Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 2010 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

2010

Patent Classification Using Ontology-Based Patent Network Analysis

Meng-Jung Shih
National Chiao Tung University, mjshih@gmail.com

Duen-Ren Liu
National Chiao Tung University, dliu@iim.nctu.edu.tw

Recommended Citation

Shih, Meng-Jung and Liu, Duen-Ren, "Patent Classification Using Ontology-Based Patent Network Analysis" (2010). *PACIS 2010 Proceedings*. 95.
http://aisel.aisnet.org/pacis2010/95

PATENT CLASSIFICATION USING ONTOLOGY-BASED PATENT NETWORK ANALYSIS

Meng-Jung Shih, Institute of Information Management, National Chiao Tung University, Hsinchu, Taiwan, mjshih@gmail.com

Duen-Ren Liu, Institute of Information Management, National Chiao Tung University, Hsinchu, Taiwan, dliu@iim.nctu.edu.tw

Abstract

Patent management is increasingly important for organizations to sustain their competitive advantage. The classification of patents is essential for patent management and industrial analysis. In this study, we propose a novel patent network-based classification method to analyze query patents and predict their classes. The proposed patent network, which contains various types of nodes that represent different features extracted from patent documents, is constructed based on the relationship metrics derived from patent metadata. The novel approach analyzes reachable nodes in the patent ontology network to calculate their relevance to query patents, after which it uses the modified k-nearest neighbor classifier to classify query patents. We evaluate the performance of the proposed approach on a test dataset of patent documents obtained from the United States Patent and Trademark Office (USPTO), and compare it with the performance of the three conventional methods. The results demonstrate that the proposed patent network-based approach outperforms the conventional approaches.

Keywords: Patent classification, Patent network analysis, k-nearest neighbor, Patent ontology network.

1 INTRODUCTION

Patent are valuable intellectual properties of organizations and require effective management to sustain an organization's competitive advantage. Because of developments in technology, the number of patents has increased rapidly in recent years. How to manage the constantly growing volume of patents is thus becoming an important issue. Patent classification is an important part of patent management. However, it is usually performed manually, such that categorizing new patent documents correctly is a laborious process. Hence, there is a pressing need for an effective patent classification approach.

Basically, patent classification can be regarded as a text categorization problem that involves assigning a class to a patent document. Most existing studies consider information content to classify patent documents, and several classification algorithms have been developed based on different content features (e.g., Larkey, 1999; Fall et al., 2003 and 2004; Trappey et al., 2006; Loh et al., 2006; Kim & Choi 2007; Cong & Tong, 2008). Some approaches utilize citation relationships to improve the performance of patent classification (Lai & Wu, 2005; Li, et al., 2007); while others employ patent metadata, such as the inventor's name, and thereby achieve improvements in the classification performance (Richter & MacFarlane, 2005).

In this study, we propose a novel patent network-based classification method to yield better prediction accuracy. Patent metadata (ontology) provides rich information that can be used to infer possible relationships between patent documents. The proposed method utilizes patent metadata to construct an ontology-based patent network for class prediction. Patent documents and ontology (e.g. inventor and patent class) form as patent nodes and ontological nodes of the constructed network. In addition, semantic relationships between patents and ontological nodes are derived to link the nodes of the patent network. Based on the network, the neighboring patents and ontological nodes of a query patent are identified to predict the class of the query patent. We conduct experiments to assess the performance of the proposed approach with that of the conventional approaches on real-world patent data. The results show that the proposed approach outperforms conventional patent classification methods.

The remainder of this paper is organized as follows. The next section contains a review of the literature on patent classification methods and ontology-based network analysis. In Section 3, we describe the proposed patent network-based classification methodology. We discuss the experiment results in Section 4, and then summarize our conclusions in Section 5.

2 LITERATURE REVIEW

2.1 Patent Classification

Patent classification schemes categorize patent documents. In recent years, a considerable number of such schemes have been proposed (e.g., Kim & Choi, 2007; Kohonen, et al., 2000; Lai & Wu, 2005; Larkey, 1999; Richter & MacFarlane, 2005; Cong & Tong, 2008; Cong & Loh, 2010; Trappey, et al., 2006). The features extracted from patent documents for classification purposes can be divided into three types: content features, citation information and metadata.

2.1.1 Content-based patent classification

Since patent classification is formulated as a text categorization problem that involves assigning a patent document to the correct class, most studies only consider patent content information to address the problem (e.g., Loh, et al., 2006). In content-based patent classification approaches, the content of patent documents is represented by vectors of term weights. The similarity of two patent documents is defined as the cosine value of their term vectors (Yang, 1994). The most popular term weighting function is term frequency / inverse document frequency (*tfidf*), developed by Salton and Buckley (Salton & Buckley, 1988).

Based on the similarity of patent documents, the *kNN* classifier selects the *k*-nearest neighbors of a query patent to predict the class of the patent based on majority vote. The class that most of neighboring patents belong to is chosen as the class of the query patent.

Instead of using the full text of a patent document as the basis for classification, some approaches classify patent documents by considering normative sections, such as the abstract, background, and results (Kim & Choi, 2007; Fall, 2003, 2004; Larkey, 1999; Cong & Tong, 2008; Loh, et al., 2006; Trappey, et al., 2006). These studies regard the patent document's abstract as the most informative feature (Larkey, 1999; Loh, et al., 2006)..

2.1.2 Citation-based patent classification

In real-world applications, patent documents are linked through citations that imply the connections and relationships between the citer and the cited. Approaches that utilize citations have been proposed (Lai & Wu, 2005; Li et al., 2007). The co-citation approach (Lai & Wu, 2005) classifies a query patent according to the majority vote of the classes of its cited patents. For example, suppose a query patent cites five documents in the basic patent set. If three of the cited patents belong to class *C1* and the other two belong to class *C2,* the query patent will be assigned to class *C1.* Note that the co-citation approach, uses the grouping result of patents, which are clustered according to the co-citation frequency and linkage strength of each pair of basic patents, as the classes, rather than the well-known UPCs (United States Patent Classification) or IPCs (International Patent Classification).

These studies demonstrate that citation-based patent classification performs better than content-based classification. In our work, we also consider the citation relationships between patent documents when constructing the patent ontology network.

2.1.3 Metadata-based patent classification

Metadata is defined as "information that describes data". The metadata in a patent document, such as inventors' names and assignees' names, may be correlated with the document's content and can be used for classification purposes. Richter & MacFarlane (2005) showed that patent classification based on a document's metadata can improve the accuracy of the results. Their approach uses metadata, such as the inventor's name, the applicant's name and the IPC code to help classify commercial intellectual property. Because the approach considers text, inventor and IPC metadata simultaneously, it yields a better classification result. Patent documents are mapped into vectors of terms, inventors' names and IPCs. For the text, the weights of terms are calculated by the *tfidf* approach (Salton and Buckley, 1988); the weight of each inventor is calculated as $\sqrt{1/\#\text{inv}}$, where #inv is the total number of inventors of the patent; and the weight of each IPC code is calculated as $\sqrt{1/(\#\text{ipc}+1)}$, where #ipc is the number of IPC code assigned to the patent. Note that the primary IPC is weighted twice as high as other IPC assigned to the patent. After compiling the vectors, the similarity between two patent documents can be calculated. The *kNN* classifier is then used to identify the class of the query patent based on the similarity (cosine value) of patent documents.

One limitation of the above method is that it only works well when the inventors of a query patent also exist in the training set. The method does not utilize indirect relationships to help classify patents developed by new inventors who are not included in the training set. In contrast, our method constructs a patent ontology network; thus, indirect relationships can be used to classify patent documents more flexibly and accurately.

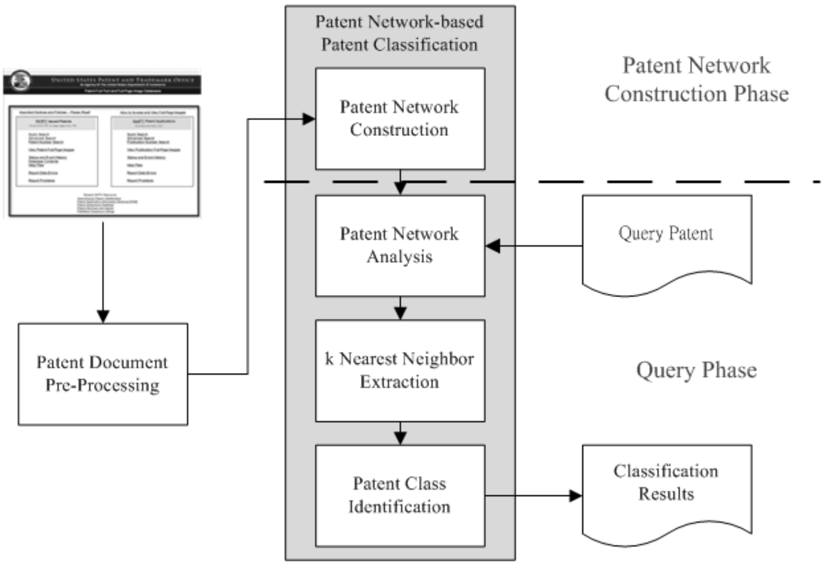

Figure 1. The patent network-based patent classification process

2.2 Ontology-Based Network Analysis

O'Hara, et al. (2002) developed an ontology-based network analysis method to examine ontology-based social networks that help identify communities of practice (i.e., groups of individuals interested in a particular job, procedure, or work domain). The ontology-based social network is formed by object instances (e.g. person, paper, conference) and semantic relationships (e.g. authorOf, attended) between instances. The rationale behind the method is that the relevance values of nodes increase with the number of semantic paths leading to the object of interest. The instances and their relationships in the ontology network are analyzed by a breadth-first, spreading-activation search algorithm that traverses the semantic relations between instances. In this approach, the relationships and their weights are selected manually and pre-defined.

We modify the above method and use it for patent network analysis to measure the relevance of a query patent and the nodes in a patent ontology network. The weights of relationships are generated automatically according to the semantic relevance of two nodes. Then, the *k* nodes with the highest relevance to the query patent are used to predict the class of the patent.

3 PATENT NETWORK-BASED PATENT CLASSIFICATION

The proposed patent network-based classification approach is implemented in two phases: 1) patent network construction; and 2) patent network analysis, which includes *k* nearest neighbor extraction and patent class identification.

3.1 Patent Ontology Network Construction

Figure 1 shows the patent network-based patent classification process. The first step involves building a patent ontology network. The relations between instances (nodes) are identified to construct the network. The weights of all the relationships among nodes are derived by the functions described in this section. Relationships (connections) of zero degree are dropped and the network is trimmed to form the final patent ontology network for classification. The proposed patent ontology network contains four types of instances (nodes) and eight types of relations (edges). The node types are patent, UPC class, inventor, and assignee (e.g., a research institute). The weights of the relationships are calculated by the functions listed in Table 1.

Table 1. The relationship metrics used in the patent ontology network

Relationship Weights	Patent p_2	Class c_2	Inventor v_2	Assignee a
Patent p_1	$R_{PP}(p_1, p_2)$	$R_{PC} = \begin{cases} 1: p_1 \in c_2 \\ 0: p_1 \notin c_2 \end{cases}$	$R_{PV} = \begin{cases} 1: p_1 \text{ invented by } v_2 \\ 0: \text{not related} \end{cases}$	$R_{PA} = \begin{cases} 1: p_1 \text{belonging to } a \\ 0: \text{not related} \end{cases}$
Class c_1		N/A	$R_{CI}(v2, c1)$	$R_{CA}(c_1, a)$
Inventor v_1			$R_{II}(v1, v2)$	$R_{IA} = \begin{cases} 1: v_1 belonging\ to\ a \\ 0: not\ related \end{cases}$

$R_{PP}(p_1, p_2)$ denotes the relationship between two patents. Both citations and co-citations are considered active relations between two patents, as shown in Eq. 1:

$$\begin{cases} R_{PP}(p_1, p_2) = \mathrm{w_{cite}} \times Cite(p_1, p_2) + \mathrm{w_{co\text{-}cite}} \times CoCite(p_1, p_2) \\ \mathrm{w_{cite}} + \mathrm{w_{co\text{-}cite}} = 1 \end{cases}, \quad (1)$$

where $Cite(p_1,p_2)$ is the citation relation between p_1 and p_2 defined as

$$Cite(p_1, p_2) = \begin{cases} 1, \text{if the citation exists (either } p_1 \text{ cites } p_2 \text{ or } p_2 \text{ cites } p_1) \\ 0, \text{otherwise.} \end{cases}$$

and $CoCite(p_1,p_2)$ is the degree of co-citing between p_1 and p_2 defined as

$$CoCite(p_1, p_2) = \frac{|CitedBy(p_1) \cap CitedBy(p_2)|}{|CitedBy(p_1) \cup CitedBy(p_2)|},$$

where $CitedBy(p_1)$ and $CitedBy(p_2)$ are the sets of patents cited by p_1 and p_2, respectively.

$R_{II}(v_1, v_2)$ represents the degree of patents that belong to two inventors and is defined as follows:

$$R_{II}(v_1, v_2) = \frac{|\mathrm{Patents}(v_1) \cap \mathrm{Patents}(v_2)|}{|\mathrm{Patents}(v_1) \cup \mathrm{Patents}(v_2)|}, \quad (2)$$

where $Patents(v_1)$ and $Patents(v_2)$ are the sets of patents belonging to v_1 and v_2, respectively.

$R_{CI}(v_2, c_1)$ represents the ratio of patents belonging to a specific inventor v_2 to the number of patents in a patent class c_1, and is defined as follows:

$$R_{CI}(v_2, c_1) = \frac{|\mathrm{Patents}(v_2)|}{|\mathrm{Patents}(c_1)|}, \quad (3)$$

where $Patents(c_1)$ is the set of patents belonging to class c_1.

$R_{CA}(c_1, a)$ represents the importance and maturity of a technology of assignee a in a specific technology field, i.e. class c_1 , as shown in Eq. 4:

$$R_{CA}(c_1, a) = \frac{\sum_{\mathrm{p_i} \in \mathrm{Patents}(a) \cap \mathrm{Patents}(c_1)} NumCitations(p_i, a, c_1)}{\sum_{p_j \in \mathrm{Patents}(c_1)} NumCitations(p_j, c_1)}, \quad (4)$$

where $NumCitations(p_i, a, c_1)$ is the number of patents in class c_1 that cite assignee a's patent p_i ; and $NumCitations(p_j, c_1)$ is the number of patents in class c_1 that cite patent p_j.

Figure 2 shows an example of a patent ontology network that includes the four types of nodes, i.e., patent, class, inventor and assignee. The weights of relations are calculated using the equations listed in Table 1.

The patent ontology network is a base map for classifying unclassified patents. In the next sub-section, we describe the classification process based on patent network analysis. Classifying a patent and assigning it to the most suitable class involves three steps: patent network analysis, k-nearest neighbor extraction and patent class identification.

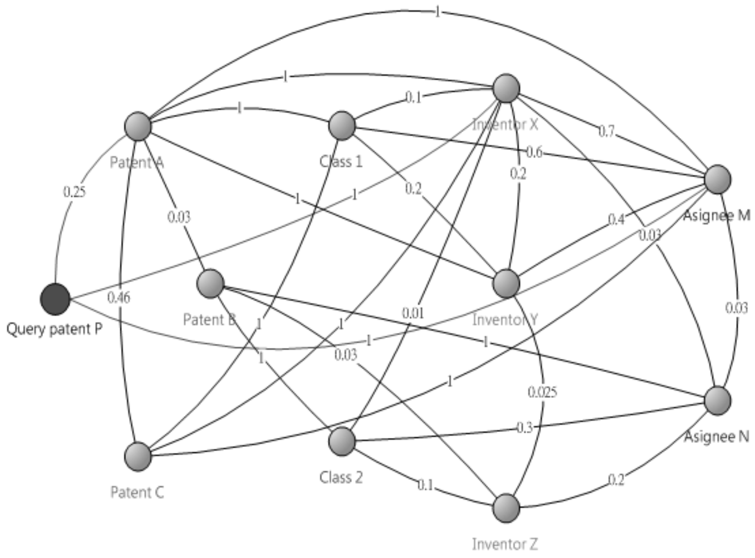

Figure 2. An example of patent ontology network

3.2 Patent Network Analysis

To classify a patent document, we first search the patent ontology network to find patent nodes, inventor nodes and assignee nodes that have connections with the query patent. For example, in the network in Figure 2, *X* is the inventor of query patent *P* and the assignee is *M*. Patent *P* also has citation relationships with other patents. These connections are therefore evaluated to derive their respective weights using the equations listed in Table 1.

After determining all the connections and weights between the query patent and the nodes in the patent ontology network, we calculate the relevance of the query patent to each node in the patent ontology network. The algorithm used for patent network analysis is a modification of the ontology-based network analysis algorithm developed by O’Hara et al. (2002) for identifying an individual’s communities of practice. Our algorithm calculates the weights of the nodes and their relations to derive correlations in the metadata. More specifically, it implements a breadth-first, spreading-activation search and traverses the relations between the nodes until it reaches a link threshold, which is the maximum number of consecutive links between nodes that can be traversed. Appendix A details the steps of the patent network analysis algorithm.

3.3 K-Nearest Neighbor Extraction

After calculating the relevance of the query patent document to the nodes in the patent ontology network, the *k* nodes with highest relevance scores to the query patent document are extracted and used to identify the most appropriate class for a patent.

3.4 Patent Class Identification

Let S_q be the set of nodes identified in the step of *k*-nearest neighbor extraction. In this step, the nodes in S_q are used to determine the class of the query patent *q*. Unlike the classical *kNN* method, which can only find neighboring nodes of the same type, the proposed method can find *k* nodes of various types by using the results of patent network analysis. We only use patent and class nodes to calculate the scores of candidate classes because they are more suitable for interpreting patent classes. For “patent” nodes, the more relevant a patent node *q* is to the query patent, the greater the likelihood that the query patent belongs to the class of that patent node. In addition, for “class” nodes, the more relevant a class node *c* is to the query patent, the greater the likelihood that the query patent belongs to the class of that node. We denote the set of identified patent nodes and the set of identified class nodes as S_q^{P} and S_q^{C}, respectively. Note that $S_q^{\mathrm{P}}, S_q^{C} \subset S_q$.

The next step evaluates the predicted scores of candidate classes, which are selected from the identified patent nodes and class nodes. The predicted scores $F_{q,c}$ for a given query patent *q* and a given class *c* are calculated as follows:

$$F_{q,c} = \sum_{d \in S_q^{\mathrm{P}}} w_d \ B_{d,c}^{\mathrm{P}} + \sum_{d \in S_q^{\mathrm{C}}} w_d \ B_{d,c}^{\mathrm{C}} \tag{5}$$

where w_d denotes the weight of node *d*; and $B_{d,c}^{\mathrm{P}}$ and $B_{d,c}^{\mathrm{C}}$ are defined as follows:

$$B_{d,c}^{\mathrm{P}} = \begin{cases} 1, \text{ if node } d \text{ represents a patent belonging to class } c \\ 0, \text{otherwise} \end{cases},$$

$$B_{d,c}^{\mathrm{C}} = \begin{cases} 1, \text{if node } d \text{ represents class } c \\ 0, \text{otherwise} \end{cases}$$

After obtaining all the predicted scores of classes in *C*, the class with highest score is taken as the class of the query patent.

4 EXPERIMENT AND EVALUATION

4.1 Data Collection

To evaluate the performance of the proposed approach, we conducted experiments on the collection of patent documents obtained from USPTO. The dataset contains 1,231 patent documents divided into 5 UPCs, as shown in Table 2. We use a patent’s UPC to denote its class.

The documents in the database records are divided into two sets: 1) a training set (70% of the collected dataset) containing the patent documents whose classes are known; and 2) a test set (30% of the collected dataset) containing patent documents whose classes are to be determined.

Table 2. The collected patent dataset.

Class NO.	Class Title	Data Instances
29	Metal Working	246
257	Active Solid-State Devices	273
324	Electricity: Measuring and Testing	221
438	Semiconductor Device Manufacturing: Process	286
709	Electrical Computers and Digital Processing Systems: Multicomputer Data Transferring	205

4.2 Evaluation Metrics

We used standard classification performance metrics, namely, the *accuracy rate*, *precision rate*, *recall rate*, and *F-measure* (Salton & Buckley, 1988; Van Rijsbergen, 1979), to evaluate the performance of the classifiers. These metrics have been widely used in information retrieval and machine learning studies.

Classification accuracy was used to assess the overall performance, as shown in Eq. 6:

$$Accuracy = \frac{\text{\# of correctly classified patents}}{\text{total \# of patents}} \tag{6}$$

Precision, *recall* and *F-measure* were used to assess the classification performance. For instances of class *i*:

$$Precision\ (i) = \frac{\text{\# of correctly identified patents for class } i}{\text{total \# of patents idientified as class } i} \tag{7}$$

$$Recall\ (i) = \frac{\text{\# of correctly identified patents for class } i}{\text{total \# of patents in class } i} \tag{8}$$

Finally, to obtain a single performance measure, we used a simple *F-measure* to balance the *precision* and *recall scores,* as shown in Eq. 9:

$$F - measure\ (i) = \frac{2 \times \text{precision}(i) \times \text{recall}(i)}{\text{presioion}(i) + \text{recall}(i)} \tag{9}$$

Precision and *recall* evaluate whether a classification is successful. If both parameters yield high scores in a classification experiment, the approach's performance is considered ideal. However, precision and recall are usually in conflict with each other, so the *F-measure* is used to balance the two results.

4.3 Network-based Patent Classification

4.3.1 Link threshold of similarity calculation for k nearest neighbor

The number of links in the patent network to expand has a significant effect on the results. The *k*-nearest neighbor extraction step attempts to identify the nodes that are most similar to the query patent document within the boundary defined by the given link threshold. If we limit expansion to only one link, all identified nodes have a direct relation to the query patent document. However, as the number of links increases, the number of nodes that have an indirect link to the query patent will also increase.

Table 3 shows the performance of the patent network-based classification module under different link thresholds. The best performance is achieved when the link threshold = 3. Hence, we set the link threshold = 3 in the following experiments.

Table 3. The performance of the network-based classification module under different link thresholds

Link Threshold	Accuracy	Avg. precision	Avg. recall	Avg. F-measure
1	33.2	31.4	31.8	31.6
2	57.6	58.1	55.4	56.7
3	74.9	77.6	74.9	76.2
4	67.8	66.3	64.7	65.5

4.3.2 Types of Nodes in the Patent Ontology Network (link threshold= 3)

The types of nodes used in the patent ontology network also affect the results. We tried to find the best types via experiments. As shown in Table 4, the patent ontology network with four types of nodes, namely, patent, class, inventor and assignee nodes, yields the best performance.

Table 4. The performance of the network with different combinations of nodes

Node types used	Accuracy	Avg. precision	Avg. recall	Avg. F-measure
Patent / class /inventor	61.9	68.8	65.3	67.0
Patent / class / assignee	68.5	66.1	71.4	68.6
Patent / class/ inventor / assignee	74.9	77.6	74.9	76.2

4.4 Comparison of Different Patent Classification Methods

We compare four patent classification approaches: a content-based approach, a citation-based approach, a metadata-based approach and the proposed patent network-based patent classification methods. The content-based approach uses the similarity of content (title and abstract), and adopts the *kNN* classifier to predict the class of a query patent based on similarity measures of patents. The citation-based approach determines the class of a query patent according to the majority of classes of its cited patents. For metadata-based approach, the neighbors are chosen based on the similarities of the content (title and abstract), inventor and IPC. This approach also uses the *kNN* classifier to predict the class of a query patent. Note that our proposed patent network-based approach uses the relevance of nodes in the patent ontology network. A particular feature of the *kNN* classifier applied in the proposed patent network-based approach is that the neighbors can be of different types, such as patents and classes, whereas the other three methods only search for neighbors among patents.

Table 5 shows the performances of the compared patent classification approaches. The proposed patent network-based approach achieves the best performance in terms of accuracy (74.9%) and the F-score (76.2%). The second best approach, the metadata-based approach, considers the IPC when deciding the class of a query patent. The IPC denotes a kind of classification and may correlate with the UPC, which represents the class of a patent. Thus, it is not reasonable to consider IPC when making UPC class predictions. The Metadata-based (text + inventor + IPC) method may be affected by the correlation between IPC and UPC and thus yields a good result.

Table 5. The experiment results of the compared patent classification methods

Types of Patent Classification	Accuracy	Avg. precision	Avg. recall	Avg. F-measure
Content-based (Title + Abstract)	45.2	47.8	45.4	46.6
Citation-based (Co-citation)	57.6	54.2	62.8	58.2
Metadata-based (text+inventor+IPC)	71.3	75.6	68.7	72.0
Metadata-based (text+inventor)	52.6	71.6	56.5	63.2
Metadata-based (inventor)	41.1	62.4	46.2	53.1
Patent Network-based	74.9	77.6	74.9	76.2

5 CONCLUSION

In this paper, we have proposed a novel patent network-based classification approach to obtain better prediction results. The patent network-based method derives the weights of the relationships between different types of nodes in the patent network. Based on the patent network analysis, the classification result can be improved by considering the neighboring patent nodes and class nodes of a query patent in making class prediction. Our experiment results demonstrate that the proposed patent network-based approach outperforms the conventional approaches.

Acknowledgement

This research was supported in part by the National Science Council of the Taiwan under the grant NSC 96-2416-H-009-007-MY3.

References

Cong, H. and Tong, L. H. (2008). Grouping of TRIZ Inventive Principles to facilitate automatic patent classification. Expert Systems with Applications, 34, 788-795.

Cong, H. and Loh, H. T. (2010). Pattern-Oriented Associative Rule-Based Patent Classification. Expert Systems with Applications, 37(3), 2395-2404.

Fall, C. J., Torcsvari, A., Benzineb, K. and Karetka, G. (2003). Automated categorization in the International Patent Classification. In SIGIR Forum, 10-25.

Fall, C. J., Torcsvari, A., Benzineb, K. and Karetka, G. (2004). Automated categorization of German-language Patent Documents. Expert Systems with Applications, 26(2), 269-277.

O'Hara, K. Alani, H. and Shadbolt, N. (2002). Identifying Communities of Practice: Analysing Ontologies as Network to Support Community Recognition. In Proceeding Conference International Federation Information Processing, World Computer Congress.

Kim, J. H. and Choi, K. S. (2007). Patent document categorization based on semantic structural information. Information Processing and management, 43, 1200-1215.

Kohonen, T., Kaski, S., Lagus, K., Salojavi, J., Honkela, J., Paatetro, V., et al. (2000). Self organization of a massive document collection. IEEE Transactions on Neural Networks, 11(3), 574-585.

Lai, K. K., Wu, S. J. (2005). Using the Patent Co-citation Approach to Establish a New Patent Classification System. Information Processing and Management, 41, 313-330.

Larkey, L. S. (1999). A Patent Search and classification system. In Proceedings of the fourth ACM conference on Digital libraries, 179-183.

Li, X., Chen, H. C., Zhang, Z., Li, J. (2007). Automatic Patent Classification using Citation Network Information: An Experimental Study in Nanotechnology. In Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, 419-427.

Loh, H. T., He, C. and Shen, L. (2006). Automatic classification of patent documents for TRIZ users. World Patent Information, 28(1), 6-13.

Richter, G. and MacFarlane, A. (2005). The Impact of metadata on the accuracy of automated patent classification. World patent Information, 27(1), 13-26.

Salton, G. and Buckley, C. (1988). Term-Weighting Approaches in Automatic Text Retrieval. Information Process Management, 24(4), 323-328.

Trappey, A.J.C., Hsu, F. C., Trappey, C. V., Lin, C-I. (2006). Development of a patent document classification and search platform using a back-propagation network. Expert Systems with Applications, 31, 755-765.

Van Rijsbergen, C. J. (1979). *Information retrieval* (2nd ed.). London: Butterworths.

Yang, Y. (1994). Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorisation and Retrieval. In: Croft WB, van Rijsbergen CJ, editors. Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, 3-6 July. New York: ACM, 13-22.

Appendix A.

The algorithm of patent network analysis.

```
Initialize the weight of all nodes to 1
Create a relationship-array of relationships and weights
Set the query patent document as the active node
Mark the current node as unlocked and add it to a node-array
Loop to the maximum number of links to traverse
  Search for the current node in the node-array
  If found:
    Mark the node as locked
Set the node as the active node
    Get all nodes connected to the current node with a relationship in the relationship-array
    Loop to  the number of connected nodes
      If the node is not in the node-array (new node)
        Weight of node=initial weight + current node weight * weight of connecting relation
      Mark node as unlocked and add it to node-array
      If the node is already in the node-array
        Weight of node=node weight + current node weight * weight of connecting relation
    End loop
  If not found then exit
End loop
Relevance of a node = Weight of node / n
(n= the minimum number of links traversed to reach the node starting from the query node)
```