Research Journal of Applied Sciences, Engineering and Technology 11(7): 788-797, 2015
DOI: 10.19026/rjaset.11.2042
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: June 8, 2015 Accepted: July 8, 2015 Published: November 05, 2015

Research Article

Development of Novel Patent Classification Framework by Exploiting Semantic Deep Learner

[1]E.V.R.M. Kalaimani, [2]P. Anbalagan and [3]E. Kirubakaran
[1]Department of CSE, Arasu Engineering College, Kumbakonam,
[2]Department of EEE, BIT Campus, Anna University,
[3]Department of SSTP (Systems), BHEL, Tiruchirappalli, India

Abstract: Knowledge documents are growing remarkably to serve the organization for information processing and various management tasks. Text mining is hard but important research topic in knowledge discovery where hidden information is extracted from unstructured and semi-structured data. Patents are rich knowledge source needed to be organized efficiently and conveniently. Patent documents are used for gathering business intelligence and identifying key trends in technology development. The main focus of this study is to propose a electrical patent classification framework based on Semantic Deep Learner (SDL). In this framework, initially key terms of the patent documents are extracted and represented using Vector Space Model (VSM), the importance of the key terms are weighted based up on their frequencies using TF-IDF. The semantic similarity between the key features is computed using cosine measure. Terms with higher correlations are synthesized into a smaller set of features. Finally the semantic deep learner is trained using the correlated features and accordingly patents are classified. The target output identifies the category of a patent document based on a hierarchical classification scheme of the International Patent Classification (IPC) standard. Our approach is new to the patent domain and shows some improvement in the classification accuracy when compared to the other state of art classifier.

Keywords: International patent classification, semantic deep learner, semantic similarity, text mining, TF-IDF, topic classification, vector space model

INTRODUCTION

Over the last decade, with the increasing availability of powerful computing platforms and high capacity storage hardware, the number of digital documents exceeds the capacity of manual control and management. Document classification is one of the most crucial techniques to organize the documents in a supervised manner. Text classification plays an important role in many application domains. People are increasingly required to handle wide ranges of information from multiple sources. As a result, knowledge management systems are implemented by enterprises and organizations to manage their information and knowledge more efficiently. Knowledge management includes sorting useful knowledge from information, storing knowledge in good order and infer new knowledge from an existing knowledge base. Turban and Aronson (2001) focus an explicit knowledge management, i.e., management of semi-structured documents such as patent documents.

Patents give exclusive rights to the inventor for using and protecting his intellectual property. The International Patent Classification (IPC) is a standard taxonomy developed and administered by the World Intellectual Property Organization (WIPO) for classifying patents and patent applications. The IPC covers all areas of technology and is currently used by the industrial property offices of more than 90 countries. The use of patent documents and the IPC for research is interesting for several reasons. The IPC covers a range of topics that spans all human inventions and uses a diverse technical and scientific vocabulary. A large part of it is concerned with chemistry, mechanics, computers, electrical and electronics. Necessarily, the IPC is thus a complex, hierarchical taxonomy, which has been refined for 30 years. Over 40 million documents have been classified in it worldwide. Furthermore, all domain experts in national and regional patent offices currently classify patent documents manually. These experts have an intimate knowledge of the IPC and aim to provide excellent and consistent classifications (Tikk *et al*., 2005). Looking from economical side, Intellectual Patent Rights (IPR) are becoming one of the most important mechanisms for business in extracting economical value from

Corresponding Author: E.V.R.M. Kalaimani, Department of CSE, Arasu Engineering College, Kumbakonam, India

creativity and encouraging greater investment in innovation.

The objective of this study is to propose an approach for patents classification framework using SDL in electrical domain. Deep learning has emerged as a new area of machine learning research since 2006. Deep learning (or sometimes called feature learning or representation learning) is a set of machine learning algorithms which attempt to learn multiple-layered models of inputs and to train complex and deep models on large amounts of data, in order to solve a wide range of text mining and Natural Language Processing (NLP) task Glorot *et al*. (2011) and Liu *et al*. (2012). Restricted Boltzman Machines (RBM) have been used as a generative models of many different types of data including labeled or unlabeled images and bag of words that represents documents. The feature learning is trying to learn a new transformation of the previously learned features at each level, which is able to reconstruct the original data. The greedy layer-wise unsupervised pre-training is based on training each layer with an unsupervised learning algorithm, taking the features produced at the previous level as input for the next level. Finally, the set of layers with learned weights could be stacked to initialize a deep supervised predictor, such as a neural network classifier, or a deep generative model, such as a Deep Boltzman Machine (Glorot *et al*., 2011).

Our novel approach explores the document patent classification framework using SDL. The automatic document classification methodology is described in the following steps. Initially, the important terms are extracted from patent documents and represented as feature vectors using VSM and features vectors are weighted using TF-IDF based on the frequency of terms in a patent document. Next, Cosine measure is applied to find the similarities between the features and depicted in a correlation matrix in order to synthesize features into a smaller set representing key features within the patent domain. Finally SDL is trained using the consolidated set of key features available in a correlation matrix. The output of SDL gives the category of the corresponding patent document. As deep learner is better than neural networks and other learners, the accuracy of the classification will be improved to some extent especially for patent document. Applying deep learning technique for classification of the documents is a new approach in patent domain.

LITERATURE REVIEW

Document categorization: Categorization can be divided in two principal phases. The first phase is document representation and the second phase is classification. Document categorization is the process of assigning a document into more than one pre-defined document classes Antonie and Zaiane (2002). Another method is document clustering that splits many documents into groups according to the similarity between documents. Similarity is measured by evaluating key representing attributes and features among documents. Both document categorization and document clustering extract and use the features of the document for group assignment. The main distinction between categorization and clustering is that document categorization compares document features and pre-defined class features and selects the most suitable document class. Document clustering divides a set of documents into groups without using pre-defined classes.

The traditional document categorization is to classify documents by experts within a specific domain. Since experts are costly and vary in capabilities and generate the result of classification is not accurate and reliable. Because of these reasons, automatic document categorization has become an important research area.

Patent analysis: Patent classification is one of the application areas in text mining. Text classification approaches for patent classification problems have to manage simultaneously very large size of hierarchy, large documents, huge features set and multi-labeled documents (Karki, 1997). International Patent Classification (IPC) is a standard taxonomy Tikk *et al.* (2005) developed and maintained by World Intellectual Property Organization (WIPO). IPC consists of about 80000 categories that cover the whole range of industrial technologies. There are 8 sections at the higher level of hierarchy, 128 classes and 648 subclasses. The IPC is a complex hierarchical system with layers of increasing order. For example, section: G physics, class: G02 Optics, subclass: G02C spectacles, main group G02C5 construction of non-Optical parts. The survey on machine learning methods for text classification and its various challenges were discussed in Sebastiani (2002, 2005). The issues with respect to the representation of documents and learning Salton *et al*. (1975) were proposed. The hierarchies for classifying a large corpus of web contents (Dumais and Chen, 2000) were implemented. There are large number of statistical classification and machine learning techniques for text classification including KNN classifier (Ko and Seo, 2000), Centroid based technique (Drazic *et al*., 2013), Naïve Bayes classifier (McCallum and Nigam, 1998) and Support Vector Machine classifier (Joachims, 1998). These machine learning techniques are applied to patent analysis (Fig. 1).

Document representation model: The Vector Space Model (VSM) proposed by Salton *et al*. (1975) is a

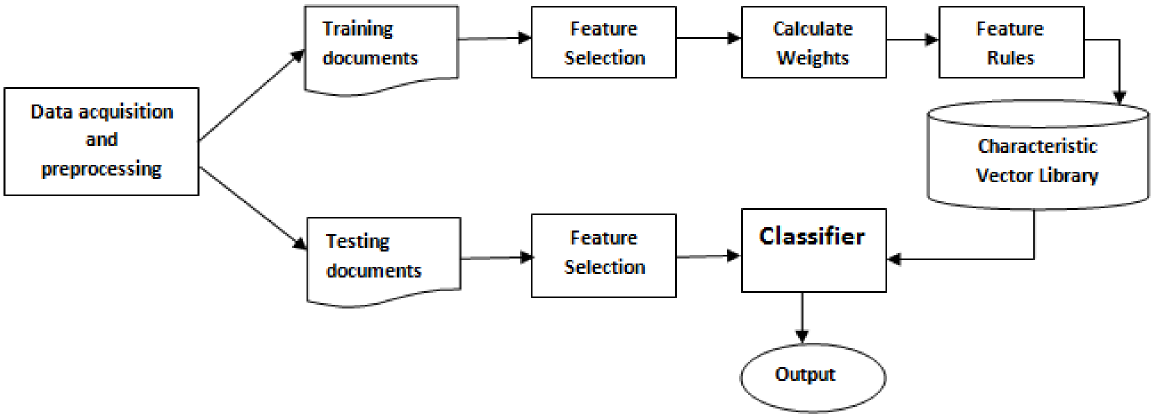


Fig. 1: Steps in document categorization

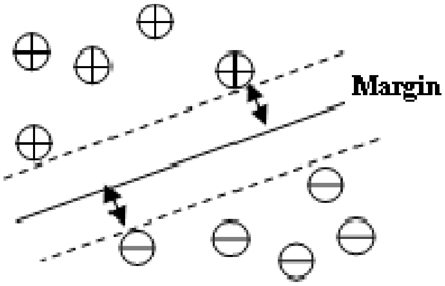


Fig. 2: Maximum-margin hyperplane

common technique for document representation in classification. In this model each document is represented as vectors of features. Each feature is associated with a weight. Usually these features are simple words. The feature weight can be simply a boolean indicating the presence or absence of the word in document, its occurrence number in document or it can be calculated by a formula like the well known *tf*idf* method. VSM has been widely used in traditional information retrieval and for automatic document categorization. There are three key steps where terms are first extracted from the document text, then the weights of the indexed terms are derived to improve the document retrieval accuracy and then the documents are ranked with respect to a similarity measure.VSM is a multi-dimensional vector where each feature of a document is a dimension. For instance, Term Frequency (TF) and Inverted Document Frequency (IDF) are two features of a text document. After the vector of a text document is derived, a cosine function is applied to measure the similarity between two documents:

$$\text{Cos (X, Y)} = \frac{\sum_{i=1}^{n} x_i.y_i}{\sqrt{\sum_{i=1}^{n} x_i^2 * \sum_{i=1}^{n} y_i^2}} \tag{1}$$

where,

X = {x_1, x_2, ….. x_n}, x_i represents i^{th} feature of document X.

Y = {y_1, y_2, …. y_n}, y_i with similarity between X and Y calculated by using cosine function.

State-of-art classifiers: There are various document classifiers proposed by previous researchers including the Support Vector Machine (SVM), k-Nearest-Neighbor (KNN), Naive Bayes (NB), Neural Networks (NN) and Genetic algorithms.

Support vector machine: The Support Vector Machine (SVM) proposed by Joachims (1998), is a supervised learning algorithm that can be applied to classification. It is a binary linear classifier which separates the positives and negatives examples in a training set. The method represents the hyperplane that separates positive examples from negative examples, ensuring that the margin between the nearest positives and negatives is maximal. The effectiveness of SVM is superior to other methods of text classification. SVM makes a model representing the training examples as the points in a dimensional space separated by the hyperplane and it uses this model to predict a new example belongs to which side of this hyperplane. The examples used in searching the hyperplane are no longer used and only these support vectors are used to classify new case. This makes a very fast method (Fig. 2).

Naive Bayes approach: Naive Bayes classification does not require more observations for all possible combinations of the variables. Each and every variable are assumed to be independent to each other. In another words, Naive Bayes classifiers assume that the influence of a variable is independent of other variables for a given class, an assumption called class conditional independence (McCallum and Nigam, 1998). This algorithm uses the joint probability of document features to calculate the probability that a new document belongs to a specific class:

$$\text{P}(C_i|\text{d'}) = \frac{P(d'|C_i).P(C_i)}{\sum_{Cj\in C} P(d'|C_i).P(C_j)} \tag{2}$$

Table 1: Comparison of different categorization approaches

Approaches	Merits	Demerits
SVM	Faster classification Less overfitting, robust to noise	Computationally expensive
Naive Bayes	Simple and quick classification Non sensitive to irrelevant features	The simplified assumption results in low accuracy, assumes independence of features
KNN	Cost of learning process is zero. Complex concepts can be learned by local approximation with simple procedure	Increasing training data will decrease efficiency, curse of dimensionality
Genetic Algorithm	Provides an optimal solution	Need for large amount of data and the algorithm can stick at local optimal solutions.
ANN	Approaches an Expert's classification results	It require sufficient training data, Learning is to slow across multiple hidden layers
Deep Learner	Highly flexible to specify prior knowledge, handle large family of function parameterized with millions of individual parameters	Involves multiple layers with complex structures

P(C_i) is the probability of a given document belonging to class C_i and P(d'|C_i) is the conditional probability of document d belonging to a specific class C_i. The probability of document d' belonging to class ci can be derived using the following equation:

$$\mathrm{P}(\mathrm{d}'|C_i) = \prod_{j=1}^{|d'|} P(W_j|C_i) \tag{3}$$

Naïve Bayes is used to construct an unstructured text classifier with sufficient accuracy.

K-Nearest Neighbor (KNN): kNN is an algorithm that uses pre-trained documents to classify new documents based on a similarity measure (Ko and Seo, 2000). kNN uses the distance between two document vectors as a measure for their similarity. The similarity z(x, c_i) is used as the confidence score to indicate x belonging to the particular category c_i and is calculated by using an equation:

$$\mathrm{Z}(\mathrm{x}, ci) = \frac{\sum_{d_j \epsilon KNN} sim(\mathrm{x}, d_j)\mathrm{y}(d_j, C_i)}{\sum_{d_j \epsilon KNN} sim(\mathrm{x}, d_j)} - b_i \tag{4}$$

where, sim (x, d_j) is the similarity between tested document x and trained document d_j which is calculated using Euclidean distance or the cosine value between two document vectors. y (d_j c_i) is 1 (or 0) when the trained document dj (not) belongs (or does not belong) to C_i. Finally, bi is the threshold of classifying a document to categoryc_i.

Genetic algorithm: The genetic algorithm Holland (1992) works well on mixed (continuous and discrete) and combinatorial problems. Three operational components selection, crossover and mutation are used to generate various models. The GA can provide an optimal solution but is possible for the algorithm to stick at local optimal solutions. Another shortcoming of GA is the computational requirements of the algorithm, which may not be a concern when there is strong computing power.

Artificial neural network: Artificial Neural Network (ANN) is an information processing method inspired by biological nervous systems. An ANN uses interconnected processing nodes computationally linked to solve problems. Neural networks are frequently used for pattern recognition and document classification and learn by using training data to adjust the weights between connecting nodes. Some research has applied artificial neural networks to text classification. Document clusters Farkas (1994) are generated using thesaurus and neural network. Adaptive Resonance Theory (Massey, 2003) was used to cluster documents. Web-page classification system (Selamat and Omatu, 2004) that uses a neural network with inputs gained by principal component analysis and class profile-based features that contain the most regular words in each class. Table 1 shows the advantage and shortcomings of the commonly used categorization models.

Necessity of knowledge management and content delivery: Knowledge management is a process that helps organizations identify, select, organize, disseminate and transfer knowledge. The previous research indicates expertise is a part of the organizational memory that typically resides within the organization in an unstructured manner. Effective document management is a great help to the implementation of knowledge management documents Their contribution to academic research has been dramatically declining from 30% of overall contributions up to 2002, to only 10% by 2009 (Serenko *et al*., 2010). An electronic document is any electronic media content (other than computer programs or system files) that are intended to be used in either an electronic form. Even more problems are connected with complex file formats of various word processors, spreadsheets and graphics software. To alleviate the problem, many software companies distribute free file viewers for their proprietary file formats (one example is Adobe's Acrobat Reader). The other solution is the development of standardized non-proprietary file formats (such as HTML and Open Document) and electronic documents for specialized uses have specialized formats-the specialized electronic articles in physics use TeX or PostScript.

METHODOLOGY

This section depicts the detailed methodologies for patent classification. First, a document content extraction model is built to represent the patent document content with a vector values. Next a document classification model is developed based on the training set to classify the patent documents. Finally, the model is validated by testing patent documents.

Patent extraction and representation: The input patent documents are preprocessed by using a stemming and stop words removal. In this preprocessing step,the input patent documents from different domain are divided into training and testing sets. Both sets are represented as bag of words. Each one length terms are weighted using Term-Frequency (TF) and Inverse Document Frequency (IDF):

$$d_n = \{t_1, t_2 \ldots\ldots\ t_m\} \tag{5}$$

$$TF\ [d_n(t_m)] = frq\ (t_m) \tag{6}$$

In the above equations, $\{t_1, t_2 \ldots\ldots t_m\}$ are the one length unique terms of the n^{th} document; and $TF\ [d_n(t_m)]$ denotes the term frequency of m^{th} one length unique term of n^{th} document; and $frq(t_m)$ denotes the number of times m^{th} one length unique term repeated in n^{th} document. The inverse document frequency (idf) of the one length unique term is calculated by summing the frequency of the one length unique term in other documents except the respective document which is shown by an equation given below:

$$IDF[d_n(t_m)] = \sum_{a=1}^{n-1} frq[d_a(t_m)] + \sum_{a=n}^{N} frq[d_a(t_m)] \tag{7}$$

In the above equation $IDF[d_n(t_m)]$ is the inverse document frequency of m^{th} one length unique term of n^{th} document; and $frq\ [d_a(t_m)]$ denotes the number of times m^{th} one length unique term repeated in a^{th} document, where a varies from 1 to N excluding the n^{th} document. The term frequency-inverse document frequency of the one length unique term is then calculated using the term frequency and inverse document frequency of the same one length unique term. The term frequency- inverse document frequency is shown by the equation below:

$$TF\_IDF\ [d_n(u_p)] = \log\left(TF\ [d_n(u_p)]\right) \times \log\left(\frac{N}{IDF\ [d_n(u_p)]}\right) \tag{8}$$

The document vectors are constructed using the above values by using standard VSM. Similarly, two length unique terms (phrases) are identified and weighted.

Correlation matrix: The document vectors are grouped based on the similarity of the terms using cosine similarity measure based on the Eq. (1). A domain term that appears in a document with high frequency indicates that the term is a significant keyword or key features. After extracting all high frequency terms, a correlation matrix of terms is created by calculating their frequency of occurrence within same documents. The correlation of two key features $(CT_{i,}CT_j)$ appeared in set of patent documents are determined using equation.

$$X_{ij} = \frac{\sum_{i=1}^{N_D} X_{i,I}X_{j,I} - N_D\,\overline{X_i}\,\overline{X_j}}{\sqrt{\sum_{i=1}^{N_D} X^2{}_{i,I} - N_D\overline{X^2{}_i}}\,.\sqrt{\sum_{i=1}^{N_D} X^2{}_{j,I} - N_D\overline{X^2{}_j}}} \tag{9}$$

where, X_{ij} is the correlation of $CT_i\ and\ CT_j$ that appear in a set of patent documents; $X_{i,I}$ is the frequency that CT_i appeared in document D_i, $X_{j,I}$ is the frequency that CT_j appeared in document $D_j.\overline{X_i}$ is the average frequency that CT_i appeared in all documents(all D_s); $\overline{X_j}$ is the average frequency that CT_j appeared in all documents (all D_s); N_D is the total number of documents. The highly correlated key features are selected and stored as the related features list. Finally, the key features list is completed when highly correlated features are merged, a necessary step since it is easier to train SDL models using fewer variables. When a new document is uploaded into a patent knowledge management system, the key features and their frequencies are extracted from the document. The frequencies of all terms are derived. Then CTf_i is used to present the frequency of key terms CT_i in the document and CTf_{ij} to represent the frequency of related-term XT_{ij} . The correlation of XT_{ij} and CT_i are listed as X_{ij} and the final frequency of CT_i is:

$$CTF_i = CTf_i + \sum_{j=1}^{k} CTf_{ij} \,.\, CX_{ij} \tag{10}$$

After calculating the CTF of all key features, a vector of key features frequencies is listed as:

$$[CTF_{1,}CTF_{2,}CTF_{3,} \ldots\ldots\ldots\ldots CTF_n\,] \tag{11}$$

The correlation matrix is constructed based on the vectors generated mapping the top features and corresponding documents. This matrix is trained by SDL.

Document categorization using SDL: In this section, we describe the document categorization methodology based on the Semantic Deep Learner (SDL). It is a class of machine technique that exploit many layers of non-linear information processing for supervised or un-supervised extraction and transformation and for pattern analysis and classification. An advantage of SDL is that

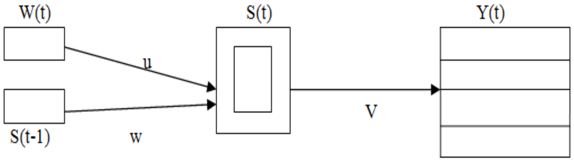


Fig. 3: Structure of the simple learning model; where, W(t): input layer; S(t): hidden layer; Y(t): output layer and u, v, w: weight matrices.

it does not need to change the network structure and achieve the target output. Another advantage of using SDL is its rapid execution when a trained network is applied. The learning stage of SDL involves a pre-training phase and a fine tuning for reconstruction. The structure of the simple learning model is depicted in Fig. 3. The two passes of SDL are described in the following section.

The hidden layer provide recurrent connection to S(t-1) and thus provide short term memory that models context of a word. The generated feature frequency matrix is given as input to SDL. Deep architecture is identical to the multi-layer physical structure of the human cerebral cortex. The neocortex, which is associated with many cognitive abilities, has a complex multilayer hierarchy. The development of intelligence follows with the multi-layer structure. From an evolutionary viewpoint, the phylogenetically most recent part of the brain is the neocortex. In humans and other primates, starting from catarhinians, the multi-layers structure began to appear in the neocortex. Therefore, a deep architecture actually represents the result of human intelligence evolution. It thus provides a possible way to achieve the ultimate target of natural language processing, which is to enable the computer to understand the human (natural) languages. The proposed deep architecture of our proposed system is depicted in Fig. 4.

Here, the correlation matrix formed using the whole document taken for training is given as input to the semantic deep learner. The deep learner will train the system based on the target given. The training is done by the hidden layer of the deep learner that exploits the target and the input. The target would be based on the documents taken for training belongs to the topic, because we know the topic (domain) for the documents taken for training. After training the system

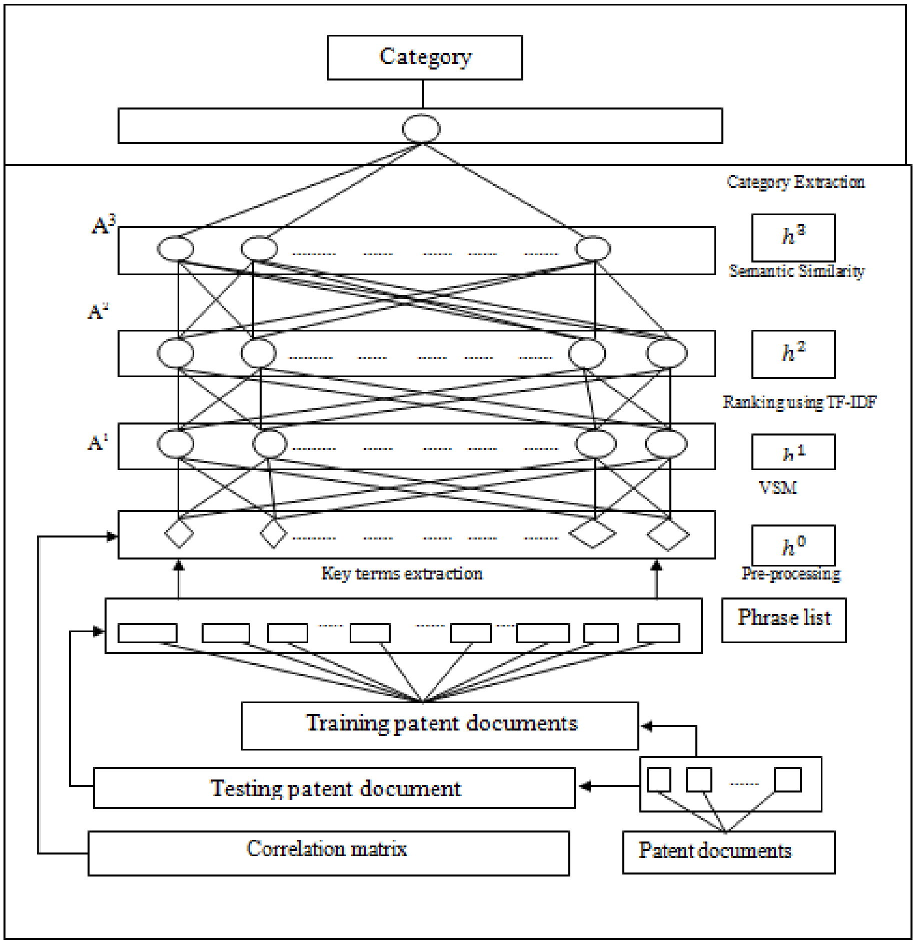


Fig. 4: Deep architecture of our proposed system

based on the semantic deep learner, the testing is done by giving the testing document. When the testing document is given as input to the system, the correlation matrix is formed for the input document using the keywords that formed the correlation in the training process. The semantic deep learner will give a score for the given input document and based on the score the document will be classified to which category it belongs.

Pre-training phase: The pre-training phase starts from the input layer. The input of every node is calculated. Then, the output from the activation functions of the nodes is derived and is passed pre-training to the next layer where processing continues until reaching the final output layer. The net input from input layer to the hidden layer node j is calculated using:

$$net_j^h = \sum_{n\in previous\ layer} w_{ij}^h\, x_i + b_j \tag{12}$$

where, w_{ij}^h is the weight of connection between input layer node i and hidden layer node j, x_i is the input of node i and b_j is the bias associated with node j. The output of node j can be determined using the following equation:

$$H_j = \mathrm{f}(net_j^h) \tag{13}$$

The activation function f(x) and the net input from hidden layer to output layer node k is computed using:

$$net_k^0 = \sum_j w_{jk}^0\, H_j \tag{14}$$

where, w_{jk}^0 is the weight of connection between hidden layer node j and output layer node k. Finally, we can determine the output of the SDL:

$$O_k = \mathrm{g}(net_k^0) = \mathrm{g}(\sum_j w_{jk}^0\, H_j) \tag{15}$$

g(x) denotes the activation function of node k.
The error of the network is expressed as:

$$\mathrm{E} = \frac{1}{2}\sum_k (T_k - O_k)^2 \tag{16}$$

where, T_k is the real output of the training data.

Fine tuning phase: In this phase, the transfer of data back from the output layer to the previous hidden layer is carried out. The fine tuning is used for determining errors and adjusting weights. Since E is defined as the function of O_k and O_k is the function of w_{jk}^0, the weight adjustment between output layer and hidden layer Δw_{jk}^0 can be expressed as:

$$\Delta w_{jk}^0 = \eta((T_k - O_k) g^{'}(net_k^0)\, H_j = \eta \partial_k^0 H_j \tag{17}$$

where,
η = The learning rate and $\partial_k^0 = (T_k - O_k)\, g^{'}(net_k^0)$

Similarly, since O_k is a function of H_j, the function of w_{ij}^{h} is the weight adjustment between hidden layer and input layer:

$$\Delta w_{ij}^{\mathrm{h}} = \eta \partial_k^0 w_{jk}^0 f^{'}((net_j^{\mathrm{h}})\, H_j = \eta\, \partial_j^{\mathrm{h}} H_j \tag{18}$$

Therefore, the weight adjustment is depicted as:

$$\Delta w_{ij} = \eta \partial_j \mathrm{f}(net_i) \tag{19}$$

where,
∂_j = The output error of layer j, net$_i$ is the input of layer i.

As described in above section, the key term frequency (CTF) vector is used to represent a patent document. Before importing the vector into the SDL, all key phrase frequencies in the vector are normalized between 0 and 1 using a transformation function:

$$CTF_i^{'} = \frac{CTF_{i,}}{\max\,(CTF_1, CTF_2, CTF_3, \ldots\ldots\ldots CTF_n)} \tag{20}$$

The output values represent the goodness-of-fit between a test document and all potential classes. In our study, IPC provides the target classes for the patent documents.

Evaluation metrics: To analyze the performance of classification, we adopt the following measure. Four cases are considered as the result of classifier to the document.

TP (True Positive): The number of documents correctly classified to the respective class.

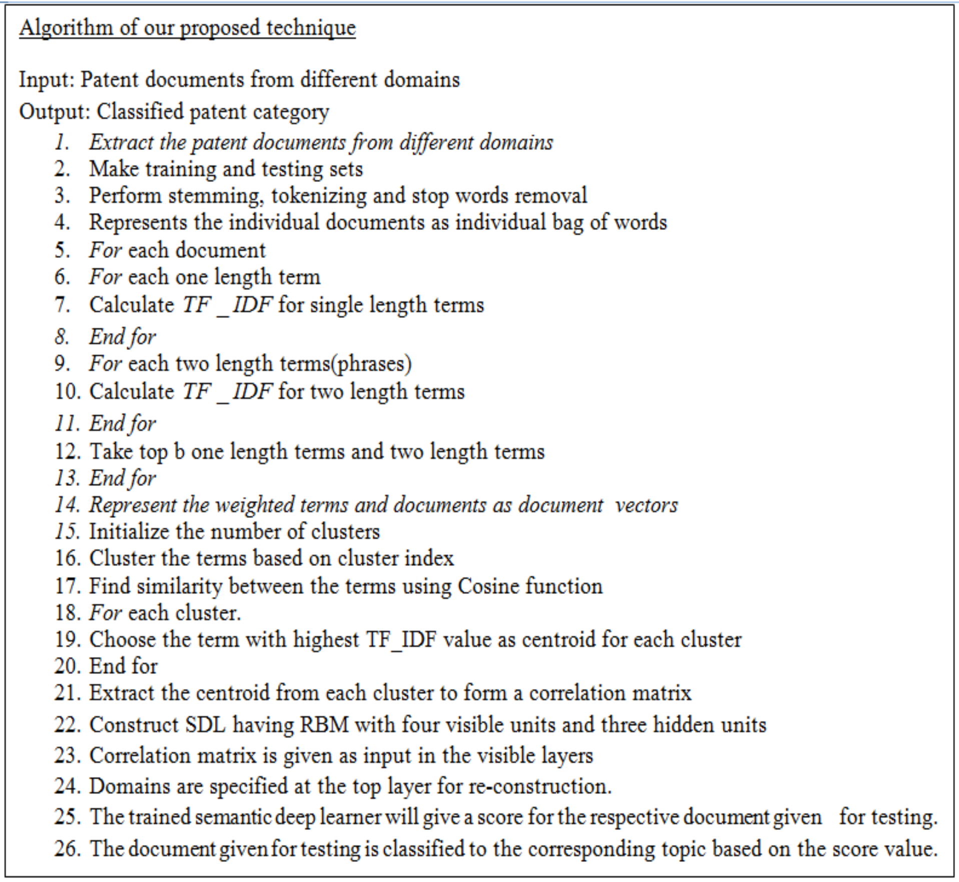
TN (True Negative): The number of documents correctly rejected from the class.

FP (False Positive): The number of documents incorrectly rejected from the class.

FN (False Negative): The number of documents incorrectly classified to the class.

Using these quantities, the performance of the classification is evaluated in terms of precision (pr), recall (re) and Fl measure. Recall is defined to be the ratio of correct assignments by the system divided by the total number of correct assignments. Precision is the ratio of correct assignments by the system divided by the total number of the system's assignments. The Fl measure is the combination of recall and precision with an equal weight.

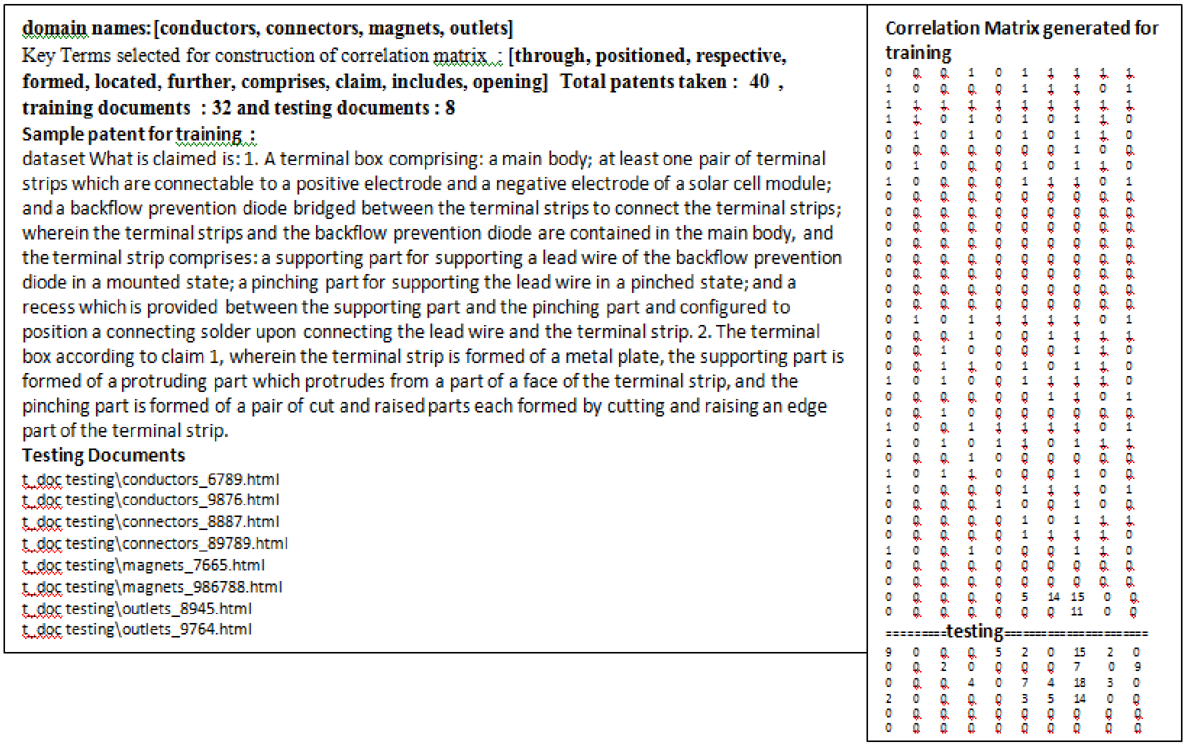
$$\mathrm{pr(Ci) = TP/TP(Ci)+FP(Ci)} \tag{21}$$



Algorithm of our proposed technique

```
Input: Patent documents from different domains
Output: Classified patent category
1. Extract the patent documents from different domains
2. Make training and testing sets
3. Perform stemming, tokenizing and stop words removal
4. Represents the individual documents as individual bag of words
5. For each document
6. For each one length term
7. Calculate $TF\_IDF$ for single length terms
8. End for
9. For each two length terms(phrases)
10. Calculate $TF\_IDF$ for two length terms
11. End for
12. Take top b one length terms and two length terms
13. End for
14. Represent the weighted terms and documents as document vectors
15. Initialize the number of clusters
16. Cluster the terms based on cluster index
17. Find similarity between the terms using Cosine function
18. For each cluster.
19. Choose the term with highest TF_IDF value as centroid for each cluster
20. End for
21. Extract the centroid from each cluster to form a correlation matrix
22. Construct SDL having RBM with four visible units and three hidden units
23. Correlation matrix is given as input in the visible layers
24. Domains are specified at the top layer for re-construction.
25. The trained semantic deep learner will give a score for the respective document given for testing.
26. The document given for testing is classified to the corresponding topic based on the score value.
```

Fig. 5: Depicts the sample experimental work



domain names: [conductors, connectors, magnets, outlets]

Key Terms selected for construction of correlation matrix : **[through, positioned, respective, formed, located, further, comprises, claim, includes, opening] Total patents taken : 40 , training documents : 32 and testing documents : 8**

Sample patent for training :

dataset What is claimed is: 1. A terminal box comprising: a main body; at least one pair of terminal strips which are connectable to a positive electrode and a negative electrode of a solar cell module; and a backflow prevention diode bridged between the terminal strips to connect the terminal strips; wherein the terminal strips and the backflow prevention diode are contained in the main body, and the terminal strip comprises: a supporting part for supporting a lead wire of the backflow prevention diode in a mounted state; a pinching part for supporting the lead wire in a pinched state; and a recess which is provided between the supporting part and the pinching part and configured to position a connecting solder upon connecting the lead wire and the terminal strip. 2. The terminal box according to claim 1, wherein the terminal strip is formed of a metal plate, the supporting part is formed of a protruding part which protrudes from a part of a face of the terminal strip, and the pinching part is formed of a pair of cut and raised parts each formed by cutting and raising an edge part of the terminal strip.

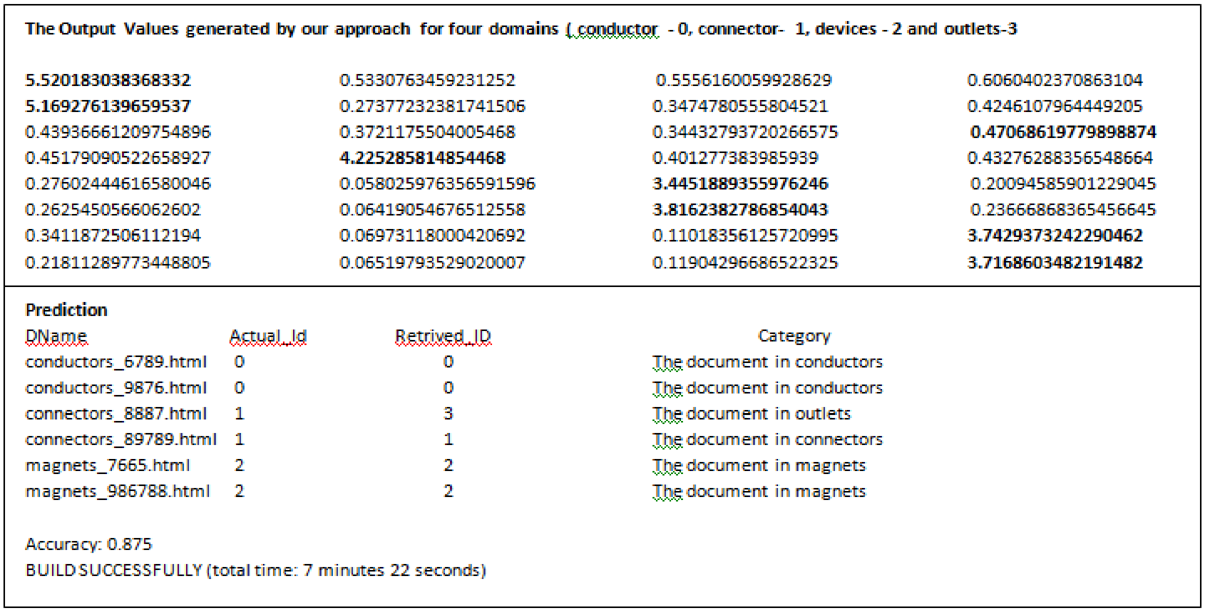
Testing Documents

```
t_doc testing\conductors_6789.html
t_doc testing\conductors_9876.html
t_doc testing\connectors_8887.html
t_doc testing\connectors_89789.html
t_doc testing\magnets_7665.html
t_doc testing\magnets_986788.html
t_doc testing\outlets_8945.html
t_doc testing\outlets_9764.html
```

Correlation Matrix generated for training

```
0    0    0    1    0    1    1    1    1    1
1    0    0    0    0    1    1    1    0    1
1    1    1    1    1    1    1    1    1    1
1    1    0    1    0    1    0    1    1    0
0    1    0    1    0    1    0    1    1    0
0    0    0    0    0    0    0    1    0    0
0    1    0    0    0    1    0    1    1    0
1    0    0    0    0    1    1    1    0    1
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    1    0    1    1    1    1    1    0    1
0    0    0    1    0    0    1    1    1    1
0    0    1    0    0    0    0    1    1    0
0    0    1    1    0    1    0    1    1    0
1    0    1    0    0    1    1    1    1    0
0    0    0    0    0    0    1    1    0    1
0    0    1    0    0    0    0    0    0    0
1    0    0    1    1    1    1    1    0    1
1    0    1    0    1    1    0    1    1    1
0    0    0    1    0    0    0    0    0    0
1    0    1    1    0    0    0    1    0    0
1    0    0    0    0    1    1    1    0    1
0    0    0    0    1    0    0    1    0    0
0    0    0    0    0    1    0    1    1    1
0    0    0    0    0    1    1    1    1    0
1    0    0    1    0    0    0    1    1    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    5    14   15   0    0
0    0    0    0    0    0    0    11   0    0
========testing====================
9    0    0    0    5    2    0    15   2    0
0    0    2    0    0    0    0    7    0    9
0    0    0    4    0    7    4    18   3    0
2    0    0    0    0    3    5    14   0    0
0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0
```

Fig. 6: Sample patents, domains, key terms and correlation matrix of our experimentation



```
The Output  Values  generated by our approach  for four domains ( conductor  - 0, connector-  1, devices - 2 and outlets-3

5.520183038368332            0.5330763459231252          0.5556160059928629          0.6060402370863104
5.169276139659537            0.27377232381741506         0.3474780555804521          0.4246107964449205
0.43936661209754896          0.372117550400546 8         0.34432793720266575         0.47068619779898874
0.45179090522658927          4.225285814854468           0.401277383985939           0.43276288356548664
0.27602444616580046          0.058025976356591596        3.4451889355976246          0.20094585901229045
0.2625450566062602           0.06419054676512558         3.8162382786854043          0.23666868365456645
0.3411872506112194           0.06973118000420692         0.11018356125720995         3.7429373242290462
0.21811289773448805          0.06519793529020007         0.11904296686522325         3.7168603482191482

Prediction
DName                  Actual_Id         Retrived_ID                    Category
conductors_6789.html    0                 0                  The document  in conductors
conductors_9876.html    0                 0                  The document  in conductors
connectors_8887.html    1                 3                  The document  in outlets
connectors_89789.html   1                 1                  The document  in connectors
magnets_7665.html       2                 2                  The document  in magnets
magnets_986788.html     2                 2                  The document  in magnets

Accuracy: 0.875
BUILD SUCCESSFULLY (total time: 7 minutes 22 seconds)
```

Fig. 7: The output of SDL showing the predicted category with accuracy

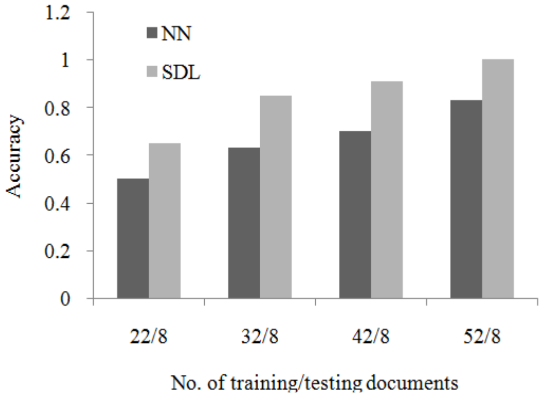


Fig. 8: Performance comparison of our proposed system SDL with neural network based classifier for patents classification having learning rate 0.3

$$re(Ci) = TP(Ci) /TP(Ci) +FN(Ci) \qquad (22)$$

$$F1(Ci) = 2^* pr(Ci)^* re(Ci)/Pr(Ci)+re(Ci) \qquad (23)$$

$$Accuracy\ (Ci) = TP(Ci)+TN(Ci)/TP(Ci)+ FP(Ci)+FN(Ci)+TN(Ci) \qquad (24)$$

EXPERIMENTAL RESULTS

Patents taken for analysis consists of 120 patents in the electrical field downloaded from http://www.freepatentsonline.com. These patents were classified in the following four domains: Conductors, Connectors, Devices and Outlets. Figure 5 depicts the sample experimental work. Figure 6 depicts the details of the domains, key terms, sample patent in electric conductor domain, testing documents and correlation matrix [doc Vs term] of training and testing sets. Figure 7 depicts the output of SDL showing the predicted category with accuracy. Figure 8 represents the classification accuracy of our proposed system based on the claim information. The result reveals that our proposed approach for classification outperforms other classifiers in the patent domain.

CONCLUSION

The approach proposed in this study suggests efficient solution for patent document classification using SDL. There are certain shortcomings in applying SDL because insufficient training data may lead to the unreliable model and at the same time the training procedure involves more computing resources. A well-trained semantic model can help companies better manage documents, the cost of computing resources can be easily justified. The patent documents extracted from WIPO, are originally classified using a hierarchical classification scheme. After the extraction of key terms and their frequencies, the resulting key feature base adequately represents the characteristics of the patent documents. The ability of deep learning approach to discover the hidden structures and features at different levels of abstraction is useful for efficiently classifying patents. The accuracy of the trained model is found to be better than other classifier and it was evidenced only with electrical patents. The extension of this study involves combining the deep learner with other approaches such as semantic smoothing models and ontology-based feature extraction methods to improve the flexibility and accuracy of the current method. Besides the classification of patent documents, our work presents framework which can be used to extract more meaningful data representation for analysis of other type of documents.

REFERENCES

Antonie, M.L. and O.R. Zaiane, 2002. Text document categorization by term association. Proceeding of IEEE International Conference on Data Mining pp: 19-26.

Drazic, M., D. Kukolj, M. Vitas, M. Pokric, S. Manojlovic and Z. Tekic, 2013. Technology Matching of the patent documents using clustering algorithms. Proceeding of IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI, 2013), pp: 405-409.

Dumais, S. and Chen, 2000. Hierarchical classification of web content. Proceeding of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp: 256-263.

Farkas, J., 1994. Generating document clusters using thesauri and neural networks. Proceeding of the 1994 Canadian Conference on Electrical and Computer Engineering. Halifax, NS, 2: 710-713.

Glorot, X., A. Bordes and Y. Bengio, 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. Proceeding of the 28th International Conference on Machine Learning, (ICML, 2011).

Holland, J.H., 1992. Genetic algorithms. Sci. Am., 4: 44-50.

Joachims, T., 1998. Text categorization with support vector machines: Learning with many relevant features. Proceeding of the 10th European Conference on Machine Learning, pp: 137-142.

Karki, M.M.S., 1997. Patent citation analysis: A policy analysis tool. World Patent Inform., 19(4): 269-272.

Ko, Y. and J. Seo, 2000. Automatic text categorization by unsupervised learning. Proceeding of the 18th Conference on Computational Linguistics (COLING’ 2000). Saarbru¨cken, Germany, pp: 453-459.

Liu, Y., S.H. Zhong and W.J. Li, 2012. Query-oriented multi-document summarization via unsupervised deep learning. Proceeding of the 26th Conference on Artificial Intelligence (AAAI'12).

Massey, L., 2003. On the quality ART1 text clustering. Neural Networks, 16: 771-778.

McCallum, A. and K. Nigam, 1998. A comparison of event models for naive bayes text classification. Proceeding of the AAAI’98 Workshop on Learning for Text Categorization. Madison, pp: 41-48.

Salton, G., A. Wong and C.S. Yang, 1975. A vector space model for information retrieval. Commun. ACM, 18(11): 613-620.

Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Comput. Surv., 34(1): 1-47.

Sebastiani, F., 2005. Text Categorization. In: Zanasi, A. (Ed.), Text Mining and its Applications to Intelligence, CRM and Knowledge Management, WIT Press, Southampton, UK, pp: 109-129,

Selamat, A. and S. Omatu, 2004. Web page feature selection and classification using neural networks. Inform. Sciences, 158: 69-88.

Serenko, A., N. Bontis, L. Booker, K. Sadeddin and T. Hardie, 2010. A scientometric analysis of knowledge management and intellectual capital academic literature (1994-2008). J. Knowl. Manage., 14(1): 13-23.

Tikk, D., G. Biró and J.D. Yang, 2005. Experiment with a Hierarchical Text Categorization Method on WIPO Patent Collections. In: Attok-Okine, N.O. and B.M. Ayyub (Eds.), Applied Research in Uncertainty Modelling and Analysis. International Series in Intelligent Technologies, Springer, 20: 283-302.

Turban, E. and J.E. Aronson, 2001. Decision Support Systems and Intelligent Systems. 6th Edn., Prentice-Hall International, Upper Saddle River, NJ.