Novel Association Measures Using Web Search with Double Checking

Hsin-Hsi Chen **Ming-Shun Lin** **Yu-Chuan Wei**
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
hhchen@csie.ntu.edu.tw;{mslin,ycwei}@nlg.csie.ntu.edu.tw

Abstract

A web search with double checking model is proposed to explore the web as a live corpus. Five association measures including variants of *Dice*, *Overlap Ratio*, *Jaccard*, and *Cosine*, as well as *Co-Occurrence Double Check* (*CODC*), are presented. In the experiments on Rubenstein-Goodenough's benchmark data set, the *CODC* measure achieves correlation coefficient 0.8492, which competes with the performance (0.8914) of the model using WordNet. The experiments on link detection of named entities using the strategies of *direct association*, *association matrix* and *scalar association matrix* verify that the double-check frequencies are reliable. Further study on named entity clustering shows that the five measures are quite useful. In particular, *CODC* measure is very stable on word-word and name-name experiments. The application of *CODC* measure to expand community chains for personal name disambiguation achieves 9.65% and 14.22% increase compared to the system without community expansion. All the experiments illustrate that the novel model of web search with double checking is feasible for mining associations from the web.

1 Introduction

In statistical natural language processing, resources used to compute the statistics are indispensable. Different kinds of corpora have made available and many language models have been experimented. One major issue behind the corpus-based approaches is: if corpora adopted can reflect the up-to-date usage. As we know, languages are live. New terms and phrases are used in daily life. How to capture the new usages is an important research topic.

The Web is a heterogeneous document collection. Huge-scale and dynamic nature are characteristics of the Web. Regarding the Web as a live corpus becomes an active research topic recently. How to utilize the huge volume of web data to measure association of information is an important issue. Resnik and Smith (2003) employ the Web as parallel corpora to provide bilingual sentences for translation models. Keller and Lapata (2003) show that bigram statistics for English language is correlated between corpus and web counts. Besides, how to get the word counts and the word association counts from the web pages without scanning over the whole collections is indispensable. Directly managing the web pages is not an easy task when the Web grows very fast.

Search engine provides some way to return useful information. Page counts for a query denote how many web pages containing a specific word or a word pair roughly. Page count is different from word frequency, which denotes how many occurrences a word appear. Lin and Chen (2004) explore the use of the page counts provided by different search engines to compute the statistics for Chinese segmentation. In addition to the page counts, snippets returned by web search, are another web data for training. A snippet consists of a title, a short summary of a web page and a hyperlink to the web page. Because of the cost to retrieve the full web pages, short summaries are always adopted (Lin, Chen, and Chen, 2005).

Various measures have been proposed to compute the association of objects of different granularity like terms and documents. Rodríguez and Egenhofer (2003) compute the semantic

Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 1009–1016,
Sydney, July 2006.

similarity from WordNet and SDTS ontology by word matching, feature matching and semantic neighborhood matching. Li et al. (2003) investigate how information sources could be used effectively, and propose a new similarity measure combining the shortest path length, depth and local density using WordNet. Matsuo et al. (2004) exploit the Jaccard coefficient to build "Web of Trust" on an academic community.

This paper measures the association of terms using snippets returned by web search. A web search with double checking model is proposed to get the statistics for various association measures in Section 2. Common words and personal names are used for the experiments in Sections 3 and 4, respectively. Section 5 demonstrates how to derive communities from the Web using association measures, and employ them to disambiguate personal names. Finally, Section 6 concludes the remarks.

2 A Web Search with Double Checking Model

Instead of simple web page counts and complex web page collection, we propose a novel model, a *Web Search with Double Checking* (*WSDC*), to analyze snippets. In *WSDC* model, two objects X and Y are postulated to have an association if we can find Y from X (a forward process) and find X from Y (a backward process) by web search. The forward process counts the total occurrences of Y in the top N snippets of query X, denoted as $f(Y@X)$. Similarly, the backward process counts the total occurrences of X in the top N snippets of query Y, denoted as $f(X@Y)$. The forward and the backward processes form a double check operation.

Under *WSDC* model, the association scores between X and Y are defined by various formulas as follows.

$$VariantDice(X,Y) = \begin{cases} 0 & \text{if } f(Y@X)=0 \text{ or } f(X@Y)=0 \\ \dfrac{f(Y@X)+f(X@Y)}{f(X)+f(Y)} & \text{Otherwise} \end{cases} \quad (1)$$

$$VariantCosine(X,Y) = \frac{min(f(Y@X), f(X@Y))}{\sqrt{f(X)\times f(Y)}} \quad (2)$$

$$VariantJaccard(X,Y) = \frac{min(f(Y@X), f(X@Y))}{f(X)+f(Y)-max(f(Y@X), f(X@Y))} \quad (3)$$

$$VariantOverlap(X,Y) = \frac{min\{f(Y@X), f(X@Y)\}}{min\{f(X), f(Y)\}} \quad (4)$$

$$CODC(X,Y) = \begin{cases} 0 & \text{if } f(Y@X)=0 \text{ or } f(X@Y)=0 \\ e^{log\left(\frac{f(Y@X)}{f(X)}\times\frac{f(X@Y)}{f(Y)}\right)^{\alpha}} & \text{Otherwise} \end{cases} \quad (5)$$

Where $f(X)$ is the total occurrences of X in the top N snippets of query X, and, similarly, $f(Y)$ is the total occurrences of Y in the top N snippets of query Y. Formulas (1)-(4) are variants of the *Dice*, *Cosine*, *Jaccard*, and *Overlap Ratio* association measure. Formula (5) is a function *CODC* (*Co-Occurrence Double-Check*), which measures the association in an interval [0,1]. In the extreme cases, when $f(Y@X)=0$ or $f(X@Y)=0$, $CODC(X,Y)=0$; and when $f(Y@X)=f(X)$ and $f(X@Y)=f(Y)$, $CODC(X,Y)=1$. In the first case, X and Y are of no association. In the second case, X and Y are of the strongest association.

3 Association of Common Words

We employ Rubenstein-Goodenough's (1965) benchmark data set to compare the performance of various association measures. The data set consists of 65 word pairs. The similarities between words, called Rubenstein and Goodenough rating (RG rating), were rated on a scale of 0.0 to 4.0 for "semantically unrelated" to "highly synonymous" by 51 human subjects. The Pearson product-moment correlation coefficient, r_{xy}, between the RG ratings X and the association scores Y computed by a model shown as follows measures the performance of the model.

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{(n-1)s_x s_y} \quad (6)$$

Where $\bar{x}$ and $\bar{y}$ are the sample means of x_i and y_i, and s_x and s_y are sample standard deviations of x_i and y_i and n is total samples.

Most approaches (Resink, 1995; Lin, 1998; Li et al., 2003) used 28 word pairs only. Resnik (1995) obtained information content from WordNet and achieved correlation coefficient 0.745. Lin (1998) proposed an information-theoretic similarity measure and achieved a correlation coefficient of 0.8224. Li et al. (2003) combined semantic density, path length and depth effect from WordNet and achieved the correlation coefficient 0.8914.

	100	200	300	400	500	600	700	800	900
VariantDice	0.5332	0.5169	0.5352	0.5406	0.5306	0.5347	0.5286	**0.5421**	0.5250
VariantOverlap	0.5517	0.6516	0.6973	0.7173	0.6923	0.7259	0.7473	**0.7556**	0.7459
VariantJaccard	0.5533	0.6409	0.6993	0.7229	0.6989	0.738	**0.7613**	0.7599	0.7486
VariantCosine	0.5552	0.6459	0.7063	0.7279	0.6987	0.7398	**0.7624**	0.7594	0.7501
CODC (α=0.15)	0.5629	0.6951	0.8051	0.8473	0.8438	**0.8492**	0.8222	0.8291	0.8182
Jaccard Coeff[*]	0.5847	0.5933	**0.6099**	0.5807	0.5463	0.5202	0.4855	0.4549	0.4622

Table 1. Correlation Coefficients of WSDC Model on Word-Word Experiments

Model	RG Rating	Resnik (1995)	Lin (1998)	Li et al (2003)	*VariantCosine* (#snippets=700) WSDC	*CODC*(α=0.15, #snippets=600) WSDC
Correlation Coefficient	-	0.7450	0.8224	0.8914	0.7624	0.8492
chord-smile	0.02	1.1762	0.20	0	0	0
rooster-voyage	0.04	0	0	0	0	0
noon-string	0.04	0	0	0	0	0
glass-magician	0.44	1.0105	0.06	0	0	0
monk-slave	0.57	2.9683	0.18	0.350	0	0
coast-forest	0.85	0	0.16	0.170	0.0019	0.1686
monk-oracle	0.91	2.9683	0.14	0.168	0	0
lad-wizard	0.99	2.9683	0.20	0.355	0	0
forest-graveyard	1	0	0	0.132	0	0
food-rooster	1.09	1.0105	0.04	0	0	0
coast-hill	1.26	6.2344	0.58	0.366	0	0
car-journey	1.55	0	0	0	0.0014	0.2049
crane-implement	2.37	2.9683	0.39	0.366	0	0
brother-lad	2.41	2.9355	0.20	0.355	0.0027	0.1811
bird-crane	2.63	9.3139	0.67	0.472	0	0
bird-cock	2.63	9.3139	0.83	0.779	0.0058	0.2295
food-fruit	2.69	5.0076	0.24	0.170	0.0025	0.2355
brother-monk	2.74	2.9683	0.16	0.779	0.0027	0.1956
asylum-madhouse	3.04	15.666	0.97	0.779	0.0015	0.1845
furnace-stove	3.11	1.7135	0.18	0.585	0.0035	0.1982
magician-wizard	3.21	13.666	1	0.999	0.0031	0.2076
journey-voyage	3.58	6.0787	0.89	0.779	0.0086	0.2666
coast-shore	3.6	10.808	0.93	0.779	0.0139	0.2923
implement-tool	3.66	6.0787	0.80	0.778	0.0033	0.2506
boy-lad	3.82	8.424	0.85	0.778	0.0101	0.2828
Automobile-car	3.92	8.0411	1	1	0.0144	0.4229
Midday-noon	3.94	12.393	1	1	0.0097	0.2994
gem-jewel	3.94	14.929	1	1	0.0107	0.3530

Table 2. Comparisons of WSDC with Models in Previous Researches

In our experiments on the benchmark data set, we used information from the Web rather than WordNet. Table 1 summarizes the correlation coefficients between the RG rating and the association scores computed by our *WSDC* model. We consider the number of snippets from 100 to 900. The results show that *CODC* > *VariantCosine* > *VariantJaccard* > *VariantOverlap* > *VariantDice*. *CODC* measure achieves the best performance 0.8492 when α=0.15 and total snippets to be analyzed are 600. Matsuo et al. (2004) used *Jaccard* coefficient to calculate similarity between personal names using the Web. The coefficient is defined as follows.

$$JaccardCoff(X,Y) = \frac{f(X \cap Y)}{f(X \cup Y)} \quad (7)$$

Where $f(X \cap Y)$ is the number of pages including *X's* and *Y's* homepages when query "*X* and *Y*" is submitted to a search engine; $f(X \cup Y)$ is the number of pages including *X's* or *Y's* homepages when query "*X* or *Y*" is submitted to a search engine. We revised this formula as follows and evaluated it with Rubenstein-Goodenough's benchmark.

$$JaccardCoff(X,Y)^* = \frac{f_s(X \cap Y)}{f_s(X \cup Y)} \quad (8)$$

Where $f_s(X \cap Y)$ is the number of snippets in which *X* and *Y* co-occur in the top N snippets of query "*X* and *Y*"; $f_s(X \cup Y)$ is the number of snippets containing *X* or *Y* in the top *N* snippets of query "*X* or *Y*". We test the formula on the same benchmark. The last row of Table 1 shows that *Jaccard Coeff** is worse than other models when the number of snippets is larger than 100.

Table 2 lists the results of previous researches (Resink, 1995; Lin, 1998; Li et al., 2003) and our *WSDC* models using *VariantCosine* and *CODC* measures. The 28 word pairs used in the experiments are shown. *CODC* measure can compete with Li *et al*. (2003). The word pair "car-journey" whose similarity value is 0 in the papers (Resink, 1995; Lin, 1998; Li et al., 2003) is captured by our model. In contrast, our model cannot deal with the two word pairs "crane-implement" and "bird-crane".

4 Association of Named Entities

Although the correlation coefficient of *WSDC* model built on the web is a little worse than that of the model built on WordNet, the Web provides live vocabulary, in particular, named entities. We will demonstrate how to extend our *WSDC* method to mine the association of personal names. That will be difficult to resolve with previous approaches. We design two experiments – say, link detection test and named entity clustering, to evaluate the association of named entities.

Given a named-entity set *L*, we define a *link detection test* to check if any two named entities NE_i and NE_j ($i \neq j$) in *L* have a relationship *R* using the following three strategies.

- **Direct Association**: If the double check frequency of NE_i and NE_j is larger than 0,

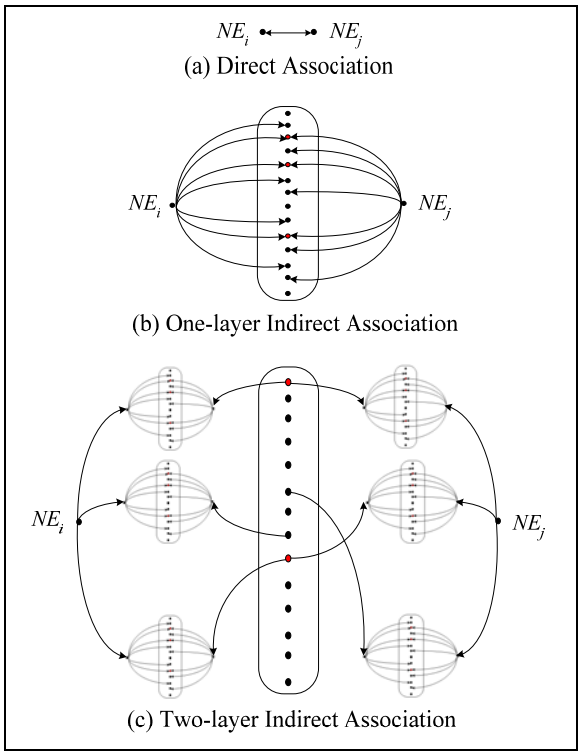


Figure 1. Three Strategies for Link Detection

i.e., $f(NE_j@NE_i)>0$ and $f(NE_i@NE_j)>0$, then the link detection test says "yes", i.e., NE_i and NE_j have *direct association*. Otherwise, the test says "no". Figure 1(a) shows the direct association.

- **Association Matrix**: Compose an $n \times n$ binary matrix $M=(m_{ij})$, where $m_{ij}=1$ if $f(NE_j@NE_i)>0$ and $f(NE_i@NE_j)>0$; $m_{ij}=0$ if $f(NE_j@NE_i)=0$ or $f(NE_i@NE_j)=0$; and *n* is total number of named entities in *L*. Let M^t be a transpose matrix of *M*. The matrix $A=M \times M^t$ is an association matrix. Here the element a_{ij} in *A* means that total a_{ij} common named entities are associated with both NE_i and NE_j directly. Figure 1(b) shows a one-layer indirect association. Here, $a_{ij}=3$. We can define NE_i and NE_j have an indirect association if a_{ij} is larger than a threshold λ. That is, NE_i and NE_j should associate with at least λ common named entities directly. The strategy of association matrix specifies: if $a_{ij} \geq \lambda$, then the link detection test says "yes", otherwise it says "no". In the example shown in Figure 1(b), NE_i and NE_j are indirectly associated when $0<\lambda \leq 3$.

- **Scalar Association Matrix**: Compose a binary association matrix *B* from the association matrix *A* as: $b_{ij}=1$ if $a_{ij}>0$ and $b_{ij}=0$ if $a_{ij}=0$. The matrix $S= B \times B^t$ is a scalar as-

sociation matrix. NE_i and NE_j may indirectly associate with a common named entity NE_k. Figure 1(c) shows a two-layer indirect association. The $s_{ij} = \sum_{k=1}^{n} b_{ik} \times b_{kj}$ denotes how many such an NE_k there are. In the example of Figure 1(c), two named entities indirectly associate NE_i and NE_j at the same time. We can define NE_i and NE_j have an indirect association if s_{ij} is larger than a threshold δ. In other words, if s_{ij} >δ, then the link detection test says "yes", otherwise it says "no".

To evaluate the performance of the above three strategies, we prepare a test set extracted from domz web site (http://dmoz.org), the most comprehensive human-edited directory of the Web. The test data consists of three communities: actor, tennis player, and golfer, shown in Table 3. Total 220 named entities are considered. The golden standard of link detection test is: we compose 24,090 (=220×219/2) named entity pairs, and assign "yes" to those pairs belonging to the same community.

Category Path in domz.org	# of Person Names
Top: Sports: Golf: Golfers	10
Top: Sports: Tennis: Players: Female (+Male)	90
Top: Arts: People: Image Galleries: Female (+Male): Individual	120

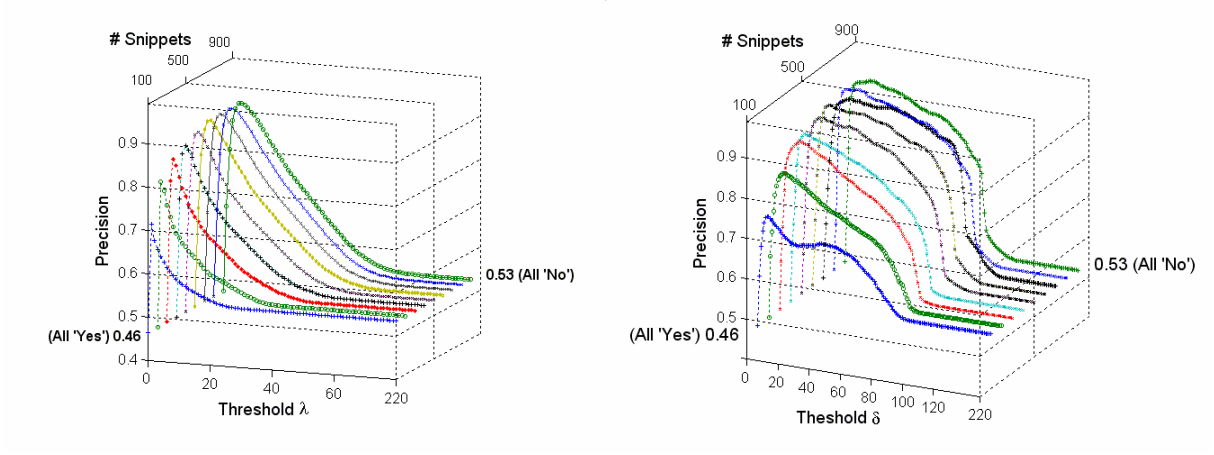
Table 3. Test Set for Association Evaluation of Named Entities

When collecting the related values for computing the double check frequencies for any named entity pair (NE_i and NE_j), i.e., $f(NE_j@NE_i)$, $f(NE_i@NE_j)$, $f(NE_i)$, and $f(NE_j)$, we consider naming styles of persons. For example, "Alba, Jessica" have four possible writing: "Alba, Jessica", "Jessica Alba", "J. Alba" and "Alba, J." We will get top N snippets for each naming style, and filter out duplicate snippets as well as snippets of ULRs including dmoz.org and google.com. Table 4 lists the experimental results of link detection on the test set. The precisions of two baselines are: guessing all "yes" (46.45%) and guessing all "no" (53.55%). All the three strategies are better than the two baselines and the performance becomes better when the numbers of snippets increase. The strategy of direct association shows that using double checks to measure the association of named entities also gets good effects as the association of common words. For the strategy of association matrix, the best performance 90.14% occurs in the case of 900 snippets and λ=6. When larger number of snippets is used, a larger threshold is necessary to achieve a better performance. Figure 2(a) illustrates the relationship between precision and threshold (λ). The performance decreases when λ>6. The performance of the strategy of scalar association matrix is better than that of the strategy of association matrix in some λ and δ. Figure 2(b) shows the relationship between precision and threshold δ for some number of snippets and λ.

In link detection test, we only consider the binary operation of double checks, i.e., $f(NE_j@NE_i) > 0$ and $f(NE_i@NE_j) > 0$, rather than utilizing the magnitudes of $f(NE_j@NE_i)$ and $f(NE_i@NE_j)$. Next we employ the five formulas proposed in Section 2 to cluster named entities. The same data set as link detection test is adopted. An agglomerative average-link clustering algorithm is used to partition the given 220 named entities based on Formulas (1)-(5). Four-fold cross-validation is employed and B-CUBED metric (Bagga and Baldwin, 1998) is adopted to evaluate the clustering results. Table 5 summarizes the experimental results. *CODC* (Formula 5), which behaves the best in computing association of common words, still achieves the better performance on different numbers of snippets in named entity clustering. The F-scores of the other formulas are larger than 95% when more snippets are considered to compute the double check frequencies.

Strategies	100	200	300	400	500	600	700	800	900
Direct Association	59.20%	62.86%	65.72%	67.88%	69.83%	71.35%	72.05%	72.46%	**72.55%**
Association Matrix	71.53% (λ=1)	79.95% (λ=1)	84.00% (λ=2)	86.08% (λ=3)	88.13% (λ=4)	89.67% (λ=5)	89.98% (λ=5)	90.09% (λ=6)	**90.14%** (λ=6)
Scalar Association Matrix	73.93% (λ=1, δ=6)	82.69% (λ=2, δ=9)	86.70% (λ=4, δ=9)	88.61% (λ=5, δ=10)	90.90% (λ=6, δ=12)	91.93% (λ=7, δ=12)	91.90% (λ=7, δ=18)	92.20% (λ=10, δ=16)	**92.35%** (λ=10, δ=18)

Table 4. Performance of Link Detection of Named Entities



(a) (b)

Figure 2. (a) Performance of association matrix strategy. (b) Performance of scalar association matrix strategy (where λ is fixed and its values reference to scalar association matrix in Table 4)

		100	200	300	400	500	600	700	800	900
VariantDice	P	91.70%	88.71%	87.02%	87.49%	96.90%	100.00%	100.00%	100.00%	100.00%
	R	55.80%	81.10%	87.70%	93.00%	89.67%	93.61%	94.42%	94.88%	94.88%
	F	69.38%	84.73%	87.35%	90.16%	93.14%	96.69%	97.12%	**97.37%**	**97.37%**
VariantOverlap	P	99.13%	87.04%	85.35%	85.17%	88.16%	88.16%	88.16%	97.59%	98.33%
	R	52.16%	81.10%	86.24%	93.45%	92.03%	93.64%	92.82%	90.82%	93.27%
	F	68.35%	83.96%	85.79%	89.11%	90.05%	90.81%	90.43%	94.08%	**95.73%**
VariantJaccard	P	99.13%	97.59%	98.33%	95.42%	97.59%	88.16%	95.42%	100.00%	100.00%
	R	55.80%	77.53%	84.91%	88.67%	87.18%	90.58%	88.67%	93.27%	91.64%
	F	71.40%	86.41%	91.12%	91.92%	92.09%	89.35%	91.92%	**96.51%**	95.63%
VariantCosine	P	84.62%	97.59%	85.35%	85.17%	88.16%	88.16%	88.16%	98.33%	98.33%
	R	56.22%	78.92%	86.48%	93.45%	92.03%	93.64%	93.64%	93.27%	93.27%
	F	67.55%	87.26%	85.91%	89.11%	90.05%	90.81%	90.81%	**95.73%**	**95.73%**
CODC (α=0.15)	P	91.70%	87.04%	87.02%	95.93%	98.33%	95.93%	95.93%	94.25%	94.25%
	R	55.80%	81.10%	90.73%	94.91%	94.91%	96.52%	98.24%	98.24%	98.24%
	F	69.38%	83.96%	88.83%	95.41%	96.58%	96.22%	**97.07%**	96.20%	96.20%

Table 5. Performance of Various Scoring Formulas on Named Entity Clustering

5 Disambiguation Using Association of Named Entities

This section demonstrates how to employ association mined from the Web to resolve the ambiguities of named entities. Assume there are n named entities, NE_1, NE_2, …, and NE_n, to be disambiguated. A named entity NE_j has m accompanying names, called *cue names* later, CN_{j1}, CN_{j2}, …, CN_{jm}. We have two alternatives to use the cue names. One is using them directly, i.e., NE_j is represented as a *community* of cue names $Community(NE_j)=\{CN_{j1}, CN_{j2}, \ldots, CN_{jm}\}$. The other is to expand the cue names CN_{j1}, CN_{j2}, …, CN_{jm} for NE_j using the web data as follows. Let CN_{j1} be an initial seed. Figure 3 sketches the concept of community expansion.

(1) **Collection**: We submit a seed to Google, and select the top N returned snippets. Then, we use suffix trees to extract possible patterns (Lin and Chen, 2006).

(2) **Validation**: We calculate *CODC* score of each extracted pattern (denoted B_i) with the seed A. If $CODC(A,B_i)$ is strong enough, i.e., larger than a

threshold θ, we employ B_i as a new seed and repeat steps (1) and (2). This procedure stops either expected number of nodes is collected or maximum number of layers is reached.

(3) **Union**: The community initiated by the seed CN_{ji} is denoted by *Community*(CN_{ji})={B_{ji1}, B_{ji2}, …, B_{jir}}, where B_{jik} is a new seed. The *Cscore* score, *community score*, of B_{jik} is the *CODC* score of B_{jik} with its parent divided by the layer it is located. We repeat *Collection* and *Validation* steps until all the cue names CN_{ji} ($1 \le i \le m$) of NE_j are processed. Finally, we have

$$Community(NE_j) = \cup_{i=1}^{m} Community(CN_{ji})$$

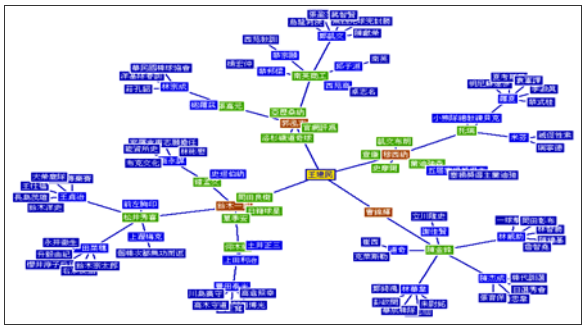


Figure 3. A Community for a Seed "王建民" ("Chien-Ming Wang")

In a cascaded personal name disambiguation system (Wei, 2006), association of named entities is used with other cues such as titles, common terms, and so on. Assume k clusters, c_1 c_2 ... c_k, have been formed using title cue, and we try to place NE_1, NE_2, …, and NE_l into a suitable cluster. The cluster $\bar{c}$ is selected by the similarity measure defined below.

$$score(NE_j, c_q) = \frac{1}{r}\sum_{i=1}^{s} count(pn_i) \times Cscore(pn_i) \quad (9)$$

$$\bar{c} = \arg\max_{c_q (1 \le q \le k)} score(NE_j, c_q) \quad (10)$$

Where pn_1, pn_2, …, pn_s are names which appear in both *Community*(NE_j) and *Community*(c_q); *count*(pn_i) is total occurrences of pn_i in *Community*(c_q); r is total occurrences of names in *Community*(NE_j); *Cscore*(pn_i) is community score of pn_i.

If *score*(NE_j, $\bar{c}$) is larger than a threshold, then NE_j is placed into cluster $\bar{c}$. In other words, NE_j denotes the same person as those in $\bar{c}$. We let the new *Community*($\bar{c}$) be the old *Community*($\bar{c}$)$\cup${CN_{j1}, CN_{j2}, …, CN_{jm}}. Otherwise, NE_j is left undecided.

To evaluate the personal name disambiguation, we prepare three corpora for an ambiguous name "王建民" (Chien-Ming Wang) from United Daily News Knowledge Base (UDN), Google Taiwan (TW), and Google China (CN). Table 6 summarizes the statistics of the test data sets. In UDN news data set, 37 different persons are mentioned. Of these, 13 different persons occur more than once. The most famous person is a pitcher of New York Yankees, which occupies 94.29% of 2,205 documents. In TW and CN web data sets, there are 24 and 107 different persons. The majority in TW data set is still the New York Yankees's "Chien-Ming Wang". He appears in 331 web pages, and occupies 88.03%. Comparatively, the majority in CN data set is a research fellow of Chinese Academy of Social Sciences, and he only occupies 18.29% of 421 web pages. Total 36 different "Chien-Ming Wang"s occur more than once. Thus, CN is an unbiased corpus.

	UDN	TW	CN
# of documents	2,205	376	421
# of persons	37	24	107
# of persons of occurrences>1	13	9	36
Majority	94.29%	88.03%	18.29%

Table 6. Statistics of Test Corpora

		M1	M2	
UDN	P	0.9742	0.9674	(↓0.70%)
	R	0.9800	0.9677	(↓1.26%)
	F	0.9771	0.9675	(↓0.98%)
TW	P	0.8760	0.8786	(↑0.07%)
	R	0.6207	0.7287	(↑17.40%)
	F	0.7266	0.7967	(↑9.65%)
CN	P	0.4910	0.5982	(↑21.83%)
	R	0.8049	0.8378	(↑4.09%)
	F	0.6111	0.6980	(↑14.22%)

Table 7. Disambiguation without/with Community Expansion

Table 7 shows the performance of a personal name disambiguation system without (M1)/with (M2) community expansion. In the news data set (i.e., UDN), M1 is a little better than M2. Compared to M1, M2 decreases 0.98% of F-score. In contrast, in the two web data sets (i.e., TW and CN), M2 is much better than M1. M2 has 9.65% and 14.22% increases compared to M1. It shows that mining association of named entities from the Web is very useful to disambiguate ambiguous names. The application also confirms the effectiveness of the proposed association measures indirectly.

6 Concluding Remarks

This paper introduces five novel association measures based on web search with double checking (*WSDC*) model. In the experiments on association of common words, *Co-Occurrence Double Check* (*CODC*) measure competes with the model trained from WordNet. In the experiments on the association of named entities, which is hard to deal with using WordNet, *WSDC* model demonstrates its usefulness. The strategies of direct association, association matrix, and scalar association matrix detect the link between two named entities. The experiments verify that the double-check frequencies are reliable.

Further study on named entity clustering shows that the five measures – say, *VariantDice*, *VariantOverlap*, *ariantJaccard*, *VariantCosine* and *CODC*, are quite useful. In particular, *CODC* is very stable on word-word and name-name experiments. Finally, *WSDC* model is used to expand community chains for a specific personal name, and *CODC* measures the association of community member and the personal name. The application on personal name disambiguation shows that 9.65% and 14.22% increase compared to the system without community expansion.

Acknowledgements

Research of this paper was partially supported by National Science Council, Taiwan, under the contracts 94-2752-E-001-001-PAE and 95-2752-E-001-001-PAE.

References

A. Bagga and B. Baldwin. 1998. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. *Proceedings of 36th COLING-ACL Conference*, 79-85.

F. Keller and M. Lapata. 2003. Using the Web to Obtain Frequencies for Unseen Bigrams. *Computational Linguistics*, 29(3): 459-484.

Y. Li, Z.A. Bandar and D. McLean. 2003. An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. *IEEE Transactions on Knowledge and Data Engineering*, 15(4): 871-882.

D. Lin. 1998. An Information-Theoretic Definition of Similarity. *Proceedings of the Fifteenth International Conference on Machine Learning*, 296-304.

H.C. Lin and H.H. Chen. 2004. Comparing Corpus-based Statistics and Web-based Statistics: Chinese Segmentation as an Example. *Proceedings of 16th ROCLING Conference*, 89-100.

M.S. Lin, C.P. Chen and H.H. Chen. 2005. An Approach of Using the Web as a Live Corpus for Spoken Transliteration Name Access. *Proceedings of 17th ROCLING Conference*, 361-370.

M.S. Lin and H.H. Chen. 2006. Constructing a Named Entity Ontology from Web Corpora. *Proceedings of 5th International Conference on Language Resources and Evaluation*.

Y. Matsuo, H. Tomobe, K. Hasida, and M. Ishizuka. 2004. Finding Social Network for Trust Calculation. *Proceedings of 16th European Conference on Artificial Intelligence*, 510-514.

P. Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 448-453.

P. Resnik and N.A. Smith. 2003. The Web as a Parallel Corpus. *Computational Linguistics*, 29(3): 349-380.

M.A. Rodríguez and M.J. Egenhofer. 2003. Determining Semantic Similarity among Entity Classes from Different Ontologies. *IEEE Transactions on Knowledge and Data Engineering*, 15(2): 442-456.

H. Rubenstein and J.B. Goodenough. 1965. Contextual Correlates of Synonymy. *Communications of the ACM*, 8(10): 627-633.

Y.C. Wei. 2006. *A Study of Personal Name Disambiguation*. Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, Taiwan.