IVIS: SEARCH AND VISUALIZATION ON HETEROGENEOUS INFORMATION NETWORKS

BY

YINTAO YU

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Adviser:

Professor Jiawei Han

Abstract

We build a system to support search and visualization on heterogeneous information networks. We first build our system on a specialized heterogeneous information network: DBLP. The system aims to facilitate people, especially computer science researchers, toward a better understanding and user experience about academic information networks. Then we extend our system to the Web. Our results are much more intuitive and knowledgeable than the simple top-k blue links from traditional search engines, and bring more meaningful structural results with correlated entities. We also investigate the ranking algorithm, and we show that the personalized PageRank and proposed Hetero-personalized PageRank outperform the TF-IDF ranking or mixture of TF-IDF and authority ranking. Our work opens several directions for future research.

To my family.

Acknowledgments

I wish to thank professor Jiawei Han for his insightful instruction and persistent support.

I also thank Yizhou Sun, Binbin Liao, Tianyi Wu, Hongbo Deng and other colleagues in DAIS group for their discussion and help.

Table of Contents

List of Figures

Chapter 1

Introduction

Information networks, including the Web [7, 15], social networks [6, 11], bibliographic networks [9] and many other networks [8] are ubiquitous and form a critical component of modern information infrastructure. With the rapid proliferation of information networks, there is an urgent need to effectively organize and search the rich data behind different networks. However, this task becomes challenging especially when the information networks examined are large and diverse. In this thesis, we first focus ourselves on a specialized *heterogeneous information network*: DBLP, upon which we are planning to support multi-dimensional search and visualization to help people, especially computer science researchers, toward a better understanding and user experience about a specialized information network in computer science domain. Our design principles and algorithms can be easily generalized to other complicated information networks. In the later part of this thesis, we will extend our design to the Web.

DBLP[1] is a well-known scientific publication repository which collects recent years' journal and conference proceedings in computer science. Upon DBLP, multi-faceted textual based search is provided to help users find relevant scientific publications. However, we notice that the current service provided in DBLP and FacetedDBLP is somehow too limited and isolated, which cannot satisfy our information need on this large information network: (**1**) it only provides simple keyword-based textural search on raw documents, which has proven to be primitive and error-prone; (**2**) For output ranking, it employs naive ranking strategies like simple publication counts ranking; (**3**) the output results returned are nothing but top-k blue links like traditional search engines. In summary, DBLP provides very limited search services which is hard to meet our daily requirements on the information networks.

[1] http://www.informatik.uni-trier.de/~ley/db/

In this thesis, we propose our multidimensional search and visualization system, iVis [2], on the DBLP information network. The contributions in this thesis can be summarized as follows:

1. We want to provide the comprehensive multi-dimensional entity search beyond simple textual-based matching. We support in iVis as *inputs* three different entities: **Author**, **Conference** and **Keyword** and the potential *outputs* of each entity query are organized into a multi-dimensional graphical view, which are much more intuitive and attractive than the simple top-k blue links, and bring more meaningful structural results with correlated entities;

2. We retrieve the top-k entities by random walk-based algorithm instead of simple ranking used by DBLP. We make use of the information network structure to propagate the scores among different entities. We show our method is much better than simple ranking methods.

3. We display the top-k entities as a structural graph of correlated entities with the advanced visualization techniques. As the graphical results show the multi-dimensional view of different entities, we support common OLAP operations, like roll-up, drill-down to reorganize the results with multiple granularity. Controlled by users at will, our iVis system can bring the previously isolated entities into one integrated, interlinked and multidimensional entity graph, upon which users can issue queries beyond simply typing keywords.

As an example, let's think about issuing a query "Query Processing" as a keyword entity and the returned results include both the top-l most relevant conference entities, like SIMGOD, VLDB, ICDE and EDBT, etc., and the top-k most relevant author entities, like "Surajit Chaudhuri", "Joseph M. Hellerstein" and "Divesh Srivastava", etc. All these entities are computed based on our top-k entity retrieval algorithm. The entities are represented as vertices of a graph with different colors depicting different entity types and different size depicting different importance. These entities are interlinked with edges representing correlations. The information displayed on the edges includes the number of papers "co-authored"(for author-author edge) or "published" (for author-conference/journal edge) between the two corresponding interlinked entities and the detailed publication records (title, conference/journal, year) etc.

[2] http://entityvis.com/

The distances between different entities represent the similarity while the edges also have properties showing the detailed publication statistics between entities involved. Users can browse the results graphically and perform multi-dimensional operations like roll-up from lower level author entities to higher level research communities. The structural results change accordingly and show the updated new results dynamically. Users can issue new queries by simply clicking the nodes of the graph, instead of typing plain text, so that the browsing and query processing of iVis are naturally integrated within one platform.

We may notice that the queries posed on iVis go beyond textual queries to entities (author, conference and keyword). Our goal is toward a better understanding of the data underlying the DBLP information network, and a better organization of the output data which brings much more meaningful structural results than the naive top-k blue links provided by the traditional DBLP. iVis can be regarded as the prototype system with first attempt toward effectively querying , retrieval and visualization on large information networks.

The reminder of this thesis is organized as follows. In Chapter 2, we will describe the framework of our system, iVis. We will focus on the key functionality implementation and the architectural design of iVis. In Chapter 3, we will present our ranking algorithm. In Chapter 4, we will describe the extension to the Web. In Chapter 5, we will give some further discussion. Chapter 6 concludes this thesis.

Chapter 2

Overall Framework

In this chapter, we will detail our multidimensional search and visualization system, iVis, on the DBLP information network. First of all, we will examine the nuts and bolts of our iVis system by a thorough demonstration of its different functionalities. Then we will elaborate our design principles and architecture of iVis. We then illustrate our research work on retrieval and visualization of DBLP information network, which are the key technical merits of our system.

2.1 iVis: Nuts and Bolts

Our iVis system supports effective search and browsing on the DBLP information network. We model and classify the underlying entities of the DBLP information network into three different categories: *author*, *conference*, and *keyword*. Different entities are not independent, but interlinked with other entities with specific relationships. The relationship among different entities is solely determined by the publication information stored in the DBLP information network. For example, if author a_1 and author a_2 co-authored a paper p with a keyword k, and the paper p is published in a specific conference (journal) c, then all these entities are interlinked via the paper p, forming a star-schema with the center as p. In this way, different entities are abstracted as nodes and their interlinked relationship are modeled as edges, so that the whole DBLP information network is represented as a huge graph. Our search and browsing operations are therefore designed on this graph-structured information network. It is worth noting that the graph structure renders itself as an impressive visualization tool, upon which both exploration and query processing can be effectively supported. In comparison with the traditional text-based retrieval system, our iVis is much more intuitive, functional and user-friendly.

As illustrated in Figure 2.1, Our iVis system provides a query input box for users. In it, users

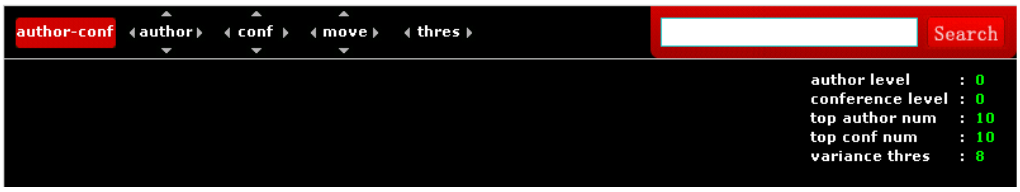
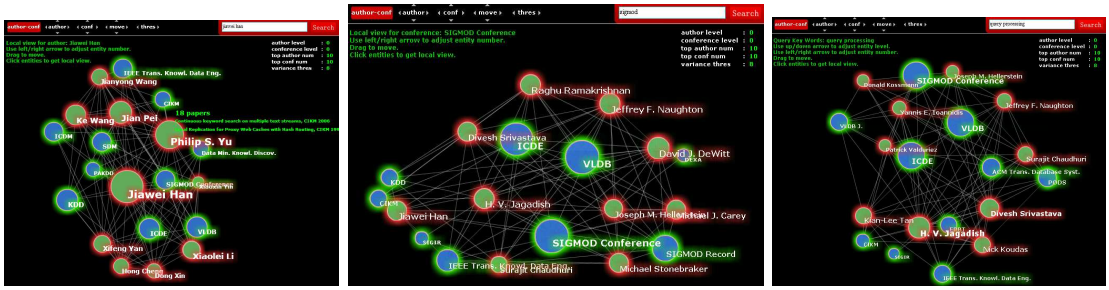


Figure 2.1: The Main Interface of iVis

can issue three different kinds of queries upon the underlying information network: *author* entity query, *conference* entity query and *keyword* query, which right correspond to the three different kinds of entities supported in the DBLP information network. For example, users can issue "Jiawei Han" as an author entity query in the query box. Similarly, "SIGMOD", "VLDB", "KDD" are potential candidate queries for conference entities, and "query processing", "data mining" can be keyword queries issued by different users. As illustrated in Figure 2.2, we tested different sample queries on our iVis system. Figure 2.2(a) shows the graphical results for the author query "Jiawei Han", with the parameters $k = 10$ and $l = 10$. It means that we are interested in the top-10 most relevant authors and conferences w.r.t. the researcher "Jiawei Han". We use green nodes to represent different author entities and blue nodes to represent different conference entities. Different correlated entities are interlink with edges. The node size indicates the relative rankings w.r.t. the query, i.e., the higher the rank, the larger size the node will be. From the diagram, we can see that, Prof. Jiawei Han actively participated in major databases conferences, like SIGMOD, VLDB and ICDE, together with major data mining conferences, like KDD, ICDM and SDM. The results are very intuitive and meet our commonsense pretty well. The detailed publication information is stored along the edges of the result graph. For example, if the mouse is moving on the edge between "Philip S. Yu" and "CIKM", a message box is shown that there exist 17 papers in total between these two entities and the publication entry can be viewed one-by-one. Figure 2.2(b) shows the query results for the conference entity query "SIGMOD". As illustrated, "VLDB" and "ICDE" are very close to SIGMOD as their corresponding node sizes are pretty large, compared with others. In the mean time, those researchers, like "David J. Dewitt", "H. V. Jagadish" etc. are avid database researchers and they play a key role in SIGMOD conferences. Figure 2.2(c) represents the results for the keyword query "query processing", which is a popular buzz word in database research. The results again very well return the top ranked author and conferences entities which have a close

relationship with the query processing research.

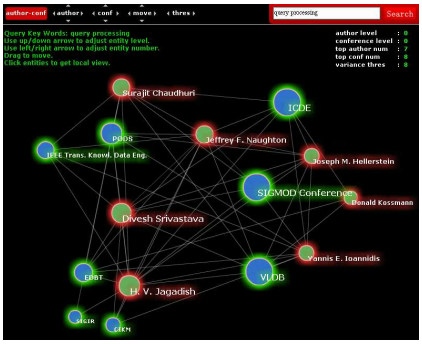


(a) Author Query: Jiawei Han (b) Conference Query: SIGMOD Conference (c) Keyword Query: Query Processing

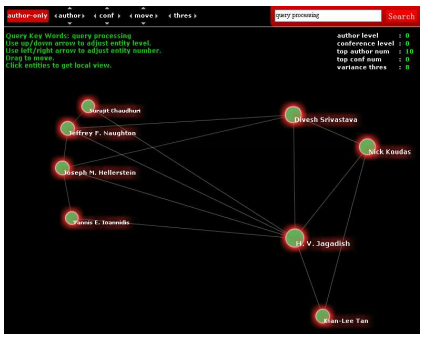
Figure 2.2: Sample Entity Queries and Their Results in iVis

Once the query has been processed and the ranked results are returned as outputs, users can choose two different options to visualize the final results: *author-conf* and *author-only*. These two options are actually two different views which keep track of snapshots of the final results from different perspectives. The author-only option select authors only as the results in order to show the top-k relevant authors in terms of queries issued. This option actually generates a localized co-authorship graph w.r.t. the query. And the author-conf combines the views of both author and conference to show a comprehensive snapshot for a query. The ranking parameters, k and l, can be specified and tuned by users per their interest, and the graphical results can be moved at will by users in any direction. Figure 2.3(a) shows the result for the keyword query "query processing" with the parameters $k = 7$ and $l = 8$, i.e., we only want to see the top-7 relevant authors and top-8 conferences in the research area of query processing. Users can tune the parameters by simply clicking the buttons of author and conference. Figure 2.3(b) demonstrates the result for "query processing" in the author-only view. Under this view, the co-authorship graph is depicted.

Note DBLP only provides the publication related information in the raw format. In this way, users can easily be buried by the huge amount of publications and confused about the search and browsing direction. In order to alleviate this problem, iVis provides a multidimensional exploration approach for users to better digest the information stored in the DBLP information network. For both author and conference entities, two OLAP-style operations: roll-up and drill-down are provided in order to help users explore the DBLP information network in different granularity. For example, different conferences can be generalized to broad research areas and different authors



(a) Keyword Query: Query Processing with $k = 7$ and $l = 8$



(b) Keyword Query: Query Processing in the author-only view

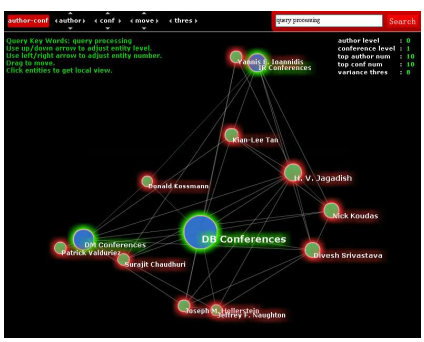
Figure 2.3: A Sample Keyword Query with Different Parameters and Different Views

can be generalized into specific research topics they participated in. As shown in Figure 2.4(a), different detailed conferences are summarized by the roll-up operation on the conference dimension to broad research areas, such as Database, Data Mining and Information Retrieval. If we go one step further to roll-up on the author dimension, we can get the result shown in Figure 2.4(b). To keep the result neat and clear, we set $k = 5$. As shown, different research communities within the research of query processing are generalized. For example, the author "Surajit Chaudhuri" is classified into the "Work flow model" researcher, while "Jeffrey F. Naughton" belongs to "DB/XML query optimization" researcher. In this way, the users can browse the information network among different granularity of the dimensions supported in the DBLP information network, instead of checking the publication one-by-one laboriously. Meanwhile, the multidimensional navigation via roll-up and drill-down operations can reformulate the DBLP information network based on the users' interest. Users can first explore the information network in a general level and then drill down to the particulars they are really interested in. The general research areas and research topics are pre-computed by RankClus [17] and NetClus [18, 21], which will be elaborated in the following sections.

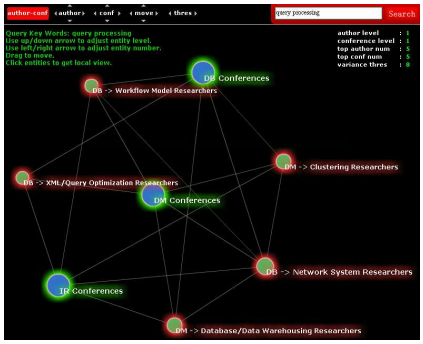
2.2 iVis: System Architecture

The system architecture of iVis is shown in Figure 2.5. We will explain the internals and mechanism of iVis by examining how a query is performed through our system.

Once the system receives a query, it will first come to the query processing module. This module



(a) Keyword Query: Query Processing with Rolling-up on the Conference Dimension



(b) Keyword Query: Query Processing with Rolling-up on Both Author and Conference Dimension

Figure 2.4: A Sample Keyword Query in the Multidimensional View with Roll-up and Drill-down Supported

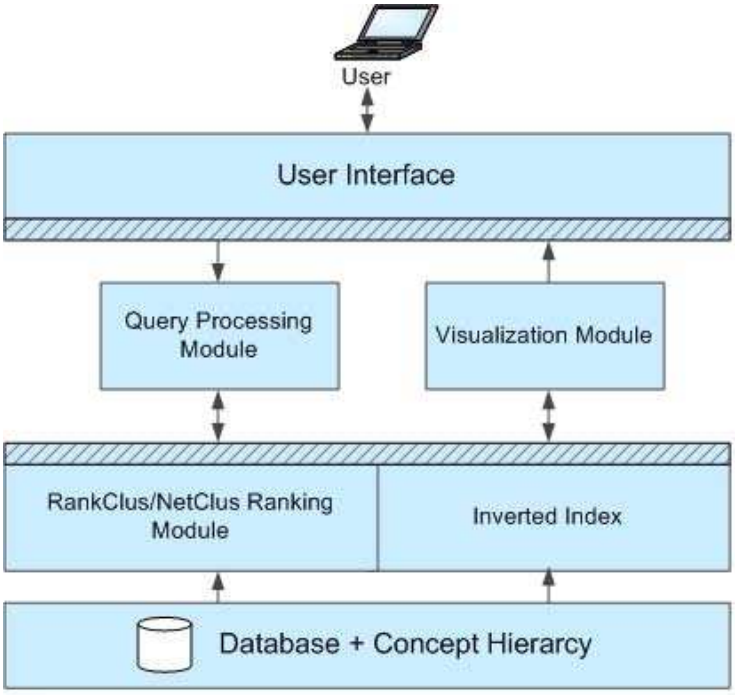


Figure 2.5: The Architecture of iVis

will understand the query type and will process other query information such as top-k number and current concept hierarchy level as well. Then along with the RankClus/NetClus module and the pre-computed inverted index, the system will be able to retrieve the top-k entities from the databases and the relevance scores of each pair of the retried entities. Finally the visualization module will deliver an integrated linked and somewhat clustered graphical view.

Among different functional modules of iVis, the entity ranking module and the visualization module provide the key functionalities and are our major technical merits. We will explain in detail in the following sections, respectively.

2.3 Entity Ranking of iVis

The ranking method will be discussed in Chapter 3.

2.4 Network Visualization of iVis

In iVis the visualization task is focused on visualizing hierarchical heterogeneous information network graphs, which consists of entities of different types, and links between them. For displaying such an arbitrary network graph, the most challenging problem is how to arrange the nodes in a 2D or 3D (2D as we use in the paper) space that effectively shows the information (links, hubs, and clusters) of the graph. Other supplemental issues include presenting multi-dimensional attributes of the graph within the 2D space.

There are many existing techniques for obtaining an aesthetically pleasing drawing of a given graph $G = (V, E)$, such as spring embedder model, graph embedder (GEM), etc. And there are many criteria existing for judging the aesthetics of a graph drawing [10]. In considering the properties of the graph we generate, and the algorithmic efficiency, we selected the LinLogLayout algorithm for the graph layout problem. As one of the many graph layout algorithm, the LinLogLayout algorithm [13, 12] holds the property that it naturally represent the cluster structures of graphs by grouping densely connected nodes and separating sparsely connected nodes. This is the very reason that LinLogLayout method is chosen in our application since that the clustering phenomenon reflects several of the properties, i.e., power-law distribution, community structure,

in a social network graph. The main contribution of the LinLogLayout algorithm is on the LinLog energy model; algorithms for minimizing the energy models are borrowed from the hierarchical algorithm of Barnes and Hun [14]. This algorithm is intrinsically a heuristic algorithm for searching better solutions that does not guarantee to find global minima. It achieves $O(E + VlogV)$ time complexity per iteration.

As another aspect of network visualization, we use some non-spatial dimensions of the graph to represent other information. We utilized the following intuitive visual affects to represent information.

- **Color**: to distinguish entities of difference types in the visualization we made to the DBLP information network, conference nodes are blue and author nodes are red;
- **Size**: size is related to nodes in the graph. Intuitively, node circles with large diameter represent that the corresponding nodes has high weight, and nodes with small diameters represent for low weight;
- **Distance**: As part of the layout algorithm, linked nodes with long distance indicate that their bond, or the weight of the link in relatively low; and linked nodes with short distances indicate they're closely related.

Chapter 3

Ranking Algorithm for Queries

The major deficiency of DBLP is its lack of effective ranking strategies. When a textual query is issued, simple text matching is performed based on traditional inverted-index and results are returned with no ranking considered. For specific authors or conferences, the relevant authors and conferences are returned solely based on the number of publication co-authored or published. In order to improve the ranking effectiveness of the DBLP information network and discover the relevance between different entities, we investigate PageRank-based ranking algorithms which goes beyond the simple TF-IDF ranking mechanism. They can makes use of the whole network structure to reinforce our ranking of entities. We propose a modification of personalized Pagerank and show it outperforms other methods.

3.1 Heterogeneous Information Network Definition

Definition 1 ***Information network.*** *An* information network *is defined as a multi-digraph* $G = (V, E, W; \phi : V \to \mathcal{A}, W : E \to \mathbb{R}^+)$*, where* V *is the vertex set,* E *is the link set,* $\mathcal{A}$ *is the type set for vertices.* ϕ *is the type mapping function, and* W *is a weight function defined on* E*.*

We observe that if the number of types $|\mathcal{A}| > 1$, then G is a **heterogeneous information network**. Otherwise, G is a **homogeneous information network**. In the case of bibliographic network, the node type set is $\mathcal{A} = \{paper(P), author(A), venue(C), term(T)\}$. Since the example is unweighted, all link weights may be set to 1. However, it is easy to conceive of another co-authorship network, in which a link between two authors may have a weight which is proportional to the number of papers coauthored by them. Such a representation provides users the ability to powerfully model a wide variety of database and web constructs in networks formed by simply recognizing the different types and relations. For example, the model can represent any database

Author	Conf	Term
Jon M. Kleinberg	AAAI	network
Jennifer Golbeck	WWW	social
Christos Faloutsos	IJCAI	data
Lise Getoor	ICDE	learn
Ravi Kumar	KDD	web
...	...	...

Author	Conf	Term
Jure Leskovec	KDD	network
Christos Faloutsos	AAAI	social
Jon M. Kleinberg	IJCAI	data
Eric Horvitz	WWW	learn
Ravi Kumar	ICDE	mine
...	...	...

Table 3.1: Results for queries "social network" and "Jure Leskovec"+"KDD"+"social network"

ER-schema, linked social web construct, or a sensor or military network with logical links between entities.

For convenience we will use $W_{v_i v_j}$ to denote the weight of an edge $\langle v_i, v_j \rangle$ in E from now on.

3.2 Problem Definition

In a heterogeneous information network, given a query, return the most relevant entities in each type with respect to the query. The query can be key words only, or a combination of key words and entities in the information network. The query entities are not necessarily from the same type. For instance, the query can be simply keywords "social network". Or the query can be author entity "Jure Leskovec" + conference entity "KDD" + key words "social network", in which multiple types of entities are part of the query. A running example of the top-5 entities in each type retrieved for these two queries are shown in Table 3.1.

3.3 Ranking Algorithm

TF-IDF Our first baseline, *TF-IDF*, replies solely on aggregation of text of each entity. For example, for each author or conference, we treat them as an aggregated document by concatenating corresponding papers, while each paper is merely a bag of words. Thus, each entity is just a document as in traditional information retrieval and can be retrieved by TF-IDF ranking. The drawback of this method is that we no longer use any link structure of the information network for retrieving the top entities. Also an unknown author who only published one or two papers but happen to contain the query words will be ranked very high, since his/her aggregation document is very short. Also note this method only supports pure keyword query.

TF-IDF + authority ranking To handle the drawback of naive TF-IDF, in this method,

we first pre-compute each entity's authority using NetClus [18]. Then we combine the authority score and TF-IDF score together with the ranking score. Since TF-IDF score has to be computed, this method also only supports pure keyword query. The drawback of this method is hard to train the parameter when combining the score. When the query words are general terms, a higher-weight towards authority score is preferred; when the query words are specific topics, a higher weight towards TF-IDF score is preferred. In the current experiment, the combined score is calculated as ($\text{TF}-\text{IDF score}^2 \times$ authority score), which turns out to be generally better than simply $\text{TF}-\text{IDF}$ score $\times$ authority score (we can not add them together because the two scores are not in the same scale).

Personalized PageRank In this method, we apply personalized PageRank as if the network is homogeneous. However, mapping the original heterogeneous information network to homogeneous one will cause information loss. In the current experiment, the initial prior weights are evenly distributed. For example, given the query "Jure Leskovec"+"KDD"+"social network", author entity "Jure Leskovec", conference entity "KDD", and term "social" and "network" all have the same weight 0.25. Formally, assume λ_0 is the initial prior weight vector, and the sum of all elements in λ_0 equals to 1. In i-th iteration, we calculate $\lambda_i = (1-d)\lambda_0 + d\lambda_{i-1}A$ until λ converges. Here, d is the damping factor and A is the transition matrix, satisfying:

$$A(i,j) = \begin{cases} 1 & \text{, if } |I(v_i)| = 0 \text{ and } i = j \\ 0 & \text{, if } |I(v_i)| = 0 \text{ and } i \neq j \\ \dfrac{W_{v_i v_j}}{\sum\limits_{v_k \in I(v_i)} W_{v_i v_k}} & \text{, otherwise.} \end{cases}$$

, where $I(v_i)$ is the neighborhood set of v_i, *i.e.*$I(v_i)$ is the set of objects have link to v_i. It is not hard to find that the sum of the elements in each row of A is 1.

Hetero-Personalized PageRank This is a modification of personalized PageRank we propose. The difference from standard personalized PageRank is that we distinguish different types of edges. For example, in standard personalized PageRank, given one paper links to 2 authors, 1 conference, and 6 terms in an unweighted heterogeneous bibliography network, each edge is equally important; therefore, the row corresponding to this paper in the transition matrix con-

tains 9 non-zero element, all equally $\frac{1}{9}$. However in *Hetero-Personalized PageRank*, we first set the weight for each type of edge. Here the simplest version can be: for an edge from paper, the edges linking to author, conference, and term have equal weight $\frac{1}{3}$, then the same type of edges share the weight again. In other words, each edge linking to author now has $\frac{1}{3} \times \frac{1}{2} = \frac{1}{6}$ weight; the edge linking to conference has $\frac{1}{3}$ weight; and each edge linking to term has $\frac{1}{3} \times \frac{1}{6} = \frac{1}{8}$ weight. The advantage of this adjustment is the propagation of the score will no longer be easily biased. The score of a paper will no longer be propagated to author type too much if it has too many authors, and similarly for term. Formally, we define $\gamma : a \times b \to \mathbb{R}^+, a \in \mathcal{A}, b \in \mathcal{A}$, which gives the weight of the set of edges linking from an object with type a to an object with type b. In the foregoing example, we have $\mathcal{A} = \{paper(P), author(A), venue(C), term(T)\}$, and γ is given as $\gamma(P, A) = \gamma(P, C) = \gamma(P, T) = \frac{1}{3}$ (actually in this example, the edges are undirected and these are the only three types of edges we have in the network). And now the transition matrix A' becomes:

$$A'(i,j) = \begin{cases} 1 & \text{, if } |I(v_i)| = 0 \text{ and } i = j \\ 0 & \text{, if } |I(v_i)| = 0 \text{ and } i \neq j \\ \dfrac{\gamma(\phi(x_i), \phi(x_j))}{\sum\limits_{a \in \mathcal{A} \text{ and } \exists v_k \text{ s.t. } (v_k \in I(v_i) \text{ and } \phi(v_k) = a)} \gamma(\phi(x_i), a)} \cdot \dfrac{W_{v_i v_j}}{\sum\limits_{v_k \in I(v_i) \text{ and } \phi(v_k) = \phi(v_j)} W_{v_i v_k}}, & \text{otherwise.} \end{cases}$$

, where $I(v_i)$ is the neighborhood set of v_i. Again the sum of the elements in each row of A' is 1. And the iteration formula is still $\lambda_i = (1 - d)\lambda_0 + d\lambda_{i-1}A'$ until λ converges.

3.4 Experiment Setup

We evaluate our methods using DBLP data and build the heterogeneous network using Star-Schema as in [18]. The dataset we are using here is a subset of whole DBLP containing 20 conferences, 28,569 papers, 2,8702 authors, 9,291 terms (stemmed and removed stop words), and 270,959 edges.

It is difficult to evaluate the quality of output rankings due to the scarcity of data that can be examined publicly. The ground truth is manually created through the method of pooled relevance judgments together with human judgments. Currently we only have labeled data for authors. Thus,

Query	# of Experts
information extraction	20
machine learning	42
semantic web	45
support vector machine	31
information retrieval	23
language model information retrieval	12
face recognition	21
semisupervised learning	21
reinforcement learning	17
privacy preservation	17

Table 3.2: Benchmark dataset

	P@10	P@20	R-prec	MAP
TF-IDF	0.24	0.165	0.167	0.119
TF-IDF+authority	0.25	0.195	0.204	0.180
Personalized PR	0.51	0.38	0.378	0.345
Hetero-Personalized PR	**0.53**	**0.41**	**0.402**	**0.355**

Table 3.3: Evaluation results

although the output of our algorithms can rank the entities in different types, we only evaluate the output ranking of authors in this experiment. The labeled data contains 10 queries. For each queries, the top authors are given as ground truth(*i.e.* relevant authors), as listed in Table 3.2. This benchmark dataset was used in evaluating expert finding problem [3]. In our experiment, we fix the damping factor to be 0.7.

The metrics we are using include Precision at rank n (P@n); R-precision (R-prec), defined as the precision at rank R where R is the number of relevant documents for the given query; MAP (Mean-Average Precision), defined as the average of the P@n values for all relevant documents.

3.5 Experiment Results and Case Studies

The evaluation results are given in Table 3.3

The results clearly show that Personalized PageRank-based methods are much better than the two baselines. While among the two baselines, TF-IDF + authority is better than TF-IDF alone. The Hetero-Personalized PageRank performs slightly better than standard Personalized PageRank. We argue that since we only have the labeled data for author, the improvement of Hetero-Personalized PageRank is not easily seen. However, we will do a case study. For the same query "graph pattern mining", the results of Personalized PageRank and Hetero-Personalized PageRank are given in Table 3.4. Though the author and term ranking are similar, we can see the Hetero-Personalized PageRank outputs a better ranking for conference. The personalized PageRank

Author	Conf	Term
Jiawei Han	IJCAI	graph
Philip S. Yu	KDD	pattern
Christos Faloutsos	AAAI	mine
Xifeng Yan	ICDE	data
Wei Wang	PAKDD	database
...	...	...

Author	Conf	Term
Jiawei Han	KDD	mine
Philip S. Yu	PAKDD	pattern
Christos Faloutsos	ICDM	graph
Xifeng Yan	ICDE	data
Jian Pei	IJCAI	database
...	...	...

Table 3.4: Results for query "graph pattern mining" (Hetero-Personalized PageRank on the right)

Author	Conf	Term
Serge Abiteboul	VLDB	xml
H. V. Jagadish	ICDE	query
Mounia Lalmas	SIGMOD Conference	data
Wenfei Fan	WWW	database
Divesh Srivastava	CIKM	document
...	...	...
H. V. Jagadish	SIGMOD Conference	xml
Michael J. Carey	VLDB	query
Divesh Srivastava	ICDE	data
Serge Abiteboul	CIKM	database
Jeffrey F. Naughton	WWW	system
...	...	...
W. Bruce Croft	SIGIR	xml
Mounia Lalmas	VLDB	query
H. V. Jagadish	ICDE	retrieval
Serge Abiteboul	SIGMOD Conference	data
Norbert Fuhr	CIKM	information
...	...	...

Table 3.5: Results for queries "xml", "xml"+"SIGMOD Conference"and "xml"+"SIGIR"

outputs "IJCAI" as the top one. This is because "IJCAI" has more papers and has longer history, and the personalized PageRank was biased, while the Hetero-Personalized PageRank is not easy to be biased.

Finally, we show the power of the combination of entity query using one example. The results of pure keyword query "xml", "xml"+ conference entity "SIGMOD Conference" and "xml" + conference entity "SIGIR" are shown in Table 3.5, respectively. Interestingly, the later two rankings give two different aspects of "xml" query.

Currently in iVis, we use Hetero-personalized PageRank to computer the ranking. The algorithm's time complexity is O(|E|) and it converges very fast, thus the computation time is minimal. Notice the DBLP subset we are using contains 270,959 edges.

Chapter 4

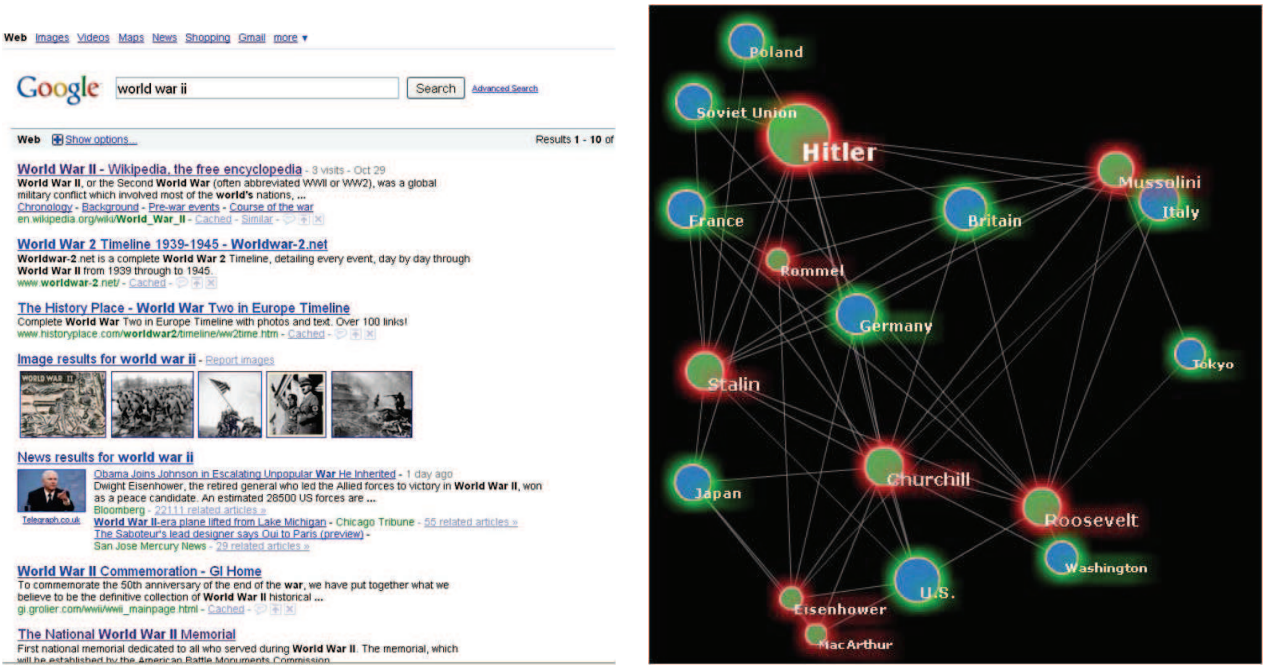
Extension to the Web

The generalization of our system is to integrate any information from the infinite source - the World Wide Web into an information network. Suppose one wants to search for articles of a specific event, we can treat terms of different types as entities, such as people's names and locations and the results can be presented as a graph composed of different kinds of entities, links as their relations. Such graph contains information in a more compact way and would be more intuitive and meaningful to show the correlations of the key entities. As shown in Figure 4.1, we can see the results of query "World War II" in Google and in iVis. The Google results are merely top-k blue links. Suppose the user previously knew nothing about the event "World War II", the Google user will need to read the retrieved webpages respectively, to see who are the important people and what are their relationships. While our system can provide the top-k entities and those top-k entities are organized as a whole into an integrated graphical view. This view is comprehensive and knowledgeable, with user-friendly interactions to enable further explorations. The relationships between the entities can also be summarized. This view will give a general summary of the query, and the user no longer need to look into the documents, to some extent. We do not assert that such view can replace the human reading, but this will definitely serve as a good complement and a vivid summarization. Another example of "Election 2008" is shown in Figure 4.2.

For a given corpus of document collection, we need to first extract the entities using some Name Entity Recognition tool (Stanford NER [1] is currently used) and index them. Two entities co-exist in a lot of sentences are linked or have higher weight on their edge. Thus we build an heterogeneous information network again. After that, the rest part of the system, including the ranking algorithm is essentially the same as what was done on DBLP data.

If we want to search the whole web, an alternative way is to send the query to search engine

[1] http://nlp.stanford.edu/software/CRF-NER.shtml



(a) search results in Google (b) in iVis

Figure 4.1: World War II search results comparison.

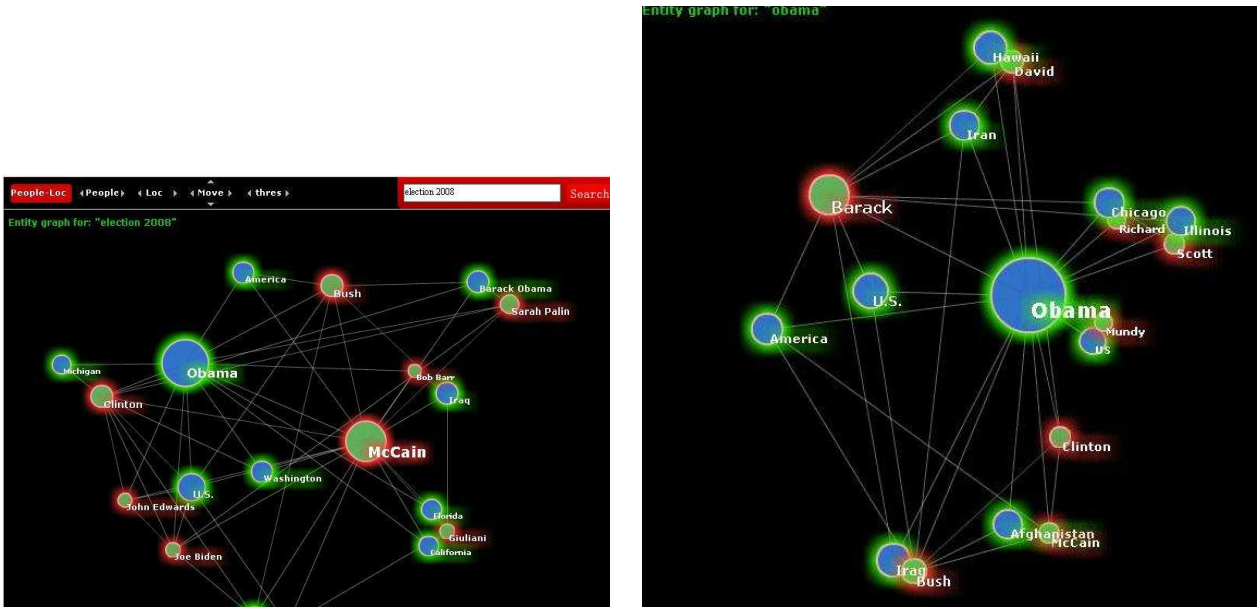


Figure 4.2: Query results of "Election 2008" and results after clicking entity "Obama"

like Google and retrieve the top 100 documents. Then for the 100 retrieved documents, we extract the entities and using some simple rules to merge the same entities (e.g. Bill Clinton and William Clinton) and thus build the heterogeneous information network online. Then similarly, we retrieve the top-k entities and render visualization. Notice extracting entities online may take up to several minutes. Interestingly, we can use different sources such as Google Web Search, Google News, Twitter or other collection of documents for different needs. We can even further utilize the time stamps contained in the web pages to capture the temporal trends. In Figure 4.1, we can see the Allies leaders Churchill and Roosevelt and also Allies generals such as Eisenhower, McArthur are close to each other. One can imagine that if we have the time stamps of each articles and generate a series of the graphical summaries along the time line, before US joined the WWII and formed the Allies with England, the distance between Roosevelt and Churchill is not that close.

Chapter 5

Discussion and Related Work

In this chapter, we discuss the research problems, related work as well as future work about iVis, explain their motivations and challenges, and suggest possible approaches to solve them. We first summarize the research issue in the project in a high level of view, then discuss them in each direction, respectively.

5.1 Overview

In iVis project, we design and implement a system to support multi-dimensional search and visualization on a specialized information network: DBLP. The system aims to facilitate people, especially computer science researchers, toward a better understanding and user experience about academic information networks.

There are many research problems related to our system. More specifically, they are related to the two objectives of our system: better understanding of the information network, and better user experience. We group the potential problems from the view of information network analysis, information retrieval and user interaction.

The functions of our system are centered in searching and exploring of information network data. How to make best use of the data is an important problem. DBLP is a well-known scientific publication repository which collects recent years' journal and conference proceedings in computer science domain. Even the power of the dataset itself has not been fully exploited. Speaking of the domain, DBLP is an academic collaboration network. With regards of the network property, it is a heterogeneous network, as well as a dynamic evolutionary network. So related research in the direction of information networking data mining can be discussed in three levels. First, focusing on DBLP data, we can develop more powerful techniques to summarize and extract knowledge from

it. Second, generalizing our methods or using our results, what kind of knowledge we can learn from the academic information network is to be explored. Finally, even further generalization on heterogeneous, dynamic and large-scale network will drive us to a thorough study of many more problems.

As a search system, there are always research problems from the view of information retrieval. The central problem for our system is: what shall we do when the information retrieved is not only linked objects, but also *how* they are linked, and even how they are clustered and ranked thanks to the links? The search could be either triggered by keywords of entities or links, or be interactively navigated on the network. The results returned by the search, however, are different from traditional search system. For the first thing, the link information itself is very important to be part of the results. For another thing, how to integrate those linked objects in a meaningful way such as in a clustered and ranked structure is a problem. Moreover, it poses new challenges on how to define the relevance and utility since the form of the retrieved information is more complicated than documents or links. In the following, we will discuss the research problems in these aspects. Either of them may contain several subtopics.

5.2 Information Network Analysis

Nowadays, people and entities are connected in all kinds of information networks. With the rapid growth of online networking applications, such as Facebook, Twitter and MySpace, it is recognized that there is abundant information contained in either online or real-world networks. For example, on Facebook users share their ideas and experiences with friends; on Twitter.com, digg.com, etc. people follow others' news according to their interest and concern. Thus the information network is a good indication how information, ideas and influence are spread away. In our system iVis we try to integrate in-depth analysis of information network and present it in an explicit way. To provide good search service on information network, we need to first achieve a good comprehension of the data. Then we can convert the data to information and information to knowledge. How deeply we understand the data decides the quality of the service we can provide.

In this section we will discuss the following issues: (1) mining knowledge by analyzing links; (2) mining knowledge about links; (3) combining text and links in information network analysis;

(4) entity identification, information extraction and data integration.

5.2.1 Mining knowledge by analyzing links

In iVis system, a fundamental function is to rank authors, conferences according to their relevance with certain topic and show a heterogeneous clusters of them. Ranking and clustering are both knowledge about the entities in the network. We can mine this kind of knowledge by analyzing the links between these entities.

In information network analysis, two most important ranking algorithms are PageRank and HITS, both of which are successfully applied to the Internet search. Both PageRank [1] and HITS [5] are evaluating the static quality of objects in information network. However, both PageRank and HITS are designed on the network of web pages, which is a directed homogeneous network. In a network containing multiple types of nodes, such as authors and conferences in DBLP network, we need to develop new method for ranking.

Clustering is another way to summarize information network and discover the underlying structures, which partitions the objects of an information network into subsets (clusters) so that objects in each subset share some common traits. By link analysis we can define the similarity of objects according to their structural feature, link in SimRank [4] and LinkClus [20]. They require a lot of computation of the pair-wise similarity.

An interesting research problem is to integrate clustering and ranking in one framework. This is motivated to generate more accurate clusters and more informative view of ranking. For example, given the DBLP data with conferences and authors in many different areas, if we rank them together, the results will be dumb and biased. Besides, the clusters should be relevant to the specific ranking users are interested in. Since different queries will generate different rankings, the clusters should also reflect this difference. So clustering and ranking are actually dependent on each other in many applications. The problem is how to integrate them smoothly.

In web search, there is an idea of facet ranking [22], which clusters the returned results for each query into different categories, to help users to better retrieve the relevant documents. That is a different concept with our problem. Facet ranking cluster sources according to their types. Take academic search for example, if the returned results include documents about papers, conferences

and persons, they are grouped into these three groups. However, in our problem definition, we already know the type of each object, but we aim to cluster objects that are conceptually similar, e.g., authors with similar interests, or conferences with similar topics.

RankClus [17] and NetClus [18] are some initial studies in this direction. Instead of combining ranking and clustering in a two stage procedure like facet ranking, the quality of clustering and ranking can be mutually enhanced in RankClus. The problem with both of them is that the ranking is irrelevant with query, and so is the clustering. We discussed our solution to query-dependant ranking in Chapter 3. Actually we also tried support query-dependant ranking using SimRank [4] by combining the similarity scores of the query entities together. But this method turns out to be not effective. To support query-relevant clustering (and possibility also ranking at the same time), we can use a two-stage methodology. In the first stage we collect relevant objects according to the relevance to the query in a traditional way. In the second stage we retrieve the ranking and clustering results pre-computed for these objects. Apparently this methodology is not an elegant solution. It only ensures the objects retrieved are relevant to the given query, but does not change the importance score and topological relation of them according to the query. The retrieved objects may be in isolated clusters in the original clustering, but should be clustered together according to their relevance in given topics. The challenge is how to adjust the criteria of clustering automatically with respect to user query. One possible method is to use the relevance score instead or in complement with a general measurement like the number of links. It requires study to see whether the clustering and ranking are still mutually strengthened in this case.

There are other research problems in this direction, including how to extend the NetClus to more general heterogeneous network, how to deal with dynamic networks, and how to extract multiple concept hierarchies automatically. By more advanced link analysis techniques, we could discover knowledge buried in depth and provide more powerful search services.

5.2.2 Mining knowledge about links

With the help of link mining technique, people are able to extract knowledge such as community structure, and authoritative sources can be discovered from the network data in which links play an important role. However, existing work mainly focus on mining knowledge based on links other

than mining knowledge tied in the links themselves. To be more specific, links can be refined, classified and distinguished though in the data they are not. For instance, in a forum where replied messages can be seen as links connecting different users within the same discussion board, these links can have very different meanings: some are supportive while some are opposed. To differentiate them help us better understand the network structure. Once the semantic meaning of the links is extracted from the mingled data, information network analysis will be facilitated in at least three ways. First, the network can be finely modeled because additional information is available other than plain links. Second, hierarchies, clusters and components discovered by different means can be compared to see if they are meaningful. Last, it enriches graph summary and influence analysis.

During the development of our system, we made an attempt to infer the advisor-advisee relationship between coauthors from collaboration network. Collaboration network is a graph composing researchers as nodes, and their collaboration as edges. From the view of knowledge discovery, people are interested in how researchers are connected to each other and how the research community is formed by each individual researcher. As a first step, identifying advisor-advisee relationship can help us answer these questions. The community evolution is motivated by the development of each individual. If we can figure out how each researcher grows from an advisee to a self-governed researcher or even an advisor, not only we can position each person in a chronological axis in a correct order, but also we can sketch the whole community in a very clear view. We can further do clustering, influence analysis and research topic evolution, etc.

Many projects have been set up to maintain such information for various research fields. These include the Mathematics Genealogy Project [2], the Computer Engineering Academic Genealogy, the AI Genealogy Project and the Software Engineering Academic Genealogy. However, all of these projects rely on manually collecting the academic genealogy data which makes them quite costly. For a given collaboration network, it is not always the case we can easily find corresponding dissertation data. Either lack of data or difficulty to identify the mapping will lead to failure. Therefore, we need to develop a general analyzing technique in order to automatically mine the relations from the network data.

Using graph mining approach, nodes or edges with certain properties such as having the largest

centrality or betweenness can be discovered. We can also compute importance of a node, and relevance of neighboring nodes, e.g. using PageRank [1]. Furthermore, we can do ranking and clustering based on the link information, e.g. using NetClus [18]. However, with all the existing method, it is still difficult to differentiate the social role from the static collaboration network. We must consider the temporal information and build a unified model for the dynamic collaboration network. The methodology is expected to apply for general dynamic network.

The problem of relationship mining is quite different from existing works on information network analysis and poses a set of unique challenges.

- *Latent relation.* The advisor-advisee relation is completely hidden in the collaboration data. There is no explicit sign who is one's advisor among numerous collaborators.
- *Time-dependent.* Social role like advisor or advisee is highly time-dependent. One could turn from an advisee to an advisor but there is no clear sign when this transition happens.
- *Scalability.* To find one's advisor it is not sufficient to only consider the information of his coauthors. The network as a whole is correlated and the search space is exponential in size. It is important to develop a method that can scale well to real large data sets.

We can formulate the problem of advising relationship mining as a probabilistic ranking problem, and use a time-constrained probabilistic factor graph model to model the dynamic collaboration network. Specifically, the advisor of each author and the advising time of all of them can be modeled together as a joint probability of as many hidden variables as authors with time constraint. An efficient algorithm to optimize the joint probability and obtain ranking score can be designed as a process of message propagation on the network.

5.2.3 Combining text and links in information network analysis

Information networks often contain abundant text, in the form of electronic documents, reports, e-mails, conversations, news, webpages, and other narratives. Text data forms a critical component of the information network. In iVis project, we also have text information rather than pure links between objects. Text data form a critical component of the information network. It is necessary to study text in order to extract semantic meaning of user interests, opinions and relations. On

the other hand, traditional information retrieval method based on text could be enhanced by link analysis. So combining the two will have benefits for both.

Social influence is an interesting problem that is recently being studied. In our system, it will be a good feature if we can show the quantitative influence from each entity to another in our results, rather than merely show the interaction between each pair of nodes. [19] presents a method for measure the influential strength. It formalizes the problem of social influence analysis as identifying which user has the highest probability to influence another user in the social network. However, the key question is still open, i.e., how to measure arbitrary pair-wise influence? To conduct social influence analysis on heterogeneous networks, which contain different types of nodes and sufficient text and link information as well, it will be a good testbed to combine text and link information in a unified model. We have following challenges.

- *Pairwise.* Social influences measure the extent to which a user is likely to be affected by decisions of their friends and colleagues. Thus it is important to model the pairwise influence in a principled method.
- *Time-dependent.* Social influences are highly time-dependent. The influence of a user on another (strongly) depends on their interactions in previous time windows.
- *Scalability.* Real social networks are getting bigger with thousands or millions of nodes. It is important to develop the method that can scale well to real large data sets.

To solve the problem we can try to combine topic modeling and network analysis, and leverages the power of both statistical topic models and discrete regularization. We can extend topic models by considering additional properties of information networks. We can also apply text mining on dynamic networks and consider the evolution of topics. By doing so, we expect to discover the force of influence that drives the evolution, as well as to discover the hidden connections between nodes, cluster nodes with different types, discover semantic information network community, and summarize the information network at high and meaningful levels.

5.2.4 Entity identification, information extraction and data integration

This problem is related to the data source preprocessing, cleaning and integration. In practice we find that to have consistent data is an important factor for either search or knowledge discovery. Data can be noisy, incorrect, or misleading. Even in DBLP, name ambiguity is a major problem which sometimes prevents effective discovery of the interest and relation of authors. Another problem is one person with multiple names in the database. To solve the former we need entity identification and to solve the latter we need data integration. Information extraction is needed when we want to limit the storage and the scope of the function.

The challenge is that in a large, diverse, and interconnected system, it is difficult to assure accuracy or even coherence among the data sources. Indeed, there are likely to be inconsistencies within many of the individual data resources themselves. It requires us to study both how to learn good models from different sources with different kinds of associated uncertainty, and how to make use of these, along with their level of uncertainty in supporting coherent decisions, taking into account characteristics of the data as well as of its source. The integration of information coming from different sources requires combining different types of statistical models with available high-level knowledge. It also requires us to consider how to integrate prior information provided by human experts into different information sources to obtain good quality of them.

5.3 Information Retrieval

In this section we address three problems related to the specialty of iVis as a information retrieval system: the measurement of the relevance of retrieved information about a information networks; the performance issue when the results are integrated graph rather than independent documents or objects; extension from closed network data retrieval to web search.

5.3.1 Measurement of relevance

In our system we don't return ranked documents one by one, but visualize the documents in a integrated way. Objects are linked and presented as a graph. Such a graph contains more information than traditional ranking. The similarity between objects can be measured by their

distance in the graph; the relation of them are indicated by the edges on the graph. So the visualized graph as a whole convey more information than isolated objects. In this case how to measure the relevance of the results, how to compare the utility become a new problem. The criteria is not only which objects are retrieved, but also the way they are presented. Traditional measures like precision and recall cannot be directly applied and the document-by-document evaluation assumption does not hold. Maybe it is time to rethink the query-by-query measurement [16] and develop new evaluation standard.

The measurement of relevance, or accuracy, is a foundation for many other problems in the information retrieval scope. To solve it we need solid study of assumptions on users' expectation when searching complicated and connected components in information networks. One of the study we can do is that similarity-based network and link-based network can be compared to see which, or in what scenario each, is more expected by users.

5.3.2 Performance

As the computation involves more than document ranking, the performance issue raises new problem. As we have mentioned in section 5.2.1, in our system we need to solve the problem of ranking according to the relevance w.r.t. query and the similarity between retrieved objects as well as the clustering in an integrated way. If we recompute the rank and clusters according to the query every time, the computation must be very scalable. Even in that case indexing technique is still required. Online computation is only tolerable if it is restricted in a reasonable small subset of data; or it could be done in a parallel way. We can use index to fetch a relevant subset of data, and redo the computation of them to generate query-specific ranking and clustering. Finally we can further refine the result presented according to the new ranking. The tradeoff between the performance and the accuracy can be studied. And that actually requires the solution to the problem proposed above: the measurement of relevance.

Chapter 6

Conclusion

In iVis project, we design and implement a system to support search and visualization on heterogeneous information networks. We first build our system on a specialized heterogeneous information network: DBLP. The system aims to facilitate people, especially computer science researchers, toward a better understanding and user experience about academic information networks. Then we extend our system to the Web. Our results are organized into a graphical view, which are much more intuitive and attractive than the simple top-k blue links from traditional search engine , and bring more meaningful structural results with correlated entities. We also investigate the ranking algorithm, and we show that the personalized PageRank and our proposed Hetero-personalized PageRank outperform the TF-IDF ranking or mixture of TF-IDF and authority ranking. Our work opens several directions for future research as discussed.

References

[1] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117, 1998.

[2] Harry B. Coonce. Computer science and the mathematics genealogy project. *SIGACT News*, 35(4):117–117, 2004.

[3] Hongbo Deng, Irwin King, and Michael R. Lyu. Formal models for expert finding on dblp bibliography data. In *Proc. 2008 Int. Conf. on Data Mining (ICDM'08)*, 2008.

[4] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *Proc. 2002 ACM SIGKDD Int. Conf. Knowledge Discovery in Databases (KDD'02)*, pages 538–543, New York, NY, USA, 2002. ACM.

[5] Jon Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[6] Jon Kleinberg. The convergence of social and technological networks. *Commun. ACM*, 51(11):66–72, 2008.

[7] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, D. Sivakumar, Andrew Tompkins, and Eli Upfal. The web as a graph. In *Proc. 2000 ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems (PODS'00)*, pages 1–10, 2000.

[8] Jure Leskovec. *Dynamics of Large Networks*. PhD thesis, Carnegie Mellon University, 2008.

[9] Michael Ley. The dblp computer science bibliography: Evolution, research issues, perspectives. In *SPIRE*, pages 1–10, 2002.

[10] Joseph Brendan Manning. *Geometric symmetry in graphs*. PhD thesis, West Lafayette, IN, USA, 1991.

[11] M. E. J. Newman. Assortative mixing in networks. *Physical Review Letters*, 89(20):208701, October 2002.

[12] M. E. J. Newman. Analysis of weighted networks. *Physical Review E*, 2004.

[13] Andreas Noack. Energy models for graph clustering. *J. Graph Algorithms Appl.*, 11(2):453–480, 2007.

[14] Andreas Noack. Modularity clustering is force-directed layout. *CoRR*, abs/0807.4052, 2008.

[15] Sriram Raghavan and Hector Garcia-Molina. Representing web graphs. page 405, 2003.

[16] S. E. Robertson. The probability ranking principle in ir. pages 281–286, 1997.

[17] Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhijun Yin, Hong Cheng, and Tianyi Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. In *Proc. 2009 Int. Conf. on Extending Database Technology (EDBT'09)*, pages 565–576, 2009.

[18] Yizhou Sun, Yintao Yu, and Jiawei Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proc. 2009 ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD'09)*, pages 797–806, 2009.

[19] Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social influence analysis in large-scale networks. In *Proc. 2009 ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD'09)*.

[20] Xiaoxin Yin, Jiawei Han, and Philip S. Yu. Linkclus: efficient clustering via heterogeneous semantic links. In *Proc. 2006 Int. Conf. Very Large Data Bases (VLDB'06)*, pages 427–438. VLDB Endowment, 2006.

[21] Yintao Yu, Cindy X. Lin, Yizhou Sun, Chen Chen, Jiawei Han, Binbin Liao, Tianyi Wu, ChengXiang Zhai, Duo Zhang, and Bo Zhao. inextcube: information network-enhanced text cube. In *Proc. 2009 Int. Conf. on Very Large Data Bases (VLDB'09)*, volume 2, pages 1622–1625. VLDB Endowment, August 2009.

[22] Oren Zamir and Oren Etzioni. Grouper: A dynamic clustering interface to web search results. pages 1361–1374, 1999.