Semantic Relationship Discovery with Wikipedia Structure

Fan Bu, Yu Hao and **Xiaoyan Zhu**
State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Sci. and Tech., Tsinghua University
buf08@mails.tsinghua.edu.cn
haoyu@mail.tsinghua.edu.cn
zxy-dcs@tsinghua.edu.cn

Abstract

Thanks to the idea of social collaboration, Wikipedia has accumulated vast amount of semi-structured knowledge in which the link structure reflects human's cognition on semantic relationship to some extent. In this paper, we proposed a novel method RCRank to jointly compute concept-concept relatedness and concept-category relatedness base on the assumption that information carried in concept-concept links and concept-category links can mutually reinforce each other. Different from previous work, RCRank can not only find semantically related concepts but also interpret their relations by categories. Experimental results on concept recommendation and relation interpretation show that our method substantially outperforms classical methods.

1 Introduction

Discovering semantic relationship between concepts is easily handled by humans but remains a obstacle for computers. During recent years, Wikipedia, the world's largest collaborative encyclopedia, has accumulated vast amount of semi-structured knowledge (e.g. 17 million concepts in total and 3 million in English), which to some extent reflects human's cognition on relationship.

Many prior researches on semantic computation with Wikipedia structure can only compute the tightness of the relationship between two concepts but not give which kind of relationship it is [Ollivier and Senellart, 2007; Adafre and de Rijke, 2005; Milne, 2007; Hu *et al.*, 2009]. This is partly due to the fact that most of these work are originated from information retrieval in which the articles and links on wikipedia are analogous to the pages and links on the web, which do not seize the specialty of Wikipedia stucture.

In this paper, we proposed concept-category graph to model Wikipedia structure in which concepts and categories are treated as different nodes. We assume that the links between concept and category (category links) and the links between concepts (related links) present two senses of relatedness. A illustration is shown in Fig. 1. Concepts can be related in two different ways: linking to same categories or linking from each other by anchor texts. Further, the category itself can be used to interpret the relationship between the concepts linking to it because it naturally represent the general character of these concepts.

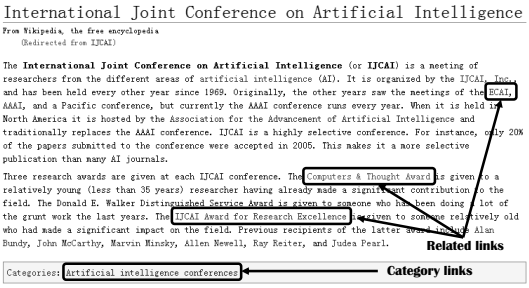

Figure 1: Illustration of related links and category links in Wikipedia.

Inspired by the *mutually reinforcing nature* of HITS [Kleinberg, 1999], an new algorithm RCRank (*joint ranking of related concepts and categories*) is proposed to jointly compute concept-concept relatedness and concept-category relatedness base on the observation that information carried in related links and category links can reinforce each other. By RCRank, we can get top-n most relevant concepts and categories for each concept on Wikipedia. For each pair of concepts, our method can return a list of categories which best interpret the relationship between them even if they do not share common categories. Experimental results on concept recommendation and relation interpretation show that our method substantially outperforms classical methods.

The contributions of this paper are twofold. First, we are the first to discover the semantic relationship between concepts with Wikipedia, which is different from previous work on relatedness computation. Second, we present RCRank, a simple link-analysis algorithm which seize the semantic characteristics of Wikipedia stucture. It is also applicable to applications where two relations can reinforce each other.

2 Our methods

2.1 Intuition

Category links and related links present two senses of relatedness in Wikipedia. Usually, people link a concept to a cat-

egory because they think the category is more general than the concept in some respects. Others may find more related concepts if they navigate through this link and view other concepts under the same category.

Related links are created from the content of articles in the form of anchor texts. If a user find a name entity in the article referring to another article, he can turn the name entity into an anchor text linking to corresponding article. Each article is a concise description of the corresponding concept. So if one concept is linked to from the article of another, they are likely to be related.

Now the question is: can we measure concept-concept and concept-category relatedness based on these two kinds of links?

To answer this question, we propose the following two hypotheses which characterize how concept-concept and concept-category relatedness mutually reinforce each other.

- H1: *If a category is shared by many related concepts of a concept, it is likely to be related to this concept.*
- H2: *If two concepts share many related categories, they are likely to be related.*

In the next two sections, we will formalize these heuristics by concept-category graph and RCRank.

2.2 Concept-Category Graph

In Wikipedia, we denote the set of concepts and categories by U and V. Cardinality of U and V are denoted by m and n. The concept-category graph is defined as follows.

Definition 1 *A **concept-category graph** is defined as a 4-tuple $G = (U, V, E_C, E_R)$ in which $E_C \subseteq U \times V$ and $E_R \subseteq U \times U$. $\langle u_i, v_j \rangle \in E_C$ iff. v_j is linked to from u_i and $\langle u_i, u_j \rangle \in E_R$ iff. u_j is linked to from u_i .*

We define $m \times n$ matrix $\mathbf{C} = [c_{ij}]$ in which c_{ij} is the relatedness between concept u_i and category v_j and $m \times m$ matrix $\mathbf{R} = [r_{ij}]$ in which r_{ij} is the relatedness between concept u_i and concept u_j. Also, we require $\mathbf{C}$ and $\mathbf{R}$ to subject to the following properties.

- For each u_i, we have $c_{ij} \geq 0$ for each v_j and there exist some $v_{j'}$ satisfying $c_{ij'} > 0$. (*non-negative*)
- For each u_i and u_j, we have $0 \leq r_{ij} \leq 1$ and $r_{ii} = 1$. (*similarity metric*)
- For each u_i and u_j, we have $r_{ij} = r_{ji}$. (*symmetry*)

The properties of $\mathbf{R}$ are natural for a relatedness metric. The *non-negative* requirement on $\mathbf{C}$ is to ensure the similarity metric property of $\mathbf{R}$ after iterations in RCRank.

2.3 RCRank

To mutually reinforce $\mathbf{R}$ and $\mathbf{C}$, we express H1 and H2 by the following equations.

$$c'_{ij} = \sum_k r_{ik} c_{kj} \tag{1}$$

$$r'_{ij} = \frac{\sum_k c_{ik} c_{jk}}{\sqrt{\sum_k c_{ik}^2} \sqrt{\sum_k c_{jk}^2}} \tag{2}$$

Eq. 1 is a "voting" process. For a given concept u_i, the new score of category u_j is voted by all the related concepts of u_i. The more related a concept is, the more important its votes are.

In eq. 2, the concepts are projected to n-dimensional category space. The *i*th row of $\mathbf{C}$ is the image of concept u_i on category space and r'_{ij} is actually the cosine similarity between u_i and u_j on category space.

Both eq. 1 and eq. 2 can be written into matrix form as

$$\mathbf{C}' = \mathbf{RC} \tag{3}$$

$$\mathbf{R}' = \mathbf{DCC}^T\mathbf{D}^T \tag{4}$$

in which matrix $\mathbf{M}^T$ is transposition of $\mathbf{M}$ and

$$\mathbf{D} = diag(\mathbf{CC}^T)^{-\frac{1}{2}}$$

in which $diag(.)$ results in a diagonal matrix in which all elements are zero except the diagonal elements whose values are from the input matrix of the same positions. $diag(\mathbf{CC}^T)$ is invertible because of the non-negative property of $\mathbf{C}$.

Given a concept-category graph G, RCRank works as follows:

1. Initialize $\mathbf{R}(0)$ and $\mathbf{C}(0)$ from the concept-category graph G.
2. For each time *t*, compute
   a. $\mathbf{C}(t) = \mathbf{R}(t-1)\mathbf{C}(t-1)$
   b. $\mathbf{D}(t) = diag(\mathbf{C}(t)\mathbf{C}(t)^T)^{-\frac{1}{2}}$
   c. $\mathbf{R}(t) = \mathbf{D}(t)\mathbf{C}(t)\mathbf{C}(t)^T\mathbf{D}(t)^T$

Before initializing $\mathbf{R}(0)$ and $\mathbf{C}(0)$ from G, we first define $\mathbf{M}_C$ and $\mathbf{M}_R$ in which the entries are 1 if the corresponding edges belong to E_C and E_R respectively, or else 0.

To satisfy the non-negative property of C, we create a dummy category for each concept without categories. Formally, denote n_d as the number of concepts without categories. we define

$$\mathbf{C}(0) = \left[\mathbf{M}_C, \mathbf{M}_C^d\right] \tag{5}$$

in which $\mathbf{M}_C^d$ is a $m \times n_d$ matrix defined as

$$(\mathbf{M}_C^d)_{ij} = \begin{cases} 1, & \text{if } u_i \text{ is the } j\text{th concept without categories} \\ 0, & \text{or else} \end{cases}$$

To satisfy the similarity metric and symmetry properties of R, we define $\mathbf{R}(0)$ as

$$\mathbf{R}(0)_{ij} = \max\left(\mathbf{I}_{ij}, \mu_G\left(\frac{(\mathbf{M}_R)_{ij} + (\mathbf{M}_R)_{ji}}{2}\right)\right) \tag{6}$$

, in which $\mathbf{I}$ is $m \times m$ identity matrix and $\mu_G \in [0, 1]$ is a parameter depends on graph G.

Intuitively, μ_G weighs the influences between category links and related links. Consider the following two extreme cases:

- If $\mu_G = 0$, the information carried in related links is ignored and the relatedness between two concepts is solely judged by how many common categories they share.

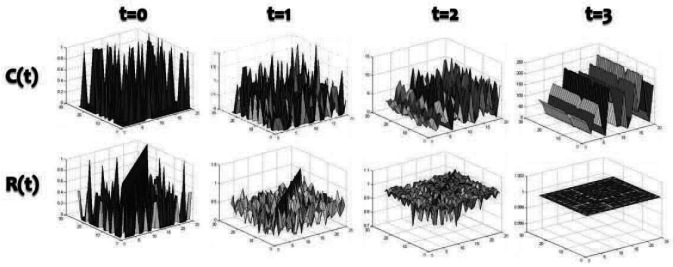

Figure 2: The converging process of **R** and **C** on a graph with 25 concepts and 20 categories. **R**(0) and **C**(0) are initialized with 90% of entries being zero.

- If $\mu_G = 1$, the categories links of a concept are submerged by the categories links of its related concepts. The relatedness between two concepts is dominated by the related concepts they share.

To balance the influences between category links of a concept and related links, for each concept i, we define

$$\mu_G(i) = \frac{\sum_j \mathbf{C}(0)_{ij}}{\Sigma_{k,j}(\mathbf{M}_R + \mathbf{I})_{ik}\mathbf{C}(0)_{kj}} \tag{7}$$

μ_G is defined as the geometric mean of $\mu_G(i)$.

$$\mu_G = \sqrt[m]{\prod_{u_i} \mu_G(i)} \tag{8}$$

From the definition we can see that μ_G is between 0 and 1 and it will decrease while E_R getting denser and increase while E_R getting sparser.

During the iteration process of RCRank, it is easy to prove that if $\mathbf{R}(0)$ and $\mathbf{C}(0)$ satisfy the basic properties, so will $\mathbf{R}(n)$ and $\mathbf{C}(n)$. Will this iteration converge and what is the stop criterion?

The convergence analysis of RCRank is given in appendix A. Intuitively, if $\mathbf{R}(0)$ and $\mathbf{D}(0)\mathbf{C}(0)$ are regarded as human annotated data, $\mathbf{R}(\infty)$ and $\mathbf{D}(\infty)\mathbf{C}(\infty)$ are considered as a priori values, the iteration of RCRank is actually a process of smoothing. The time t balance the preference between posteriority and priority.

RCRank converges exponentially fast. In our experiments on small concept-category graphs in which |U| and |V| are between 10 and 50, most of them converges on t = 3 or 4. From Fig. 2 it is easy to see that **R** and **C** will be over smoothed when they converges. In this paper, we compute **R**(1) and **C**(1) for empirical evaluations.

2.4 Semantic Interpreter

Inspired by Explicit Semantic Analysis [Gabrilovich and Markovitch, 2007], we build a *Semantic Interpreter*(SI) that maps each pair of concepts u_i and u_j into weighted vector of categories. SI(u_i, u_j) is a vector of categories in which the weight of each category v_k is defined as

$$w(v_k) = \frac{c_{ik}c_{jk}}{\sqrt{\sum_m c_{im}^2}\sqrt{\sum_n c_{jn}^2}}$$

. For simplicity, we write SI(u_i, u_i) as SI(u_i).

When we compute SI(u_i, u_j) with $\mathbf{C}(0)$, the weight of each category is determined by whether it is linked to from both u_i and u_j. When using $\mathbf{C}(1)$, SI(u_i, u_j) represents the contribution of each category on the semantic relatedness between u_i and u_j, in which the categories with non-zero weights are not necessarily linked from u_i and u_j.

3 Empirical Evaluation

3.1 Data Preparation

We implemented RCRank on a Wikipedia snapshot of August 22, 2009. After removing redirect pages, we got 2,997,315 concept pages and 539,527 category pages. Each concept linked to 3.262 categories and 19.914 concepts on average. To ensure the relatedness of related links, only mutual links were retained, which resulted in 3.698 concept links per concept.

A IA-64 server with 64 CPUs(1.3G) and 64 GB memory was used to compute sparse matrix multiplication. It took 30 hours to compute $\mathbf{C}(1)$ and $\mathbf{R}(1)$ on the whole data. For memory concern, we only retained top 50 largest entries for each row of $\mathbf{R}(1)$. For each concept u_i, we recommended concepts corresponding to the top-n largest entries in row i.

3.2 Experiment: Concept Recommendation

To evaluate the quality of concept recommendation, we carried out a blind evaluation of five methods on seven different concepts, which were: (i) *Donald Knuth*: a short biographical article with many categories. (ii) *Germany*: a very large article about a country with a few categories. (iii) *Dog*: a medium-sized introductory article about a common concept. (iv) *Hidden Markov Model*: a short technical article. (v) *Romance of Three Kingdoms*: a large narrative article about a Chinese literature. (vi) *World War II*: a very large article with many categories about a historical event. (vii) *Gmail*: a short article about a commercial product.

For each concept, we put the top 20 results returned by each method together and asked each evaluator to assign a recommendation score between 0 and 5 (5 being the best) to each result. Evaluators should consider both relatedness and helpfulness of each result. For example, *France* is semantically related to *Pierre de Fermat* since Fermat is Frenchman. But the information contained in *France* has little to do with *Pierre de Fermat*, so it is not helpful. The golden standard relatedness between each query and recommended concept was the average of marks from each evaluator. Cumulative Gain(CG) and Discounted Cumulative Gain(DCG), which are widely used in information retrieval, are used to evaluate the result list of each method on each concept. There has been a total of 5 evaluators and the average of pairwise Pearson correlation coefficient is 0.61.

Description of the Methods

We compared the following five methods.

COCATEGORY. A straight-forward method which evaluate the semantic relatedness between two concepts solely on the number of category links they have in common. It returns concepts which share most categories with the query concept.

		COCATEGORY	COCITATION	WLVM	TFIDF	RCRANK
Donald Knuth	CG	30.40	42.40	47.00	62.40	**68.00**
	DCG	14.31	21.50	23.08	26.61	**28.94**
Germany	CG	45.00	48.40	**88.20**	77.60	78.60
	DCG	19.22	19.96	**35.44**	31.62	31.88
Dog	CG	42.20	40.40	74.60	73.00	**80.40**
	DCG	20.02	17.96	28.88	27.35	**32.29**
Hidden Markov Model	CG	55.80	66.60	58.40	69.20	**73.60**
	DCG	25.32	27.46	25.74	28.65	**30.62**
Romance of the Three Kingdoms	CG	27.00	86.20	80.60	70.80	**87.20**
	DCG	13.37	34.03	33.10	28.57	**34.58**
World War II	CG	45.00	56.20	**78.00**	73.40	72.60
	DCG	19.73	23.45	**31.43**	29.00	29.93
Gmail	CG	52.20	45.40	47.20	66.40	**74.20**
	DCG	23.46	21.01	19.59	26.56	**31.51**
avg.	CG	42.51	55.09	67.71	70.40	**76.37**
	DCG	19.35	23.63	28.18	28.34	**31.39**

Table 1: CG and DCG of each method on each concept

COCITATION. In this method, two concepts are more related if their cocitation count, which is the number of concepts pointing to both of them, is larger. For a given concept *u*, **COCITATION** ranks all concepts by their cocitation counts with *u* and return the top-n concepts.

WLVM. Wikipedia Link Vector Model(WLVM) [Milne, 2007] recommends concepts by related links. Each concept is represented by a weight vector of related links. Similar to TF-IDF in information retrieval, the weight of each link $u_a \rightarrow u_b$ is:

$$w(u_a \rightarrow u_b) = |u_a \rightarrow u_b| \times log\left(\sum_{u \in \mathrm{U}} \frac{|\mathrm{U}|}{|u \rightarrow u_b|}\right)$$

, which is the link counts weighted by the probability of each link occurring. Given a concept *u*, **WLVM** ranks all concepts by their cosine similarity with *u* and return the top-n concepts.

TFIDF. This method represents each concept by a weight vector of related links just the same as **WLVM**. For a query concept, **TFIDF** returns the concepts corresponding to the related links with top-n largest weights.

RCRANK. For a query concept u_i, we recommend concepts corresponding to top-n largest entries in the *i*th row of **R**(1).

Performance of the Methods

The performance of each method is listed on Tab. 1, from which we can see that the overall performance of **RCRANK** is better the others on both CG and DCG. It is worth noticing that the overall CG of **TFIDF** is significantly better than the other baselines but the DCG of **TFIDF** is only slightly better than **WLVM**, which shows that anchor text itself is a good source for recommendation but **TFIDF** is not good enough to evaluate the relatedness for each anchor text. Compared with other methods, the performance of **RCRANK** is more stable on different kinds of concepts, especially on the concepts with short articles such as "Donald Knuth" and "Hidden Markov Model". By mapping related links into category space, **RCRANK** not only evaluates the relatedness of each linked concept but also finds more related concepts by generalizing the semantic meaning of related links on highly related dimensions in category space.

3.3 Experiment: Category Selection

In preliminary experiment on concept recommendation we found that in most cases, the category that reasonably interpret the relationship between query concept and recommended concept can be found from categories linking to from the recommended concept. Therefore in this experiment, given a query concept u_i and a recommended concept u_j, we selected a category from the category set of u_j which can best interpret the relationship between u_i and u_j.

The test set was constructed as follows. First, 20 query concepts were chosen for their diversity. For each concept, we randomly selected 25 concepts from the top 50 results returned by **RCRANK**. For each pair of concept, we asked four annotators to choose at least one category from the category set of u_j which can best interpret the relationship between u_i and u_j. The final score of each category was the total number of times it was chosen divided by the total number of annotators. In this task, we think precision is much more important than recall as far as user experience is concerned. So for each method, we only select one category for each pair of concept and evaluate its precision. The following four methods are compared in this experiment.

Random. A baseline method which randomly select one category from the category set of recommended concept u_j.

Frequency. This method select the category from the category set of recommended concept u_j which is most frequently appeared in Wikipedia.

SISingle. This method select the category from the category set of recommended concept u_j with largest weight in $\mathrm{SI}(u_j)$.

SI. This method select the category from the category set of recommended concept u_j with largest weight in $\mathrm{SI}(u_i, u_j)$.

The precision of each method is shown in Tab. 2. From the results we can see that **SI** significantly outperform the others. **SISingle** is inferior to **SI** because it does not use the information contained in query concept. The performance of **Fre-**

Random	Frequency	SISingle	SI
0.425	0.3955	0.5745	**0.659**

Table 2: Precision of each methods on category selection

Gmail			World War II		
TFIDF	WLVM	RCRANK	TFIDF	WLVM	RCRANK
1. Google 2. Hotmail 3. Gmail interface 4. April Fools' Day 5. Internet Message Access Protocol 6. Email spam 7. Post Office Protocol 8. Ajax (programming) 9. Megabyte 10. Gmail Mobile	1. Gmail 2. Question Manager 3. Stealth edit 4. Cyndi's List 5. Backrub 6. Google Code Jam 7. Search appliance 8. Google AJAX APIs 9. Ignite Logic 10. Duplicate content	1. Gmail 2. Gmail interface 3. History of Gmail 4. Gmail Mobile 5. Gears (software) 6. GMail Drive 7. Google Apps 8. PhpGmailDrive 9. Mailplane (software) 10. Yahoo! Mail	1. Strategic bombing during World War II 2. Second SinoJapanese War 3. Eastern Front (World War II) 4. Red Army 5. Operation Barbarossa 6. Japanese American internment 7. Victory in Europe Day 8. Japanese naval codes 9. Belgrade Offensive 10. Battle of Britain	1. World War II 2. European Theatre of World War II 3. List of military engagements of World War II 4. Axis powers 5. Stalin in World War II 6. Commanders of World War II 7. Participants in World War II 8. Eastern Front (World War II) 9. Soviet occupations 10. NaziSoviet economic relations	1. World War II 2. Allies of World War II 3. Axis powers 4. North African Campaign 5. Operation Sonnenblume 6. Salients, reentrants and pockets 7. Battle of Alam el Halfa 8. Operation Brevity 9. Winter Line 10. Eastern Front (World War II)

Table 3: Top 10 results from **TFIDF**, **WLVM** and **RCRANK** on "Gmail" and "World War II"

quency is worse than **Random**, which shows that frequently appeared categories are too board to represent the relationship. For example, "Living people" is selected by **Frequency** to represent the relationship between "Steven Spielberg" and "Martin Scorcese", nevertheless "American film directors" is more preferable.

3.4 Case Study

Tab. 3 shows top 10 results from **TFIDF**, **WLVM** and **RCRANK** on "Gmail" and "World War II". The results on "World War II" show that no method is significant better than the others because "World War II" is a huge article with a large number of highly related concepts. The results on "Gmail" show that the concepts recommended by **RCRANK** is focused on different aspects of "Gmail", such as "Gmail interface" and "History of Gmail". It helps user to find more relevant information if he is interested in "Gmail". By contrast, the results from **TFIDF** and **WLVM** are less relevant.

Fig. 3 shows the results by our method on the concept "Germany" in which top 12 recommended concept are shown. For each recommend concept u, we calculate SI(u, "$Germany$") and select the categories from the category set of u of which the weights are more than eighty percent of the largest weight. From the results we can see that most categories are reasonable to represent the relationship between concepts. Moreover, although many concepts in Wikipedia can be linked to dozens of categories, most of them are not suitable to interpret the reason for recommendation. Our methods can pick out the ones which are both popular and related to input concept. For example, apart from "German physicists", the concept "Max Born" is also linked to categories such as "Theoretical physicists", "Nobel laureates in Physics" and "German Lutherans". The first two categories do not give the relation between "Germany" and "Max Born" directly. The last one is related to "Germany" but is not as well-known as "German physicists".

4 Related Work

In the area of finding related concepts on Wikipedia, Adafre and de Rijke [2005] identified missing related links using a cocitation approach. Ollivier and Senellart [2007] recommended related concepts by Green method, which was a classical Markov chain tool. In [Hu *et al.*, 2009], a random walk algorithm was proposed to propagate relatedness from seed concepts to unlabeled concepts. To our knowledge, none of these methods can give semantic relationship between query concept and recommended ones.

Another related area of our work is computing semantic relatedness using Wikipedia. WikiRelate! [Strube and Ponzetto, 2006] is the first approach one this area. Given a pair of words w_1 and w_2, WikiRelate! first maps them to Wikipedia titles p_1 and p_2 and then compute semantic relatedness using various traditional methods which rely on either the content of articles or path distances in the category hierarchy of Wikipedia. Different from WikiRelate!, the Wikipedia Link Vector Model(WLVM) [Milne, 2007] represents each article by a weighted vector of anchor texts and the weights are given by a measure similar to *tf-idf*. ESA [Gabrilovich and Markovitch, 2007] is another semantic relatedness measure which achieve good results in correlation with human judgments. ESA represents each text as a weighted vector of Wikipedia-based concepts and assess the relatedness on concept space using conventional metrics.

In collaborate filtering, Breese *et al.*[1998] proposed a memory-based algorithm for predicting user's rating, which looks similar to RCRank if users and items are considered as concepts and categories. The rating score of a user on an item is computed from the weighted average scores of similar users. The weight of each similar user can be given by cosine similarity on their previous rating on other items. The main difference is this algorithm compute the relatedness in one relation(user-item) while RCRank mutually compute the relatedness between two relations.

5 Conclusion

We proposed a novel method RCRank to jointly compute concept-concept relatedness and concept-category relatedness with the mutually reinforcing assumption of related links and category links. Base on RCRank, we can discover the semantic relationship between concepts with Wikipedia, which is fundamentally different from previous work on relatedness computation. The empirical evaluation results on con-

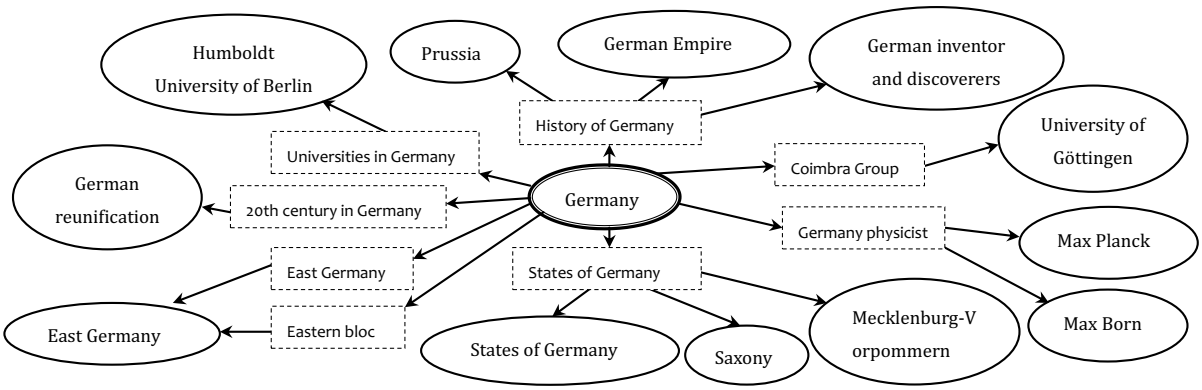

Figure 3: Recommendation results for the concept "Germany" on Wikipedia(circles for concepts and rectangles for categories).

cept recommendation and relation interpretation show that our method substantially outperforms classical methods.

Acknowledgments

This work is supported by Canada's IDRC Research Chair in Information Technology program, Project Number: 104519-006, the Chinese Natural Science Foundation grant No. 60973104, the National Basic Research Program No. 2007CB311003 and China Core High-Tech Project No. 2011ZX01042-001-002. The computing platform is provided by HPC center of Tsinghua University.

A Convergence Analysis

To analysis the convergence of RCRank, we first define the *connected* relation between two concepts as follows.

Definition 2 *Two concept nodes u_i and u_j are **connected** in G iff. there exists a sequence of nodes $\langle u_0 = u_i, u_1, ..., u_n = u_j \rangle$ in which either $\langle u_{k-1}, u_k \rangle \in E_R$ or there exists a category node v such that $\langle u_{k-1}, v \rangle \in E_C$ and $\langle u_{k-1}, v \rangle \in E_C$ for $k \in [1, n]$.*

From definition 2 it is easy to see that *connected* is a equivalence relation according to which the concepts can be partitioned into equivalence classes. The following theorem shows that if u_i and u_j are connected, the relatedness between them will converged to 1.

Theorem 1 *If **C**(0) and **R**(0) are initialized by Eq. 5 and Eq. 6 , we have*

$$\lim_{t\to\infty} r_{ij}(t) = \begin{cases} 1, & \text{if } u_i \text{ and } u_j \text{ are connected} \\ 0, & \text{or else} \end{cases}$$

Theorem 1 can be proved by showing that in each equivalence class, during each iteration, the minimum concept-concept relatedness is larger than or equal to the minimum relatedness on last iteration. The equality holds if and only if the minimum concept-concept relatedness in the corresponding equivalence class reaches to 1.

Theorem 2 *If **C**(0) and **R**(0) are initialized by Eq. 5 and Eq. 6, **D**(t)**C**(t) will converge to a equilibrium in which row i and j are identical if u_i and u_j are connected.*

Convergence of $\mathbf{D}(t)\mathbf{C}(t)$ can be proved straightforward from theorem 1. Each entry (i, j) of the equilibrium of $\mathbf{D}(t)\mathbf{C}(t)$ reflects the popularity of category v_j in the equivalence class containing u_i.

References

[Adafre and de Rijke, 2005] S. F. Adafre and M. de Rijke. Discovering missing links in wikipedia. In *In Workshop on Link Discovery: Issues, Approaches and Applications*, pages 90–97, 2005.

[Breese *et al.*, 1998] J.S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligenc*, pages 43–52, 1998.

[Gabrilovich and Markovitch, 2007] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the IJCAI*, pages 1606–1611, Hyderabad, India, 2007.

[Hu *et al.*, 2009] Jian Hu, Gang Wang, Fred Lochovsky, and Jiantao Sun. Understanding users query intent with wikipedia. In *Proceedings of WWW-09*, pages 471–480, Madrid, Spain, 2009.

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

[Milne, 2007] D. Milne. Computing semantic relatedness using wikipedia link structure. In *Proceedings of the New Zealand Computer Science Research Student conference (NZCSRSC07)*, Hamilton, New Zealand, 2007.

[Ollivier and Senellart, 2007] Y. Ollivier and P. Senellart. Finding related pages using green measures: An illustration with wikipedia. In *Proceedings of the AAAI*, pages 1427–1433, Vancouver, Canada, July 2007.

[Strube and Ponzetto, 2006] M. Strube and S.P. Ponzetto. Wikirelate! computing semantic relatedness using wikipedia. In *Proceedings of the AAAI*, Boston, MA, 2006.