Neurocomputing 127 (2014) 200–205

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Interactive patent classification based on multi-classifier fusion and active learning

Xiaoyu Zhang*

Institute of Scientific and Technical Information of China, 15 Fuxing Road, Haidian District, Beijing 100038, China

ARTICLE INFO

Article history:
Received 7 June 2013
Received in revised form
22 July 2013
Accepted 6 August 2013
Communicated by Ran He
Available online 18 October 2013

Keywords:
Patent classification
Machine learning
Semi-supervised learning
Multi-classifier fusion
Active learning

ABSTRACT

Patent classification is of great importance to effective patent analysis. Traditional manual classification suffers from the problem of low efficiency and high expense. To address this issue, an interactive patent classification algorithm based on multi-classifier fusion and active learning is proposed in this paper, which comprises the construction and update of classification model. For model construction, a sub-classifier is trained for each class of the patents by means of support vector machine. Via multi-classifier fusion, the sub-classifiers are effectively combined to acquire enhanced classifiers, based on which the classification decision can be made. For model update, active learning is used to select the most informative patents for labeling, in which dynamic batch sampling is presented to cope with the problem of redundancy in traditional batch mode. Using dynamic certainty propagation, the selected patents become more informative for active learning. By iterating model construction and update, the classification performance can be gradually refined. The interactive classification algorithm is applied to both synthetic data and patents, and its effectiveness is demonstrated by the encouraging results.

1. Introduction

The patent, as a document accessible to the general public, is an important form of intellectual property containing rich structured content regarding technological innovations [1]. The analysis of patents is a widely used method to discover inventive activity and output over different fields, regions, and time, and reveal trends in science and technology [2,3]. As the basis for patent analysis, patent classification is indispensable for effective management of patents and in-depth exploration of valuable information. Traditionally, patents are classified manually by domain experts, which is not only time-consuming and labor-intensive but also in high demand for expertise. As a result, manual classification is inefficient and costly, especially for the large-scale patent dataset.

With the remarkable progress of computer science, machine learning is thriving and can be introduced to augment patent classification [4]. Machine learning is the computational process of extracting patterns in data and making predictions based on experience learned from these patterns [5,6]. Classic machine learning can be divided into two types, i.e. supervised learning and unsupervised learning. In classification problems, the former infers classifiers from labeled training data, while the latter finds hidden structure in unlabeled data. Recent researches indicate that unlabeled data, when used in conjunction with labeled data, can produce considerable improvement in learning performance. In such circumstances, semi-supervised learning has been widely studied as an effective combination between supervised learning and unsupervised learning, which makes use of both labeled and unlabeled data for training [7,8]. In patent classification, the cost associated with the labeling process may render a fully labeled training set infeasible, whereas acquisition of unlabeled data is relatively inexpensive. In this case, semi-supervised learning can be of great practical value.

On one hand, human participation should be reduced for the sake of efficiency; on the other hand, instructive human intuition cannot be entirely eliminated since it is essential for classification model training. In this paper, an interactive patent classification algorithm is proposed. Through the collaboration between human and machine, the patent classification performance can be effectively improved compared with traditional methods [4]. Using patents labeled by the user as training data, classification model is constructed for prediction of the unlabeled patents. Based on the existing classification model, more patents are labeled and incorporated as training data for model update. Through the iteration of model construction and update, the classification performance can be effectively improved. For model construction, patent classification is treated as a set of binary classification problems corresponding to different classes. Using multi-classifier fusion, the sub-classifiers are combined to obtain enhanced classifiers, based on which the classification results are achieved. For model update,

* Tel.: +86 10 58882016.
E-mail address: zhangxy@istic.ac.cn

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.08.013

active learning is adopted to cope with the issue where labeled patents used as training data are limited while unlabeled patents are abundant and easy to access [9–11]. As a form of semi-supervised learning, active learning can actively query the user for labels, so that both the precious labeled and abundant unlabeled patents are made full use of to achieve high classification performance. To further improve the effectiveness of active learning, dynamic certainty propagation algorithm under dynamic batch mode is presented to reduce redundancy during selective sampling. Various algorithms are compared on both synthetic data and patents.

2. Methods

In order to classify the patents accurately, the classification model should be constructed effectively based on the training dataset provided by the user. For further refinement of the existing classification model, more patents should be labeled and incorporated as update instructions from the user. The construction and update of classification model can be performed repeatedly, making the classification results more and more coherent to the user's decision.

2.1. Classification model construction

2.1.1. Sub-classifier construction

Under vector space model (VSM), a patent can be denoted as a vector $\boldsymbol{x}_n$ of the patent dataset $\boldsymbol{X}=\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}=\boldsymbol{U}\cup\boldsymbol{L}$, where $\boldsymbol{U}$ and $\boldsymbol{L}$ stand for the unlabeled and labeled dataset respectively, and N is the total number of patents.

In binary classification, for patent $\boldsymbol{x}_n \in \boldsymbol{X}\,(1 \le n \le N), y_n \in \{1, -1\}$ corresponds to its label. If $\boldsymbol{x}_n \in \boldsymbol{L}$, y_n is known; while if $\boldsymbol{x}_n \in \boldsymbol{U}$, y_n is unknown and thus need to be predicted by $f(\boldsymbol{x}_n)$, where f is a classifier.

Patent classification is, in most cases, a multi-class classification problem, which is an extension of binary classification. For patent $\boldsymbol{x}_n \in \boldsymbol{X}$, its label can be represented by an I-dimensional vector $\boldsymbol{y}_n=(y_{n1}, y_{n2}, \ldots, y_{nI})$, where $y_{ni} \in \{1, -1\}$ $(1 \le i \le I)$ corresponds to the label of the ith class, and I is the number of classes. In this case, for each class i, a sub-classifier f_i should be constructed, arriving at a set of I sub-classifiers.

Support vector machine is adopted for classifier construction [12,13]. The basic idea of SVM is to learn an optimal boundary (or hyper-plane) that separates the training examples with the maximal margin. Given a training dataset $\boldsymbol{T} \subset \boldsymbol{L}$, the corresponding SVM classifier can be represented as

$$f(\boldsymbol{x}) = \boldsymbol{w}\Phi(\boldsymbol{x}) + b, \tag{1}$$

where

$$\boldsymbol{w} = \sum_{\boldsymbol{x}_t \in \boldsymbol{T}} \alpha_t \Phi(\boldsymbol{x}_t), \tag{2}$$

and Φ: $\boldsymbol{X}\to\boldsymbol{F}$ is an implicit map from the original data space $\boldsymbol{X}$ to another (usually higher dimensional) feature space $\boldsymbol{F}$ via a Mercer kernel function.

For a patent $\boldsymbol{x}_n \in \boldsymbol{X}$, $f_i(\boldsymbol{x}_n)$ not only reveals the predicted label, but also indicates the certainty in the prediction. If $f_i(\boldsymbol{x}_n) > 0$, $\boldsymbol{x}_n$ is classified as a positive example of class i, and otherwise negative. The larger $|f_i(\boldsymbol{x}_n)|$ is, the more certain the classifier is in the classification.

2.1.2. Multi-classifier fusion

To boost classification performance, multi-classifier fusion is used for the combination of all the sub-classifiers. Then based on the fusion classifiers, classification decision is made. This process is illustrated in Fig. 1, in which F_i stands for the fusion classifier for class i, and D denotes the classification decision.

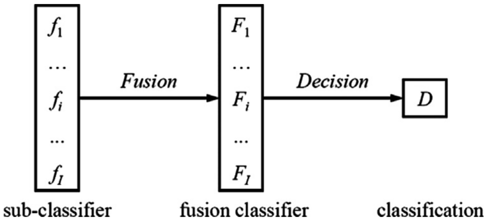

Fig. 1. Flowchart of classification model construction.

2.1.2.1. Non-fusion. Sub-classifiers can directly be used for classification without fusion. In this case, fusion classifier is just the corresponding sub-classifier

$$F_j(\boldsymbol{x}) = f_j(\boldsymbol{x}). \tag{3}$$

For all the simplicity, the non-fusion method is not an ideal choice. Because the correlation of different sub-classifiers is neglected, the classification performance can be seriously affected.

2.1.2.2. Linear fusion. Linear fusion combines the sub-classifiers according to their classification capacity. Each fusion classifier is represented by the weighted sum of sub-classifiers

$$F_j(\boldsymbol{x}) = \sum_{i=1}^{I} \delta(i,j)\mu_i f_i(\boldsymbol{x}), \tag{4}$$

where μ_i is the weight of sub-classifier f_i calculated according to its classification performance on verifying dataset $\boldsymbol{V} \subset \boldsymbol{L}$, and $\delta(i, j)$ is defined as

$$\delta(i,j) = \begin{cases} 1, i=j \\ -1, i \ne j \end{cases}. \tag{5}$$

Linear fusion method is based on the assumption that the correlation among various sub-classifiers is linear, and thus applies to linear-models. However, it is inclined to fail when coping with the more complex non-linear models.

2.1.2.3. Super-kernel fusion. To capture the correlation of sub-classifiers more flexibly, super-kernel fusion can be adopted [14]. For each $\boldsymbol{x}_v \in \boldsymbol{V}$, I classification results can be obtained from I sub-classifiers and organized into an I-dimensional vector $\boldsymbol{z}_v$

$$\boldsymbol{z}_v = [f_1(\boldsymbol{x}_v), f_2(\boldsymbol{x}_v), \ldots, f_I(\boldsymbol{x}_v)]^T. \tag{6}$$

In this way, a new training dataset $\boldsymbol{Z}$ is formed

$$\boldsymbol{Z} = \left\{\boldsymbol{z}_v | \boldsymbol{z}_v = [f_1(\boldsymbol{x}_v), f_2(\boldsymbol{x}_v), \ldots, f_I(\boldsymbol{x}_v)]^T,\ \boldsymbol{x}_v \in \boldsymbol{V}\right\}. \tag{7}$$

Based on $\boldsymbol{Z}$, a super-classifier is trained as the fusion classifier, in which SVM can be adopted as well

$$F_j(\boldsymbol{x}) = S_j(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_I(\boldsymbol{x})), \tag{8}$$

where S_j is the fusion function.

In super-kernel fusion, the correlation of sub-classifiers is depicted by function S_j, which is not based on any assumption such as linear restriction. As a result, the sub-classifiers can be combined in a better-organized way according to data's intrinsic distribution.

2.1.3. Classification decision

Based on the fusion classifiers, the final classification decision is made, which comprises the patent's class label and the certainty in