
| $K \xrightarrow{con} P$ | 3,577,111 | Keyword (topic) is contributed by paper |
| $K \xrightarrow{con} A$ | 2,397,205 | Keyword (topic) is contributed by author |
| $K \xrightarrow{con} V$ | 18,450 | Keyword (topic) is contributed by venu |

**Table 3: Meta-path Based Random Walk Performance Comparison($|P^*| = 10$)**

| NO. | N/Y | MAP | MAP@5 | MAP@10 | NDCG | NDCG@5 | NDCG@10 |
|---|---|---|---|---|---|---|---|
| 1 | N | 0.0277 | 0.0085 | 0.0129 | 0.1035 | 0.0306 | 0.0394 |
|  | **Y** | **0.0365** *** | **0.015** *** | **0.0211** *** | **0.1149** *** | **0.0459** ** | **0.0565** *** |
| 2 | N | 0.1315 | 0.0552 | 0.0773 | 0.2193 | 0.1427 | 0.1548 |
|  | **Y** | **0.1459** *** | **0.0678** *** | **0.0904** *** | **0.2307** ** | **0.1656** *** | **0.1705** ** |
| 3 | N | 0.0744 | 0.0306 | 0.0404 | 0.1539 | 0.0689 | 0.0766 |
|  | **Y** | **0.0948** *** | **0.0441** *** | **0.0582** * | **0.1707** *** | **0.0945** * | **0.1002** ** |
| 4 | N | 0.027 | 0.0042 | 0.0076 | 0.1378 | 0.0146 | 0.025 |
|  | **Y** | **0.038** *** | **0.0109** *** | **0.0153** *** | **0.1521** *** | **0.0318** *** | **0.0387** *** |
| 5 | N | 0.0436 | 0.0121 | 0.0187 | 0.1672 | 0.0476 | 0.0585 |
|  | **Y** | **0.0561** *** | **0.0257** *** | **0.0328** *** | **0.1854** *** | **0.0867** *** | **0.0885** *** |
| 6 | N | 0.0327 | 0.0234 | 0.03 | 0.0734 | 0.0693 | 0.0748 |
|  | **Y** | **0.0872** *** | **0.0359** *** | **0.0471** *** | **0.1962** *** | **0.0805** * | **0.09** * |
| 7 | N | 0.0238 | 0.0083 | 0.0097 | 0.1529 | 0.0216 | 0.0224 |
|  | **Y** | **0.0373** *** | **0.0133** *** | **0.0163** *** | **0.1718** *** | **0.0317** ** | **0.0344** ** |
| 8 | N | 0.0092 | 0.0005 | 0.0007 | 0.1397 | 0.0011 | 0.0013 |
|  | **Y** | **0.012** *** | **0.0011** *** | **0.0017** *** | **0.1476** *** | **0.0027** *** | **0.0045** *** |
| $p < 0.05$: *, $p < 0.01$: **, $p < 0.001$: *** |  |  |  |  |  |  |  |

## 8. REFERENCES

[1] P. Bhatnagar and N. Pareek. Improving pseudo relevance feedback based query expansion using genetic fuzzy approach and semantic similarity notion. *Journal of Information Science*, page 0165551514533771, 2014.

[2] X. Liu, Y. Yu, C. Guo, and Y. Sun. Meta-path-based ranking with pseudo relevance feedback on heterogeneous graph for citation recommendation. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 121–130. ACM, 2014.

[3] X. Liu, Y. Yu, C. Guo, Y. Sun, and L. Gao. Full-text based context-rich heterogeneous network mining approach for citation recommendation. In *ACM/IEEE Joint Conference on Digital Libraries*, 2014.

[4] Y. Lv and C. Zhai. Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 255–264. ACM, 2009.

[5] H. Muller, W. Muller, S. Marchand-Maillet, T. Pun, and D. M. Squire. Strategies for positive and negative relevance feedback in image retrieval. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 1, pages 1043–1046. IEEE, 2000.

[6] H. Saneifar, S. Bonniol, P. Poncelet, and M. Roche. Enhancing passage retrieval in log files by query expansion based on explicit and pseudo relevance feedback. *Computers in Industry*, 65(6):937–951, 2014.

[7] Y. Sun and J. Han. Meta-path-based search and mining in heterogeneous information networks. *Tsinghua Science and Technology*, 18(4), 2013.

[8] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. PathSim: Meta path-based top-k similarity search in heterogeneous information networks. In *Proc. 2011 Int. Conf. Very Large Data Bases*

*(VLDB'11)*, Seattle, WA, 2011.

[9] Y. Sun, B. Norick, J. Han, X. Yan, P. S. Yu, and X. Yu. Integrating meta-path selection with user-guided object clustering in heterogeneous information networks. In *Proc. of 2012 ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD'12)*, Beijing, China, 2012.

[10] X. Wang, H. Fang, and C. Zhai. A study of methods for negative relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 219–226. ACM, 2008.

[11] Y. Xu, G. J. Jones, and B. Wang. Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 59–66. ACM, 2009.

[12] X. Yu, X. Ren, Y. Sun, B. Sturt, U. Khandelwal, Q. Gu, B. Norick, and J. Han. Recommendation in heterogeneous information networks with implicit user feedback. In *Proc. of 2013 ACM Int. Conf. Series on Recommendation Systems (RecSys'13)*, pages 347–350, Hong Kong, 2013.

## Author Index