WILEY | Hindawi

Retraction

Retracted: Hybrid Graph Neural Network Model Design and Modeling Reasoning for Text Feature Extraction and Recognition

Wireless Communications and Mobile Computing

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] X. Huang, D. Qiu, C. Xiang, and H. Chen, "Hybrid Graph Neural Network Model Design and Modeling Reasoning for Text Feature Extraction and Recognition," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 5220916, 8 pages, 2022.

WILEY | Hindawi

Research Article

Hybrid Graph Neural Network Model Design and Modeling Reasoning for Text Feature Extraction and Recognition

Xuxiang Huang , Dong Qiu, Chen Xiang, and Hewei Chen

School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China

Correspondence should be addressed to Xuxiang Huang; 201931119020204@stu.hubu.edu.cn

Natural language processing is an important branch of deep learning. In particular, the classification of short texts is one of the main tasks of computer linguistics, because it ensures the security of information. Therefore, this paper reviews the text classification methods for the first time, aiming at comparing the modern methods to solve the text classification problems, determine the trend in this direction, and select the best algorithm to be applied to research and business tasks. In this paper, with the help of the emerging algorithm in the field of deep learning, a web domain knowledge text classification method based on feature words is developed with the support vector machine (SVM) as the basic classification algorithm. Two datasets representing basic research disciplines and emerging research fields are selected to verify the proposed research framework. Experiments show that the performance of this method is improved by 3% compared with the benchmark method.

1. Introduction

The convolutional neural network (CNN) has been developed as an object of application of image recognition, but recently, it has been known that it can be applied to classification of documents and can be classified into high accuracy. This paper classifies documents by the CNN with a relatively deep layer and acquires excellent classification results [1]. However, there are few examples of document classification using a character-level CNN for Japanese documents. As a cause, there is a method called the transition learning using the learning result of other tasks as a method to deal with the problem. In this paper, we apply the residual network to the conventional character-level CNN [2]. We also apply this method to various datasets and confirm their effects. Convolutional neural networks are attractive. In a short time, they have become a disruptive technology, breaking all the most advanced algorithms in many fields, such as text, video, and voice, far beyond their initial application in image processing. The CNN consists of many neural network layers. The two different types of layers, convolution and pooling, are usually alternating. The depth of each filter in the network increases from left to right. Finally, it usually consists of one or more fully connected layers.

Advances in microelectronics and information technology have led to a wide range of applications for the real-time processing of large data streams [3]. For example, many simple operations in everyday life, such as using a credit card or telephone, require the automatic creation, analysis, and processing of a wide range of data. Because these operations are often carried out by a large number of participants, distributed and massive data streams are required. Similarly, social media contains large streams of network-specific and textual data [4]. The problem of creating models and algorithms is therefore relevant.

In order to ensure information and public safety, the analysis of content containing illegal information in telecommunications networks (including data related to terrorism, drug trafficking, online extremism, and preparations for protest movements or mass riots) is important to ensure information security [5]. The goal of this paper is to compare modern approaches to solving text classification problems, to discover trends in the field and to select the best algorithms for research and commercial tasks.

Text classification methods lie at the intersection of the two fields of information retrieval and machine learning. Their similarities lie in the way that the documents themselves are represented and in the quality of the algorithms. Currently, many methods and their different variations have been developed to classify text. Each group of methods has its advantages and disadvantages, areas of application, characteristics, and limitations [6].

Of particular interest are the following cases where the data is in the form of streams, for example, in telecommunication networks, some difficulties arise because model training is always based on a file of a set of attributes [7]. These sets of attributes may change over time, and it is necessary to consider the possible changes in the underlying distribution of data when constructing stream classifiers [8]. The preferred choice of method is one that can support incremental learning, i.e., we want the chosen method to support incremental learning, where the classifier learns on each sample in real time. In incremental learning, the training samples arrive sequentially during the sampling process; therefore, the classifier must constantly correct the training results and the classifier must constantly correct the training results and retrain itself [9]. In nonincremental training, the entire training sample is provided in its entirety at once. The entire training sample is provided all at once. Obviously, in the case of incremental learning, the behaviour of the classifier changes during operation, which reduces its predictability and may make the system difficult to adjust. At the same time, incremental learning makes the system more flexible and able to adapt to changing conditions.

The specificity of the classification process in streams is also due to the fact that it is not always possible to control the speed at which data arrives. Some of the document categories that may be encountered in a stream are only sporadic. Detecting such rare classes can be difficult [10]. Classes may be difficult to find, in which case it becomes very difficult to classify text. In such a case, the "I" becomes an extremely difficult task. The comparison of classifier construction methods is a rather difficult task, as different input data may lead to different results. It is therefore necessary to perform software implementations and performance calculations for training and testing on the same set of documents [11].

2. Related Work

With the continuous enrichment of web resources, web-oriented text classification research is gaining more and more attention [12, 13]. In this research, we use the network structure of vdcnn and we use the network which is suitable for Japanese document classification. As a preliminary study on the transition learning, Moo et al. report that the transition learning works well when the similarity of the transition between the transitive and the transitional task is high in the document classification using the neural network.

The traditional text classification methods for the massive amount of information on the web are mainly manual classification and organization. However, the traditional methods using manual classification have many drawbacks, such as high labor and high material and energy consumption and low consistency of classification results [14]. Currently, the general technical process of text classification is as follows: first, the unstructured text data is preprocessed to represent the text in a structured form, which is called feature representation; second, feature selection is performed to select the feature items that best represent the text content to reduce the dimensionality of the feature vector space; then, the training document set is used to construct and train the classifier; finally, the new text is classified using the constructed classifier [15]. Finally, the constructed classifier is used to classify the new text. In [15], the maximum entropy model is applied to the text classification study in conjunction with the current text classification study. Reference [16] extracted features from text using the deep belief network and classified the extracted features using the softmax regression classifier, and the experiments showed that text classification using the deep belief network has good performance. Reference [17] proposed a text similarity weighting algorithm based on the semantic similarity of the knowledge network and conducted Chinese text classification experiments on the algorithm, and the results showed that the method improved the performance in text classification compared with the traditional text similarity.

Although the scheme of the abovementioned researchers has achieved certain results, the recognition of special characters is not very accurate but the CNN model designed in this paper also has good recognition ability for special symbols.

3. Text Classification

A distinction should be made between classification and clustering. In document classification, categories are predefined, whereas in clustering, they are not and even information about their number may not be available. It may be absent.

Formally, the formulation of the classification problem can be expressed in the following way.

There is a document set $D = \{d_1, \cdots, d_{|D|}\}$ and a set of possible categories (classes) $C = \{c_1, \cdots, c_{|D|}\}$. The unknown objective function $D \times C \longrightarrow \{0, 1\}$ given by the following:

$$\Phi(d_j, c_i) = \begin{cases} 0, & d_j \notin c_i, \\ 1, & d_j \in c_i. \end{cases} \tag{1}$$

A classifier needs to be constructed that is as close as possible to Φ. In the formulation of this problem, it should be noted.

A classification is called an exact classification if the classifier gives an exact answer.

$$\Phi : D \times C \longrightarrow \{0, 1\}. \tag{2}$$

If the classifier determines the classification status value of a document, the classification is called a threshold classification.

$$\text{CSV} : D \longrightarrow [0, 1]. \tag{3}$$

In general, the process of supervised learning (supervised learning) is as follows. The system is presented. The system presents a set of examples relating to the previous ones. This set of data is sometimes referred to as the training sample L. It is used to train a classifier and to determine the values of its parameters so that the classifier gives the best results at that value. Next, the system generates solution rules that are used to classify a set of instances into a given class. The quality of the separation is checked using a test sample of instances T. It is necessary to satisfy the following conditions:

$$L \cap T = \varnothing, \\ \Omega = L \cup T \subset C \times D, \tag{4}$$

where Ω means a set of example and Φ means the values of the objective function.

If in a task, each document $d \in D$ may correspond to only one c to only one category $C \in c$, then, there is a single-valued classification and, i.e., any number of categories.

A special case of disambiguation classification is binary classification, where a collection of documents needs to be divided into two nonoverlapping categories, for example, the task of determining the tone of speech (positive or negative) or the task of detecting or, with the help of a binary classifier, solving the task of spam detection (whether a message is spam).

The solution to the classification problem consists of four successive steps.

(1) Preprocessing and indexing of documents

(2) Reducing the dimensionality of the feature space

(3) Constructing and training the classifier using machine learning methods

(4) Evaluating the quality of the classification

When selecting a particular classification algorithm, the specific characteristics of each algorithm should be taken into account. There is still an unsolved problem. The method for determining the set of classification features, their number, and calculating the weights remains unsolved. In deep learning algorithms, the accuracy of classification depends heavily on the availability of a training. The accuracy of classification depends heavily on the availability of appropriately sized training samples. Preparing such samples, sampling is a very time-consuming process. Until now, the problem of selecting parameters for some algorithms in the training phase has remained unresolved. In the following, each phase is considered in detail. The different algorithms used to construct the classifier, the experiments conducted with these algorithms, and the results of these experiments are described in detail.

3.1. Preprocessing and Indexing of Documents. Text preprocessing includes symbolization, removal of function words (semantically neutral words such as conjunctions and prepositions). The following work was carried out by morphological analysis (partitioning part of speech and word roots). This allowed for a significant reduction in the dimensionality of the space. As a result, all meaningful words occurring in the document are represented as features of the document.

A document index is a construction of some numerical model of a text that translates the text into a representation suitable for further processing.

For example, the bag-of-words model allows a document to be represented as a multidimensional vector consisting of words and their weights within the document [18]. In other words, each document is a vector in a multidimensional space whose coordinates correspond to the number of words and whose values correspond to the weights.

A common indexing model is Word2vec [19]. It represents each word as a vector containing information about the contextual (relevant) words. Another indexing model is based on taking into account n-grams, i.e., sequences of adjacent characters. Obviously, the same approach should be used for training and testing documents.

3.2. Features That Reduce the Dimensionality of the Feature Space. Different computational complexity and different classification methods depend directly on the dimensionality of the feature space. This is why the use of classifiers often resorts to reducing the number of features (terms) used.

By reducing the dimensional term space, it is possible to reduce the effects of overlearning—a phenomenon where the classifier is guided by random or incorrect features.

An overtrained classifier works well on the instances on which it has been trained and significantly worse on the test data. To avoid overtraining, the number of training instances should be proportional to the number of trainees. In some cases, reducing the dimensionality of the feature space by a factor of 10 (or even 100) may only lead to a deterioration in the nonsignificance of the classifier.

There are several ways to determine document feature weights. The most common one is the TF-IDF function. Its basic idea is to give more weight to words that have a high frequency in a particular document and a low frequency in other documents.

The term frequency (TF) is calculated—an estimate of the importance of a term in a single document d—using the following formula:

$$\text{TF} = \frac{n_{td}}{n_d}. \tag{5}$$

IDF reduces the weight of common words according to the following formula:

$$\text{IDF} = \log \left(\frac{|D|}{D_t} \right). \tag{6}$$

The total weight of terms in individual documents relative to the entire document set is calculated by the following formula:

$$V_{t,d} = \text{TF} \cdot \text{IDF}. \tag{7}$$

It should be noted that equation (7) assesses the importance of a term based only on its frequency of occurrence in the document, without regard to the order of the terms.

Latent semantic analysis (LSA) was also used to reduce the dimensionality. LSA is a method to learn the implicit relationship between from massive text data and then obtain the expression characteristics of documents and words. The basic idea of this method is to comprehensively consider which documents will some words appear at the same time, so as to determine the similarity between the meaning of the word and other words. By first constructing a word document matrix M, find the low-rank approximation of the matrix to mine the association relationship between. The process is mainly divided into four steps: calculating the word document matrix, singular value decomposition, selecting the first k feature roots and the corresponding feature vectors to reconstruct the matrix M, and mining the semantic relationship using the correlation coefficient matrix. Term space has also been used for LSA using singular matrix decomposition [20], flow mutual information (PMI) [21] (a measure of association), and conditional random fields (CRF). (a generalization of the latent Markov model). Some studies [22] have applied statistical criteria and relative entropy to probability distributions, called information amplification factors, or Kullback-Leibler divergence.

The following is a summary of these methods, including the advantages and disadvantages.

The naive Bayes approach (NB) refers to the probabilistic classification method setting $P(c_i|d)$ as the probability of the document. The $i = 1, \cdots, |C|$ represented by the vector $d = (t_1, \cdots, t_n)$ corresponds to the category c_i. The task of the classifier is to find such values of c_i and d that the probability $P(c_i|d)$ is the maximum.

$$\text{CSV}(d) = \arg \max_{c_i \in C} P(c_i|d). \tag{8}$$

To calculate the value of $P(c_i|d)$, use Bayes' theorem.

$$PP(c_i|d) = \frac{P(c_i)P(d|c_i)}{P(d)}. \tag{9}$$

Calculating $P(c_i|d)$ is difficult because of the large number of features $t_1, \cdots, t_n$, so we make the "naive" assumption that any two coordinates are considered random and are statistically independent of each other. Then, one can use the following formula:

$$P(d|c_i) = \prod_{k=1}^{n} P(t_k|c_i). \tag{10}$$

All probabilities are then calculated using the maximum likelihood method.

4. Programme of This Paper

4.1. KNN. The k nearest neighbour (KNN) method refers to the metric classification method. In order to find the category corresponding to document d, the classifier compares d with all documents in the training sample L, i.e., for each $d_z \in L$, it calculates a distance $\rho(d_z, d)$. The k documents closest to d are then selected from the training set. According to the k nearest neighbour method, document d is considered to belong to the category that is the most common among the neighbours of a given document, i.e., for each category c_i is computed as a ranking function.

$$\text{CSV}(d) = \sum_{d_z \in L_k(d)} \rho(d_z, d).\Phi(d_z, c_i), \tag{11}$$

where $L_k(d)$ is the k closest documents between L and d and $\Phi(d_z, c_i)$ is a known value, which is the document already classified in the training sample.

4.2. Neural Networks. In order to classify the category classification, the following learning is carried out. First, the model uses the CNN, which is similar to the VDCNN. As a dataset in 50 dimensions, the set of different news articles is classified as the dataset of the transitional and the transition destination and the genre is classified as a category in order to have the size of the embedded vector.

A VDCNN with a convolution layer of 17 layers is referenced using chainer ∗1 ∗. The length of the input string is 1024 characters. The initial value of he is used to initialize the convolution layer of the network. In the following, the layers of the implemented VDCNN are assumed to be embed1, conv1, re2, RE5, FC6, and fc8 from the lower layer. The stochastic gradient descent method is used for the learning of a network containing four convolutional layers in each res layer.

In view of the limitations of current research methods, in order to accurately detect and identify the structural features of domain knowledge, this paper proposes a research framework based on the deep graph neural network learning representation method and the specific research process is shown in Figure 1. The proposed research framework consists of the following parts: data preprocessing module, feature extraction module, graph network model module, and domain knowledge structure visualization module.

The weight and bias of FC6 to fc8 are randomly initialized. The transition weight and bias are tested in two cases:

5. Experimental Results and Considerations

5.1. Dataset. The dataset is shown in Table 1.

The AFPBB dataset has four categories: news, environment sclence it, fifties, and sports. The livedoor dataset uses four categories (news, life hack, livedoor Homme, and sports watch), which are considered to be relatively close to AFPBB data.

5.2. Result. Learning the network of transponders is performed by AFPBB datasets. The dataset of the destination is the livedoor dataset. In this case, we compared the correct
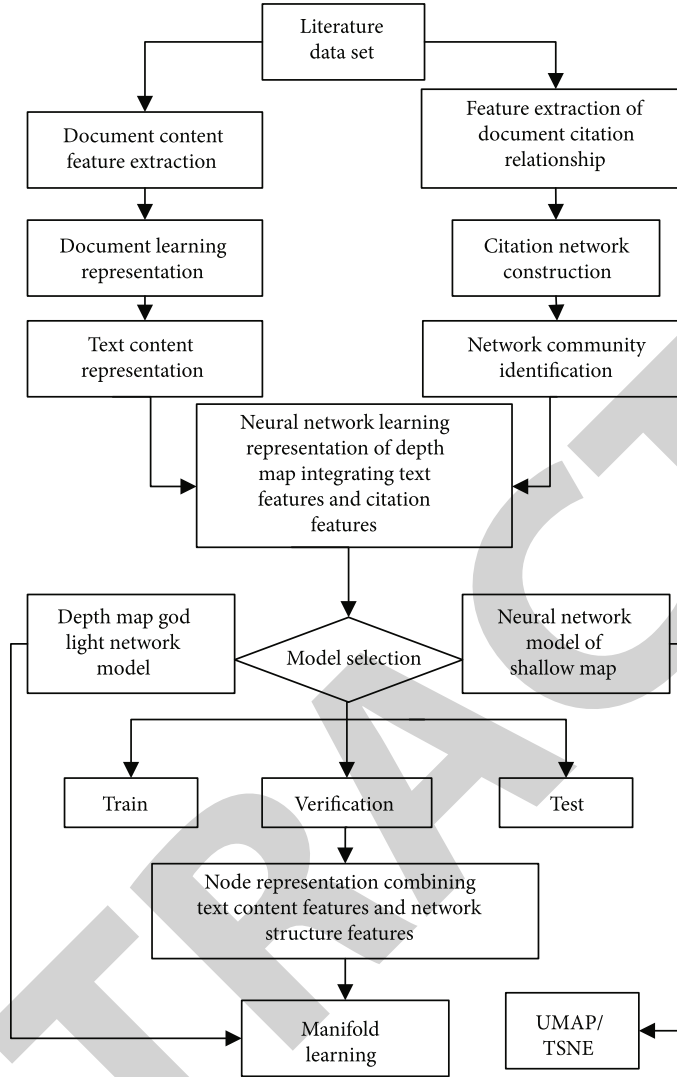
Figure 1: Specific flow chart.

Table 1: Dataset used in Table 1.

	Dataset name	Number of training data	Number of test tables	Number of categories
Transition sources	AFPBB	46800	5200	4
Transferring destinations	Livedoor	2700	300	4

solution for the test data in the two cases described. In addition, in order to clarify the effect of the transition learning in the case where the number of data in the transition destination is small, the comparison of the number of the training data of the transition to the first 1/2, 1/4, 1/8, and 1/16 was also compared. The results of comparison of the correct answers to test data are shown in Table 2 and Figure 2.

As shown in Figure 2, the inaccuracy of Init is the most accurate for all data sizes. In the VDCNN, it is very effective to use transfer learning between similar datasets at least. On the other hand, when the weight and bias after the transition are fixed, the accuracy is lower than that of scratch and the inverse effect is reversed in the other case. Moreover, the

Table 2: Average answer to test data.

Model	Original	1/2	1/4	1/8	1/16
Scratch	0.87	0.75	0.73	0.61	0.5
Fixed	0.8	0.76	0.74	0.69	0.65
Init	0.94	0.92	0.89	0.87	0.8

accuracy of scratching without reducing the data size and the accuracy of scratching are almost the same when the data size is 1/8. In addition, as the data size decreases, the difference in the accuracy of the scratches and Init increases [23, 24].
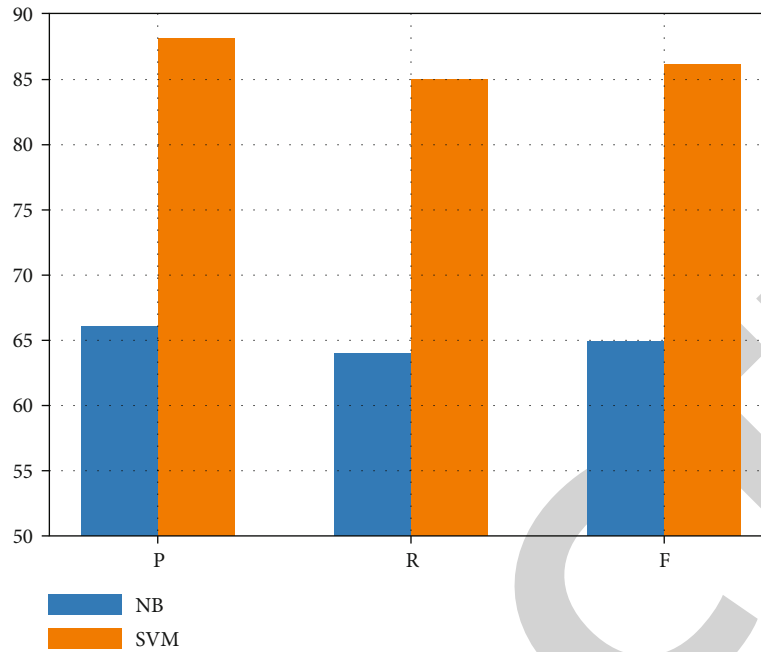
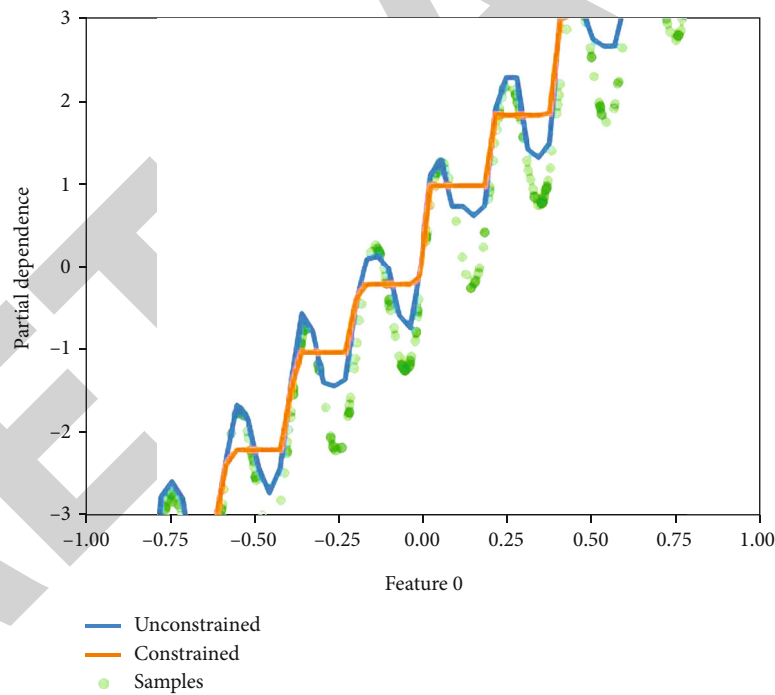Figure 2: Experimental results of "machine learning" dataset.

Figure 3: Results of the division of direct reference network communities in the field of physical disciplines.

As can be seen in Figure 3, the citation networks in both the physics and blockchain domains exhibit a clear modular structure. Specifically, the modularity of the direct citation network structure for the physics discipline is 0.81, resulting in 12 communities, while the modularity of the citation network for the blockchain domain is 0.46, resulting in 8 communities. It should be noted that the results of subsequent document representation learning models as well as neural network models will be based on the results of citation network community classification. The coloring in the visualization stage of the domain knowledge network structure and the labeling in the graph neural network model will use the results of the citation relationship community delineation of the document nodes as a reference. The results of
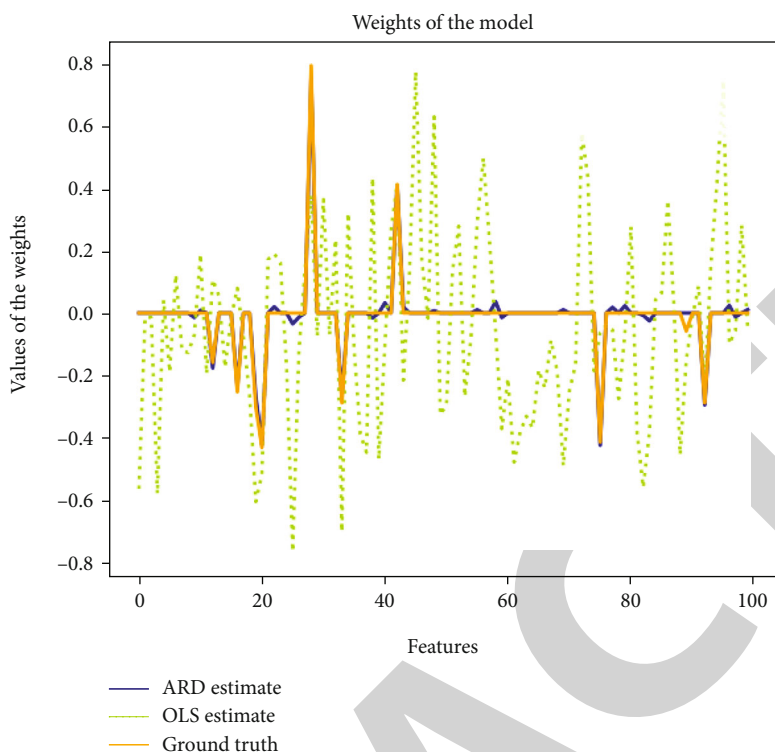
Figure 4: Results of UMAP-based dimensionality reduction for textual content visualization in the physical discipline.

domain text content analysis obtained based on document representation learning and stream learning algorithms are shown in Figure 4.

As shown in Figure 4, comparing the two visualizations, we find that the citation communities in physics show better aggregation characteristics, while the content analysis results in the "blockchain" domain are more inconsistent with the citation network community results, showing a lower aggregation of similar citation communities in terms of content. After coloring by citation network communities, it is more obvious that document representation learning can indeed characterize and measure the knowledge structure of more mature physical disciplines well. However, for the emerging "blockchain" domain, the results obtained from document representation learning and citation network structure community segmentation methods are relatively different. This may be due to the fact that the emerging field is at an early stage of exploration and the trend of integration with other fields is emerging but no clear topic or subfield has been formed yet.

6. Conclusions

Text classification of data in the Internet can solve the problem of information clutter to a greater extent and discover data containing domain knowledge. This paper proposes a feature word-based text classification method for web domain knowledge, using the support vector machine as the basic classification algorithm. In the future study, we can consider adding other features (such as word position, dependent syntax, and semantic roles) to verify the effects of different features or feature combinations on the classification of knowledge text. Experiments show that the performance of this method is improved by 3% compared with the benchmark method.

Data Availability

The dataset used in this paper are available from the corresponding author upon request.

Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work.

References

[1] X. Sheng, X. Wu, and Y. Luo, "A novel text mining algorithm based on deep neural network," in *2016 international conference on inventive computation technologies (ICICT)*, vol. 2, pp. 1–6, Coimbatore, India, 2016, August.

[2] R. Ali, S. Lee, and T. C. Chung, "Accurate multi-criteria decision making methodology for recommending machine learning algorithm," *Expert Systems with Applications*, vol. 71, pp. 257–278, 2017.

[3] A. K. Mukhopadhyay and S. Samui, "An experimental study on upper limb position invariant EMG signal classification based on deep neural network," *Biomedical Signal Processing and Control*, vol. 55, p. 101669, 2020.

[4] X. Wei, H. Lin, L. Yang, and Y. Yu, "A convolution-LSTM-based deep neural network for cross-domain MOOC forum post classification," *Information*, vol. 8, no. 3, p. 92, 2017.

[5] A. B. Wilcox and G. Hripcsak, "The role of domain knowledge in automating medical text report classification," *Journal of the American Medical Informatics Association*, vol. 10, no. 4, pp. 330–338, 2003.

[6] G. Cai, Y. Fang, J. Wen, S. Mumtaz, Y. Song, and V. Frascolla, "Multi-carrier *M*-ary DCSK system with code index modulation: an efficient solution for chaotic communications," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 6, pp. 1375–1386, 2019.

[7] C. C. Aggarwal and C. Zhai, "A survey of text classification algorithms," in *Mining Text Data*, pp. 163–222, Springer, Boston, MA, 2012.

[8] A. Genkin, D. D. Lewis, and D. Madigan, "Large-scale Bayesian logistic regression for text categorization," *Technometrics*, vol. 49, no. 3, pp. 291–304, 2007.

[9] K. Chandra, A. S. Marcano, S. Mumtaz, R. V. Prasad, and H. L. Christiansen, "Unveiling capacity gains in ultradense networks: using mm-wave NOMA," *IEEE Vehicular Technology Magazine*, vol. 13, no. 2, pp. 75–83, 2018.

[10] A. Radwan, M. Fátima Domingues, and J. Rodriguez, "Mobile caching-enabled small-cells for delay-tolerant e-Health apps," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 103–108, Paris, France, 2017.

[11] V. Sindhwani and P. Melville, "Document-word co-regularization for semi-supervised sentiment analysis," in *2008 eighth IEEE international conference on data mining*, pp. 1025–1030, Pisa, Italy, 2008, December.

[12] F. B. Saghezchi, A. Radwan, J. Rodriguez, and T. Dagiuklas, "Coalition formation game toward green mobile terminals in heterogeneous wireless networks," *IEEE Wireless Communications*, vol. 20, no. 5, pp. 85–91, 2013.

[13] P. An, Z. Wang, and C. Zhang, "Ensemble unsupervised autoencoders and Gaussian mixture model for cyberattack detection," *Information Processing & Management*, vol. 59, no. 2, p. 102844, 2022.

[14] J. Attenberg, P. Melville, and F. Provost, "A unified approach to active dual supervision for labeling features and examples," in *Joint European conference on machine learning and knowledge discovery in databases*, pp. 40–55, Berlin, Heidelberg, 2010, September.

[15] R. Bekkerman, H. Raghavan, J. Allan, and K. Eguchi, "Interactive clustering of text collections according to a user-specified criterion," *IJCAI*, vol. 7, pp. 684–689, 2007.

[16] S. Palanisamy, B. Thangaraju, O. I. Khalaf, Y. Alotaibi, S. Alghamdi, and F. Alassery, "A novel approach of design and analysis of a hexagonal fractal antenna array (HFAA) for next-generation wireless communication," *Energies*, vol. 14, no. 19, p. 6204, 2021.

[17] P. Melville and V. Sindhwani, "Active dual supervision: reducing the cost of annotating examples and features," *NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pp. 49–57, 2009.

[18] S. Poulis and S. Dasgupta, "Learning with feature feedback: from theory to practice," in *Artificial Intelligence and Statistics*, pp. 1104–1113, PMLR, 2017.

[19] W. Peng, T. Sun, S. Revankar, and T. Li, "Mining the "voice of the customer" for business prioritization," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 3, no. 2, pp. 1–17, 2012.

[20] P. Yang and W. Gao, "Information-theoretic multi-view domain adaptation: a theoretical and empirical study," *Journal of Artificial Intelligence Research*, vol. 49, pp. 501–525, 2014.

[21] J. Attenberg, P. Melville, and F. Provost, "Guided feature labeling for budget-sensitive learning under extreme class imbalance," *ICML Workshop on Budgeted Learning*, 2010.

[22] P. Wang and C. Domeniconi, "Towards a universal text classifier: transfer learning using encyclopedic knowledge," in *2009 IEEE international conference on data mining workshops*, pp. 435–440, Miami, FL, USA, 2009, December.

[23] F. A. Al-Mekhlafi, R. A. Alajmi, Z. Almusawi et al., "A study of insect succession of forensic importance: Dipteran flies (diptera) in two different habitats of small rodents in Riyadh City, Saudi Arabia," *Journal of King Saud University-Science*, vol. 32, no. 7, pp. 3111–3118, 2020.

[24] A. Algalil, A. Fahd Mohammed, and S. P. Zambare, "New species of flesh fly (Diptera: Sarcophagidae) Sarcophaga (Liosarcophaga) geetai in India," *Journal of Entomology and Zoology Studies*, vol. 4, no. 3, pp. 314–318, 2016.