Extracting Social Networks Among Various Entities on the Web

Yingzi Jin[1], Yutaka Matsuo[2], and Mitsuru Ishizuka[1]

[1] University of Tokyo, Hongo 7–3–1, Tokyo 113-8656, Japan
eiko-kin@mi.ci.i.u-tokyo.ac.jp, ishizuka@i.u-tokyo.ac.jp
[2] National Institute of Advanced Industrial Science and Technology
y.matsuo@aist.go.jp

Abstract. Social networks have recently attracted much attention for their importance to the Semantic Web. Several methods exist to extract social networks for people (particularly researchers) from the web using a search engine. Our goal is to expand existing techniques to obtain social networks among various entities. This paper proposes two improvements, i.e. *relation identification* and *threshold tuning*, which enable us to deal with complex and inhomogeneous communities. Social networks among firms and artists (of contemporary) are extracted as examples: Several evaluations emphasize the effectiveness of these methods. Our system was used at the International Triennale of Contemporary Art (Yokohama Triennale 2005) to facilitate navigation of artists' information. This study contributes to the Semantic Web in that we increase the applicability of social network extraction for several studies.

1 Introduction

Social networks explicitly exhibit relationships (called *ties* in social sciences) among individuals and groups (called *actors*). They have been studied in social sciences since the 1930s. To date, vastly numerous studies using social network analysis have been conducted [22]. In the context of the Semantic Web, social networks are crucial to realize a web of trust that facilitates estimation of information's credibility and its provider's trustworthiness [10]. Ontology construction is also related to social networks: P. Mika discusses the relation between the community and emergent ontology from a social network perspective [18]. Information sharing and recommendation [19,9] on social networks are other applications that are served by the Semantic Web. Our lives are influenced strongly by social networks without our knowledge of their implications. For that reason, many applications are relevant to social networks [23].

Social networks are obtained from various sources, such as e-mail archives, FOAF documents, and DBLP. For example, T. Finin et al. extract a social network from the web by collecting FOAF documents [7]. Particularly, several studies have been undertaken to use a search engine to extract social networks from the entire web [11,16,17]. Co-occurrence of names on the web, which is basically obtained by posing a query including two names to a search engine, is commonly used as proof of relational strength. Using a search engine to recognize the relation of two entities (or two words) has increasingly gained attention in the field of natural language processing [5,12,24].

E. Franconi, M. Kifer, and W. May (Eds.): ESWC 2007, LNCS 4519, pp. 251–266, 2007.

This study is intended to expand current social-network mining techniques using a search engine to obtain a social network among various entities. Specifically in this paper, two improvements are proposed in order to apply our method to complex and inhomogeneous communities: *relation identification* and *threshold tuning*. We extract two social networks as examples: artists of contemporary art, and famous firms in Japan. We must identify the relation types such as alliances and lawsuits; consequently, we can make elaborate queries and apply text processing to extract a social network among firms. Our algorithm adds a *relation keyword* to the search query to emphasize a specific relationship. Extracting a social network of artists, on the other hand, requires adaptive tuning of thresholds because the appearance of each artist on the web is completely different. Optimal thresholds are sought to invent appropriate edges between entities.

Our contributions are summarized as follows: First, through the two improvements, i.e. relation identification and threshold tuning, which respectively focus on complex and inhomogeneous communities on the web, social network extraction becomes more generally applicable to various entities. We argue the general social network extraction in the last part of the paper, which can cultivate existing studies using social networks in the Semantic Web. Second, because our method can extract relations from among entities, it can output machine-processable knowledge about the relations automatically from the information on the current web. Although some approaches exist to generate RDF statements by web mining, our study provides an alternative; our intuition is that extracting a social network might provide information that is only recognizable from the network point of view. For example, the *centrality* of each firm is identified only after generating a social network.

The next section introduces related studies. Section 3 describes the investigation of different appearance of entities on the web and addresses our ideas to obtain various social networks from the web. Sections 4 and 5 introduce our case studies, which specifically investigate two types of networks: those of firms and artists. In Section 6, before we conclude the paper, we propose a general architecture of social network extraction and discuss applications of the extracted social networks to the Semantic Web.

2 Related Works

Numerous studies have obtained and analyzed social networks on the web: L. Adamic collects relations among students from web link structure and text information, and characterizes the social networks among Stanford students and MIT students [1]. T. Finin describes a large collection of FOAF documents (over 1.5 million) from the web and analyzes the structure of friendship networks in the Semantic Web [7]. Trust calculation [10] is a major application of social networks. Some studies seek other applications: A. McCallum and his group present an end-to-end system that automatically integrates both e-mail and web content to help users maintain large contact databases [6]. Aleman-Meza et al. use relational data from both FOAF and DBLP to detect relationships among potential reviewers and authors of scientific papers [2].

Several studies have particularly addressed use of a search engine for social network extraction. In the mid-1990s, H. Kautz and B. Selman developed a social network extraction system called the *Referral Web* [11]. The system uses a search engine to retrieve web documents that include a given personal name. Recently, P. Mika developed

Flink, a system for extraction, aggregation, and visualization of online social networks for the Semantic Web community [17]. A social network of 608 researchers from both academia and industry is extracted and analyzed. The web-mining component of Flink, similarly to that used in Kautz's work, employs co-occurrence analysis. The strength of relevance of two persons, X and Y, is estimated by putting a query X AND Y to a search engine: If X and Y share a strong relation, we can usually find much evidence on the web such as links found on home pages, lists of co-authors in technical papers, organizational charts, and so on. In Flink, the strength of relations among individuals is calculated using the Jaccard coefficient $n_{X\cap Y}/n_{X\cup Y}$, where $n_{X\cap Y}$ represents the number of hits yielded by the query X AND Y and $n_{X\cup Y}$ represents the number of hits by the query X OR Y. The two researchers are considered to share a relation if the value is greater than a certain threshold. The term "*Semantic Web* OR *ontology*" is added to the query for name disambiguation.

Matsuo et al. developed a system called *POLYPHONET*, which also uses a search engine to measure the co-occurrence of names [15,16]. In their study, several co-occurrence measures [13] have been compared, including the matching coefficient ($n_{X\cap Y}$), mutual information, Dice coefficient, Jaccard coefficient, and overlap coefficient. The overlap coefficient $n_{X\cap Y}/\min(n_X, n_Y)$ performs best according to the experiments. In addition, POLYPHONET was operated at several AI conferences in Japan and a couple of international conferences to promote participants' communication. For disambiguating personal names, key phrases such as affiliations are added to queries.

We regard the two studies by Mika and Matsuo as relevant precedent studies, and propose some improvements to increase the applicability of that approach.

3 Extraction of Social Networks

3.1 Problem of Existing Methods

The fundamental idea underlying the existing studies by Mika and Matsuo is that *the strength of a relation between two entities can be estimated by co-occurrence of their names on the web*. The criteria to recognize a relation, such as the measure of co-occurrence and a threshold, are determined beforehand. An edge will be invented when the relation strength by the co-occurrence measure is higher than the predefined threshold. Although the approach is effective for extracting a social network of researchers, our preliminary study indicates that it does not perform well for various entities on the web.

As the first reason, co-occurrence-based methods become ineffective when two entities co-occur universally on numerous web pages. For example, when we want to infer two firms' relations from the web, we submit a query "*Matsushita* AND *JustSystem*"[1] to a search engine. Consequently, we are referred to as many as 425,000 pages, for which the Jaccard coefficient is 0.031. However, this figure is unreliable considering the media effect on the web. In the domain of firms, many relations are published in news reports and on news releases that are distributed on the web. Many web pages describe and comment on the relation if the news is given attention by media services or people. Conversely, if it were not attention given, only a small number of pages

[1] Both are names of famous Japanese corporations.

would describe the relations. Considering that media effects influence the number of web pages, co-occurrence of names on the web is not always available to represent the relational strength of two entities.

For the second reason, co-occurrence-based methods function ineffectively when applied to *inhomogeneous* communities. An inhomogeneous community means, in this paper, a community that includes people in different fields, different nations, or different cultures, where a relation is difficult to obtain using a single criterion. The researchers' communities (of the same research field) usually present a homogeneous character; for that reason, using a single criterion to calculate the relation works well. In contrast, the international artist community is more inhomogeneous. For example, two Japanese artists, "*Taisuke Abe*" and "*Jun Oenoki*", have no prior relationship, but their Jaccard coefficient is high: 0.024. Two international artists "*Beat Streuli*" from Switzerland and "*Nari Ward*" from Jamaica have co-participated in several exhibitions, but their coefficient is low: 0.0009. This happens because the community consists of many people from different contexts. For that reason, it is difficult to precisely recognize the relation using a single criterion.

We consider that the precedent studies on the research domain implicitly use the following two assumptions:

Assumption 1. Generally, web pages are created according to results of two actors' co-participation in events. Therefore, the number of web pages is assumed to show a useful correlation to the strength of two actors.

Assumption 2. A community to be extracted as a social network is assumed to be homogeneous.

In the following section, we will introduce our improvements, *relation identification* (in Section *3.2*) and *threshold tuning* (in Section *3.3*), which respectively mitigate violations of these assumptions. Furthermore, to emphasize the effectiveness of our methods, we apply each method to our case studies: Extracting social networks of firms (in Section *4*) and artists (in Section *5*). A general extraction model bundling these different extraction methods will be described in Section *6*.

3.2 Relation Identification

In social sciences, the definition of a weak or strong tie might vary among contexts [14]. For example, the frequency or degree of relations affects that strength; multiple relations between two actors also can imply a stronger tie. In the firm case, the types of relations define the strength: For example, a capital alliance relation is stronger than a business alliance relation. Consequently, to present a tie among firms, it is appropriate that we identify the concrete relations of firms. As a solution, we add some word or combination of words to a search query. Using this strategy, we can efficiently identify relations among firms. For example, when we wish to extract lawsuit relations, we add a term "*lawsuit*". We issue a query "*Matsushita* AND *JustSystem* AND *lawsuit*" so that the search engine will return the lawsuit pages that are associated with the two firms. Then we can conduct text processing to these pages to validate the relation's existence. This idea is similar to keyword spices [20], which extend queries for domain-specific web searches. Question answering systems also construct elaborate queries for using search engines [21].

We call the keyword to be added a *relation keyword*. By adding relation keywords, we can extract particular relations among entities, which can be a solution for validation of **Assumption 1**. Below, we explain some issues about relation types and extraction of relation keywords.

Relation Types. It is considered that a pair of entities has multiple relations. For example, two firms share alliance and lawsuit relations. Each relation is typed in a more detailed way. Alliance relations between firms include capital alliances and business alliances, where the former usually represents a stronger relation than the later. A lawsuit relation has multiple stages: at some time, it will be settled by mutual accommodation or by final judgement. Consequently, the relation can be typed into the claim phase and the accommodation phase. For dynamic and complex relational networks, it is important to distinguish such typical and temporal relations for detailed analyses of social networks [14,22].

Relation Keyword Extraction. To extract particular types of relations between firms, we need some relation keywords. The intuitive method for finding relation keywords is to select terms that appear often in target pages (where the target relation is described) and which do not appear in other pages. Therefore, as a training corpus, we must collect annotated web pages that describe specific relations of the firms. Once we find appropriate relation keywords, we can extract the relations among many firms.

Collecting and annotating the training corpus requires many hours of tedious work. In our study, we also try to use a search engine to extract relation keywords. This method is identical to that of Mori's work [19], in which a specific word w_c is assigned, which can represent the relation most precisely. If we want to retrieve an alliance relation, we add "*alliance*" (denoted as w_c) to a search query; words that co-occur frequently with it also become good clues to discern the relation. We use the Jaccard coefficient $n_{w_c \cap w}/n_{w_c \cup w}$ to measure relevance of word w to word w_c. The words w with large Jaccard coefficients are also used as relation keywords aside from w_c. It would saves costs of annotating training data with relevance or non-relevance manually.

3.3 Threshold Tuning

In studies of social network analysis, network questionnaires have traditionally been conducted. Typically, participants are asked "Please name your four closest friends." The respondents would then list the relations that are personally important. In other words, the relation is recognized by a subjective criterion for each participant. We propose to use this subjective criterion for the solution against **Assumption 2**. For example, even if the relation between "*Beat Streuli*" and "*Nari Ward*" is weaker than the objective standard, it is important to "*Beat Streuli*" if there are no other persons with a stronger relation. Consequently, we might add an edge between them.

We employ two criteria that correspond to objective and subjective importance of relations for actors. We first invent edges using objective criteria with a consistent threshold T. Then we invent edges using subjective criteria for actors who have no certain number M of edges. This procedure alleviates the problem of some nodes having too many edges and some nodes being isolated. The combination of two criteria enables

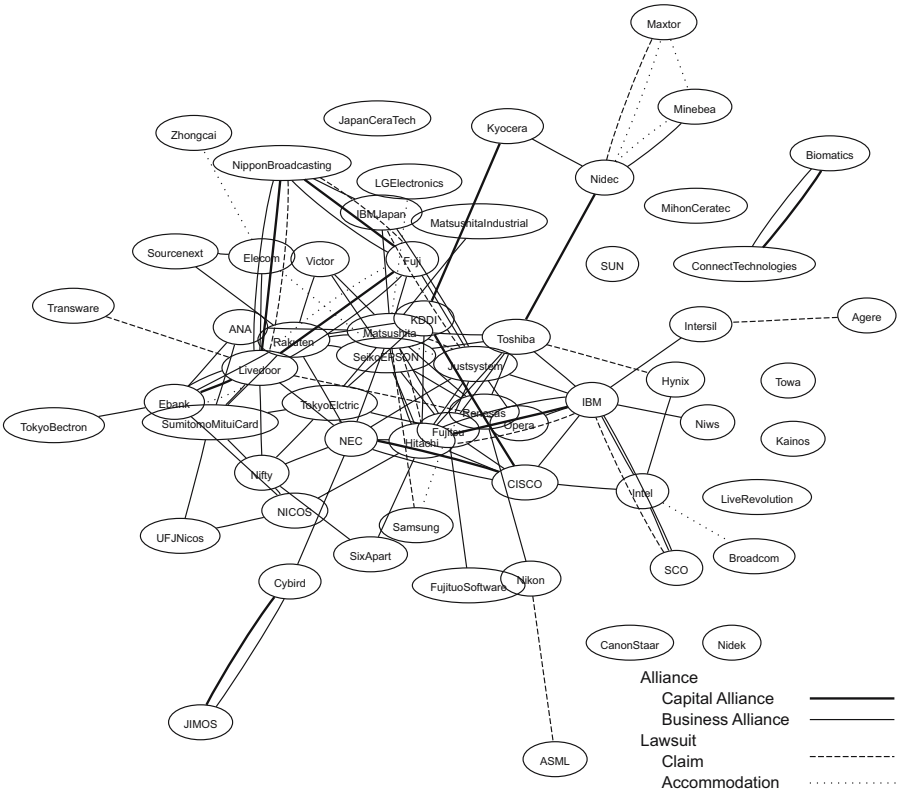


Fig. 1. Social network of 60 firms in Japan

more exhaustive extraction for every node than the previous method, although it sometimes yields low precision. For that reason, we must find the appropriate parameters so that the target network is extracted as precisely as possible.

Setting Parameters for Each Community. Parameters vary according to the domain of a community. For example, T in the researcher community might be higher than that in artist community, simply because researchers' names are more likely appear on the web than artists' names. Therefore, some training data are necessary to learn the appropriate values for each target community. Simply, the parameters are tuned so that the performance of relation identification is maximized: We maximize the F-value. More effective ways to determine the parameters are bootstrapping or user interaction. For the bootstrapping method, we can repeat the sampling and estimation process to determine parameters; for the user interaction method, we can use the users' feedback to reconstruct the network dynamically using the best parameters that can maximize the F-value.

4 Social Network Extraction for Firms

We describe the extraction of a firm network as a case study of *relation identification* (mentioned in Section *3.2*). Many relationships among firms are published in news

Table 1. Relation keywords extracted from the web using Jaccard coefficient

Alliance relation	t_w	**Capital alliance**	t_w	**Business alliance**	t_w
alliance AND *corporate*	1.000	*operation* AND *capital*	1.000	*alliance* AND *business*	1.000
alliance AND *stock*	0.878	*capital* AND *manage*	0.553	*alliance* AND *company*	0.475
alliance AND *company*	0.704	*capital* AND *company*	0.548	*alliance* AND *operation*	0.459
alliance AND *system*	0.565	*capital*	0.543	*alliance* AND *develop*	0.437
alliance AND *business*	0.534	*capital* AND *manage*	0.533	*alliance* AND *company*	0.432

Lawsuit relation	t_w	**Claim phase**	t_w	**Accommodation phase**	t_w
violate AND *lawsuit*	1.000	*violate* AND *sue*	1.000	*lawsuit* AND *accommodate*	1.000
violate AND *claim*	0.514	*patent* AND *sue*	0.533	*accommodate* AND *company*	0.648
violate AND *judge*	0.490	*sue* AND *technology*	0.486	*accommodate* AND *announce*	0.646
violate AND *court*	0.458	*sue* AND *develop*	0.483	*accommodate* AND *develop*	0.641
violate AND *indemnify*	0.444	*sue* AND *relevance*	0.469	*accommodate* AND *product*	0.640

articles and on news releases that are distributed on the web. In our work, we extract alliance and lawsuit relations as respective representatives of positive and negative relations among firms. We further distinguish these relations into two detailed relations: capital and business alliance relations, and claims and accommodation of lawsuit relations. A social network of 60 firms in Japan is extracted; it includes IT, communication, broadcasting, and electronics firms. We will describe details of our system and experimental results.

4.1 System Flow

Our system has two major procedures: an online procedure and an offline procedure. In the offline procedure, relation keywords for each relation are obtained beforehand using the methods introduced in Section *3.2*. We gathered 456 pages and 165 pages for alliance and lawsuit relations, respectively, from Nikkei Net and IP News site[2]. As preprocessing, we first eliminate all html tags and scripts; then we extract the body text of pages and apply a part-of-speech tagger Chasen[3] to choose nouns and verbs (except stop words). These words are candidates of relation keywords. We also use combinations of two words as candidates. We measure the score of

function $R_{ELATION}E_{XTRACTION}$ (D, x, y, W)
  $score_{xy} \leftarrow 0$
  $S \leftarrow$ GetSentences(D, x, y)
  for each $s \in S$ **do**
    if s **contains** "x" *and* s **contains** "y" **then**
    $score_s \leftarrow \sum_{w_i(\in W)}$ **contained in** $_s\, t_{w_i}$
    if $score_s > score_{xy}$ **then**
      $score_{xy} \leftarrow score_s$
  done
  if $score_{xy} > score_{thre}$ **then**
    do set an edge between x and y in G
  done

Fig. 2. A procedure to extract relations by text processing

[2] Nikkei Net (http://release.nikkei.co.jp/) is a famous online business newspaper. IP News (http://news.braina.com/judge.html) is an online news archive on intellectual property issues.
[3] http://chasen.naist.jp/hiki/ChaSen/

each candidate word / phrase by calculating the Jaccard coefficient with specific relation keywords w_c[4]. Candidates with the highest scores are recognized as relation keywords. Table 1 shows the top five relation keywords and their Jaccard scores denoted as t_w[5].

In the online procedure, a list of firms and specific relation types is given as input; the output is a social network of firms. Three steps exist: making queries, Google search, and network construction. First, we make queries by adding relation keywords to each pair of firms. We use top n_q relation keywords from Table 1. Then, we put these queries into the Google search engine to collect top-n_p web pages. (In this experiment, we set $n_q = 2$ and $n_p = 5$.) Lastly, for each downloaded document D, we conduct text processing to judge whether or not the relation actually exists. A simple pattern-based heuristic (as described in Fig. 2) is useful in our experience: We first pick up all sentences S that include the two firm names (x and y), and assign each sentence the sum of relation keyword scores t_w in the sentence. The score of firms x and y is the maximum of the sentence scores. If $score_{xy}$ is greater than a certain threshold (in other words, if the two firms seem to have the target relation with high reliability), an edge is invented between the two firms.

4.2 Results and Evaluation

The obtained network for 60 firms in Japan is shown in Fig. 1. Black lines represent alliances (bold ones are capital alliances and thin ones are business alliances) and red lines represent lawsuits (bold ones are in the claim phase and thin ones are in the accommodation phase).

The precision and recall of our system are shown in Table 2. For ${}_{60}C_2 = 1770$ pairs of firms, 113 pairs actually show alliance relations. Our system extracted 70 pairs correctly. There were actually 21 and 100 pairs of capital and business alliances; our system extracted 9 and 60, respectively. Compared with alliances, the lawsuit relations have higher recall, probably because lawsuit relations are described in rather common formats using words such as *judgment*, *lawsuit*, or *accommodate*.

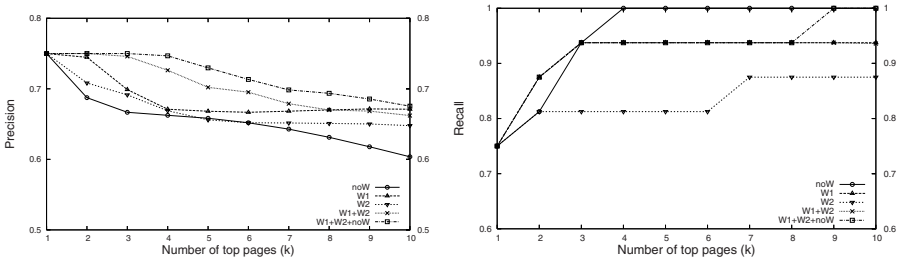
Table 2. Precision and Recall of the System

Target relation	Precision	Recall
Alliance	60.9% (70/115)	62.0% (70/113)
capital alliance	75.0% (9/12)	42.9% (9/21)
business alliance	67.4% (60/89)	60.0% (60/100)
Lawsuit	61.5% (16/26)	100% (16/16)
claim phase	63.6% (14/22)	87.5% (14/16)
accommodation	72.7% (8/11)	88.9% (8/9)

Although they are not comparable technically, we obtained alliance and lawsuit relations from Nikkei Net and IP News, and compared the precision and recall to our results. The precision values at these sites are 100%, but the recall of alliance and lawsuit relations among 60 firms are low: 22.8% and 68.8%, respectively. This is true because these sites deal little with information on small companies and corporations that are capitalized with foreign capital (i.e. foreign companies).

[4] We used *alliance* AND *corporate* as w_c for alliance relations. Furthermore, we use the word appearing in the first lines in Table 1 as w_c for each relation: We determine these words through preliminary experiments.

[5] In our experiment, we mainly used web pages that had been composed in Japanese. For that reason, relation keywords are translated from Japanese.



(a) Precision of retrieved pages (b) Recall of relations

Fig. 3. Evaluation of relation keywords for lawsuit relations

Some detected relations are wrong: As one example, Hitachi and IBM are shown to be embroiled in a lawsuit relation, but they actually are not. Our algorithm took the sentence *"Hitachi and HDD, a subsidiary of IBM have been sued a Chinese HDD maker for patent violations"* as spurious proof of a lawsuit relation. Some relations are described using uncommon phrases (such as *trouble* and *uproar*) that do not appear often in the training corpus. More sophisticated text processing might improve the results in these cases.

4.3 Effectiveness of Relation Keywords

The effectiveness of relation keywords is shown in this section. We compared the information contained in retrieved pages merely by using a pair of names as a search query to add relation keywords to the query. We compared the five methods described below:

noW: A firm pair (without relation keywords) is used as a query.
W1: A firm pair and the top-weighted relation keyword (w_1) are used as a query.
W2: A firm pair and the second-weighted relation keyword (w_2) are used as a query.
W1+ W2: It generates two queries – W1 and W2.
W1+W2+noW: It generates three queries – W1, W2, and noW.

The **noW** is considered to be the existing method (i.e. Mika and Matsuo's method). The others are variations of the proposed method. In all cases, we downloaded the same number of web pages. All other conditions are identical.

Figure 3 shows the results. Overall, the proposed methods perform better than the existing method (**noW**) with respect to precision. The precision and recall are respectively 65.7% and 95.0% if we do not use any relation keywords. Relation keywords improve the precision using the same number of downloaded documents. By integrating multiple queries (as **W1+W2+noW** case), we can achieve the highest precision as 71.9% while retaining high recall (92.5%).

5 Social Network Extraction for Artists

In this section, we describe the algorithm of *threshold tuning* (described in Section *3.3*) for extracting a social network of artists of contemporary art.

```
/* First, we invent edges using two objective criteria: T_ov and T_co. */ ..... (step 1)
for each x ∈ L and y ∈ L
    if (overlap(x, y)> T_ov AND cooc(x, y)> T_co)
        do set an edge between x and y in G

/* Then, invent edges using two subjective criteria M_1 and M_2 (≤ M_1). */ ..... (step 2)
for each x ∈ L
    do Y_x ← ConnectedNodes(x),      /* Y_x are nodes set connected with x. */
       Ȳ_x ← L \ Y_x, Ȳ'_x ← L \ Y_x
    while |Y_x| < M_1 and Ȳ_x ≠ φ      /* |Y_x| is the number of nodes in Y_x. */
        y ← argmax overlap(x, y_j), Ȳ_x ← Ȳ_x \ {y}
                y_j∈Ȳ_x
        if overlap(x, y)> T_ov OR cooc(x, y)> T_co ..... (step 2a)
            do set an edge between x and y in G, Y_x ← Y_x ∪ {y}
    done
    while |Y_x| < M_2 and Ȳ'_x ≠ φ
        y ← argmax overlap(x, y_k), Ȳ'_x ← Ȳ'_x \ {y}
                y_k∈Ȳ'_x
        if overlap(x, y)> 0 AND cooc(x, y)> 0 ..... (step 2b)
            do set an edge between x and y in G, Y_x ← Y_x ∪ {y}
    done
done
```

Fig. 4. Detailed Algorithm of threshold tuning used at the Yokohama Triennale 2005

5.1 System Flow

This system includes online and offline procedures. In the offline procedure, we tune four parameters: T_{ov}, T_{co}, M_1, and M_2. For them, T_{ov} and T_{co} are thresholds to invent edges by the overlap coefficient and matching coefficient, and M_1 and M_2 are the minimum numbers of edges for each node. We sample 1000 pairs of artists as training data: 146 positive examples and 854 negative examples. We change the values of parameters, classify every pair of artists into positive and negative using the parameters, and find the optimal values where the F-value is maximized: $T_{ov} = 0.82$, $T_{co} = 20$, $M_1 = 5$ and $M_2 = 1$. We try different settings for the four parameters; T_{ov} is changed from 0 to 1 at every 0.01, and T_{co} is changed from 0 to 60 in steps of 5, M_1 and M_2 are incremented from 0 to 5[6].

For the online procedure, a list of artists' names are given as input; the output is a social network of artists. Three steps exist: making queries, Google search, and network construction. First, we make queries for each pair of names. Then we put them into the Google search engine to obtain the hit counts. Finally, we construct a social network after tuning the parameters.

A detailed algorithm to generate a social network is shown in Fig. 4. Edges are added using an objective criterion (in step 1): An edge is added between the nodes if

[6] We might use more sophisticated algorithms such as hill-climbing searches. However, we do not specifically examine the optimization method in this paper. For that reason, we employed a simple (but reliable) approach.

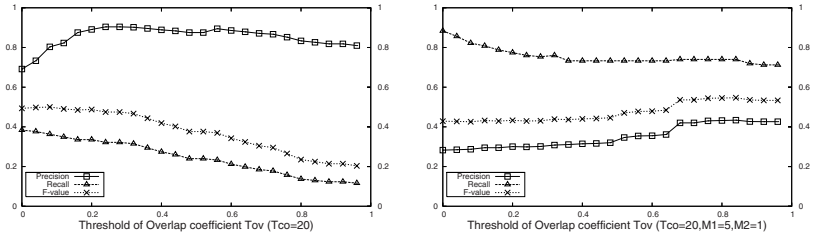
Table 3. Maximized precision, recall, and F-value using the precedent approach

Cases	T_{ov}	T_{co}	Precision	Recall	F-value	Extracted number*	Correct number*
case (a)	0.24	30	92.9%	26.7%	0.41	42 (42, 0, 0)	39 (39, 0, 0)
case (b)	0	0	14.6%	100%	0.25	1000 (1000, 0, 0)	146 (146, 0, 0)
case (c)	0.05	20	76.4%	37.7%	0.50	72 (72, 0, 0)	55 (55, 0, 0)

∗: Numbers in brackets are numbers of edges invented in step 1, step 2a, and step 2b.

Table 4. Maximized precision, recall, and F-value using the proposed approach

Cases	T_{ov}	T_{co}	M_1	M_2	Precision	Recall	F-value	Extracted number	Correct number
case (a)'	0.24	30	3	2	34.4%	65.1%	0.45	277 (42, 227, 8)	95 (39, 54, 2)
case (b)'	0	0	0	0	14.6%	100%	0.25	1000 (1000, 0, 0)	146 (146, 0, 0)
case (c)'	0.05	20	1	0	55.4%	49.3%	0.52	130 (72, 58, 0)	72 (55, 17, 0)
case (d)	0.82	20	5	1	43.4%	74.0%	0.55	249 (23, 212, 14)	108 (19, 84, 5)



(a) Existing algorithm (b) Proposed algorithm

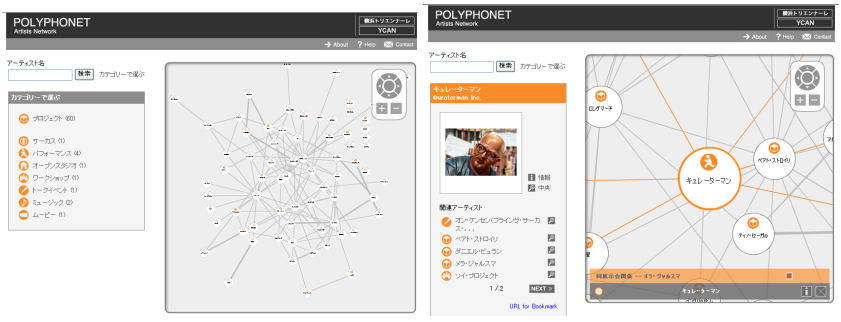
Fig. 5. Precision, recall and F-value for different T_{ov}

the overlap coefficient and the matching coefficient are both over the thresholds. Then subjective criteria are used to add edges (in step 2): If node x has less then M_1 edges, we choose nodes that have the strongest relations with node x. Node x is connected to the other nodes until the number of edges reaches M_1 (in step 2a). After that, if node x has no M_2 edges yet, we add edges in descending order of overlap coefficient (in step b).

Although the algorithm is highly customized for dealing with web information, the concept is simple. We use the objective criteria (using T_{ov} and T_{co}) first, and the subjective criteria (using M_1 and M_2) subsequently. It is important to combine multiple criteria to infer the relations among artists correctly from the available web information.

5.2 Evaluation

The existing approach by Mika and Matsuo generates a social network based on an objective criterion with a predefined threshold. It corresponds to the case where $M_1 = 0$ and $M_2 = 0$ in our algorithm. To compare the existing method with our method, we tune T_{ov} and T_{co} so that precision, recall, and F-value are maximized, respectively. The results are shown in Table 3. The maximal recall is 100% by setting T_{ov} and T_{co} as zero (which means the algorithm recognizes all the pairs having a relation), which yields precision as low as 14.6%. Conversely, the maximal precision is 92.9% when the recall is as low as 26.7%. The precision is 76.4% and the recall is 37.7% when the F-value is maximized.



(a) The whole network. (b) Centering artist *Curatorman*.

Fig. 6. System Interface for Yokohama Triennale 2005

Our algorithm can achieve better performance in either case. Table 4 shows results of our algorithm using four parameters. Even if we set T_{ov} and T_{co} as identical to those in Table 3, we can achieve better results by adjusting M_1 and M_2. The most balanced parameters achieve F-value of 0.55, which is more than 0.05 points better than the proposed algorithm. Figure 5 shows a notable difference: the proposed algorithm produces high recall while maintaining modest precision. It is useful when the purpose is to promote navigation and communication using a social network.

In this section, we emphasize detection of relationships using only the hit number of search engine. This is treated as a first step in the Yokohama Triennale system. As second step, we further identify concrete relation types from web pages retrieved by names of artists who are considered as related; we also filter out noisy edges to improve the precision. Details about the relation type identification are available from [16].

5.3 Navigation Site for Yokohama Triennale

Our system was put into operation on the official support site for Yokohama Triennale 2005 (http://mknet.polypho.net/tricosup/) to provide an overview of the artists (133 artists with 71 projects) along with informational navigation for users. At exhibitions, it is usual for participants to enjoy and evaluate each work separately. However, our supposition was that if participants knew the background and relations of the artists, they might enjoy the event more. For that purpose, the system provided relations of artists and evidential web pages for users.

The system interface is shown in Fig. 6. It was implemented using Flash display software to facilitate interactive navigation. The system provides a retrieval function. Information about the artist is shown on the left side if a user clicks a node. In addition, the edges from the nodes are highlighted in the right-side network. The user can proceed to view the neighboring artists' information sequentially, and can also jump to the web pages that show evidence of the relation.

6 General Extraction of a Social Network Using a Search Engine

Based on the two case studies described in the preceding sections, this section presents and explains an architecture to support general social network extraction from the web using a search engine. The types of social networks depend on their purpose [22]. A "good" social network should represent a target domain most appropriately.

We consider that social network extraction is generally written as

$$f(\mathbb{S}_r(X, Y), \Theta) \rightarrow \{0, 1\} \tag{1}$$

where $\mathbb{S}_r(X, Y)$ is an m-dimensional vector space $(S_r^{(1)}(X, Y), S_r^{(2)}(X, Y), \ldots, S_r^{(m)}(X, Y))$ to represent various measures for X and Y in relation r. For example, $S_r^{(i)}(X, Y)$ can be either $n_{X \cap Y}$ (matching coefficient), $n_{X \cap Y}/n_{X \cup Y}$ (Jaccard coefficient), or $n_{X \cap Y}/min(n_X, n_Y)$ (overlap coefficient). It can possibly be a score function based on sentences including both mentions of X and Y (as the algorithm in Section 4). The parameter Θ is an n-dimensional vector space $(\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(n)})$. For example, Θ can be as a combination of T_{ov}, T_{co}, M_1, and M_2 as the algorithm in Section 5. The function f determines whether an edge should be invented or not based on multiple measures and parameters.

A social network should represent the particular relations of entities depending on purposes. Therefore, function f should not always be the same. We must have a method to infer an appropriate function f, thus the algorithm inevitably consists of an offline module and an online module. Function f is learned from the training examples and provides good classification to other examples.

In the online phase, it is important to extract a social network from the web in an efficient manner. We must consider how to use a search engine better and how to process web documents efficiently and correctly. Generally, the procedure consists of three steps:

Making queries. Two entities are used to generate a query. Basically, we put a query X AND Y to a search engine. In this paper, we add relation keywords to extract a particular type of relation efficiently. A combination of multiple queries might improve the result, as explained in Section 4. Entity disambiguation is another important issue that has already been addressed in several studies [3,4].

Google search. We put the queries into a search engine. Sometimes the counts are used to infer relational strength. In other cases, we download some documents (or snippets) and investigate the mentions of X and Y. A good combination of Google counts and text analysis would make the search more efficient and scalable, as discussed in [16].

Network construction. We use Google counts and downloaded text as evidence to construct a social network. The value of function f is calculated and the existence of an edge is determined. Usually, the obtained social network is visualized and reviewed. Sometimes we must change settings of the algorithm (or increase the training data) and repeat the entire process to improve the quality.

Previous studies have emphasized how to calculate the strength of two names on the Web in the **Google search** step, simply using X AND Y as query and construct networks based on objective criteria. Our method, i.e., *relation identification* and *threshold tuning*

Table 5. Centrality of firms in the extracted social network

(a) Eigenvector centrality.

Rank	Name	Value
1	Matsushita	0.366
2	Hitachi	0.351
3	NEC	0.289
4	Fujitsu	0.275
5	Toshiba	0.263
6	Rakuten	0.257
7	Just System	0.241
8	KDDI	0.208
9	Tokyo Electric	0.207
10	Seiko Epson	0.204

(b) Betweenness centrality.

Rank	Name	Value
1	Matsushita	168.981
2	IBM	149.192
3	NEC	144.675
4	Hitachi	136.978
5	Toshiba	113.239
6	Rakuten	109.887
7	Just System	77.175
8	Livedoor	74.141
9	CISCO	64.558
10	Fujitsu	56.081

are proposed for **Making queries** and **Network construction** steps respectively for complex and inhomogeneous communities. All of these methods are combined into our architecture of general extraction of social networks for various entities.

The obtained network is useful for Semantic Web studies in several ways. For example (inspired by [2]), we can use a social network of artists for detecting COI among artists when they make evaluations and comments on others' work. We might find a cluster of firms and characterize a firm by its cluster. Business experts often make such inferences based on firm relations and firm groups, so the firm network might enhance inferential abilities in the business domain. As a related work, F. Gandon et al. built a Semantic Web server that maintains annotations about the industrial organization of Telecom Valley to partnerships and collaboration [8].

We present a prototypical example of applications using a social network of firms. We calculate the *centrality*, which is a measure of the structural importance of a node in the network, for each firm on the extracted social network (on alliance relations). Table 5(a) shows the top ten firms by eigenvector centrality. These firms have remained large and reliable corporations in Japan for decades. Table 5(b) shows the top ten by betweenness centrality. Interestingly, IBM, Livedoor, and Cisco are on the list. These firms might bridge two or more clusters of firms: IBM and Cisco are United States firms and form alliances with firms in multiple clusters; Livedoor is famous for its aggressive M & A strategy in Japan. Such information can only be inferred after extracting a social network. There seem to be many potential applications that can make use of social networks in the Semantic Web.

7 Conclusion

This paper describes methods of extracting various social networks from the web. To date, numerous studies have addressed the researcher domain to estimate extraction methods. It is an important test-bed. Nevertheless, the next step must be taken to depart from the domain of researchers. This paper steps further to show that researcher networks might be an easy domain for social network extraction from the web. Our method, equipped with relation identification and threshold tuning that specifically

focus on complex and inhomogeneous communities respectively, can extract other types of social networks: those of firms and artists. We show various evaluations of the methods along with discussions of the application of social network in the context of the Semantic Web. The proposed architecture toward general extraction of social networks, which bundles these different extraction methods, will enable us to extract various social networks from available information on the web.

In addition to some direct applications of social networks, we believe that a network point of view is important for knowledge integration and articulation and for (lightweight) ontology emergence. The combination of social networks and ontology emergence might prepare a fertile ground for Semantic Web research.

Acknowledgements. This research was supported by the New Energy and Industrial Technology Development Organization (NEDO) as project ID 04A11502a.

References

1. L. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211–230, 2003.
2. B. Aleman-Meza, M. Nagarajan, C. Ramakrishnan, A. Sheth, I. Arpinar, L. Ding, P. Kolari, A. Joshi, and Tim Finin. Semantic analytics on social networks: Experiences in addressing the problem of conflict of interest detection. In *Proc. WWW2006*, 2006.
3. N. Aswani, K. Bontcheva, and H. Cunningham. Mining information for instance unification. In *Proc. ISWC2006*, 2006.
4. R. Bekkerman and A. McCallum. Disambiguating web appearances of people in a social network. In *Proc. WWW2005*, 2005.
5. H. Chen, M. Lin, and Y. Wei. Novel association measures using web search with double checking. In *Proc. COLING-ACL2006*, pages 1009–1016, 2006.
6. A. Culotta, R. Bekkerman, and A. McCallum. Extracting social networks and contact information from email and the web. In *CEAS-1*, 2004.
7. T. Finin, L. Ding, and L. Zou. Social networking on the semantic web. *The Learning Organization*, 2005.
8. F. Gandon, O. Corby, A. Giboin, N. Gronnier, and C. Guigard. Graph-based inferences in a semantic web server for the cartography of competencies in a telecom valley. In *Proc. ISWC2005*, 2005.
9. S. Ghita, W. Nejdl, and R. Paiu. Semantically rich recommendations in social networks for sharing, exchanging and ranking semantic context. In *Proc. ISWC2005*, 2005.
10. J. Golbeck and B. Parsia. Trust network-based filtering of aggregated claims. *International Journal of Metadata, Semantics and Ontologies*, 2006.
11. H. Kautz, B. Selman, and M. Shah. The hidden Web. *AI magazine*, 18(2):27–35, 1997.
12. F. Keller and M. Lapata. Using the web to obtain frequencies for unseen bigrams. *Computational Linguistics*, 29:459–484, 2003.
13. C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. The MIT Press, London, 2002.
14. P. Marsden. Measuring tie strength. *Social Forces*, 63:482–501, 1984.
15. Y. Matsuo, M. Hamasaki, H. Takeda, J. Mori, D. Bollegala, Y. Nakamura, T. Nishimura, K. Hasida, and M. Ishizuka. Spinning multiple social networks for semantic web. In *Proc. AAAI-06*, 2006.
16. Y. Matsuo, J. Mori, M. Hamasaki, H. Takeda, T. Nishimura, K. Hasida, and M. Ishizuka. POLYPHONET: An advanced social network extraction system. In *Proc. WWW2006*, 2006.

17. P. Mika. Flink: Semantic web technology for the extraction and analysis of social networks. *Journal of Web Semantics*, 3(2), 2005.
18. P. Mika. Ontologies are us: A unified model of social networks and semantics. In *Proc. ISWC2005*, 2005.
19. J. Mori, M. Ishizuka, T. Sugiyama, and Y. Matsuo. Real-world oriented information sharing using social networks. In *Proc. ACM GROUP2005*, 2005.
20. S. Oyama, T. Kokubo, and T. Ishida. Domain-specific web search with keyword spices. *IEEE TKDE*, 16(1):17–27, 2004.
21. G. Ramakrishnan, S. Chakrabarti, D. Paranjpe, and P. Bhattacharyya. Is question answering an acquired skill? In *Proc. WWW2004*, 2004.
22. J. Scott. *Social Network Analysis: A Handbook (2nd ed.)*. SAGE publications, 2000.
23. S. Staab, P. Domingos, P. Mika, J. Golbeck, L. Ding, T. Finin, A. Joshi, A. Nowak, and R. Vallacher. Social networks applied. *IEEE Intelligent Systems*, pages 80–93, 2005.
24. P. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proc. ECML2001*, pages 491–502, 2001.