INTERNATIONAL JOURNAL OF CIRCUITS, SYSTEMS AND SIGNAL PROCESSING
DOI: 10.46300/9106.2021.15.25 Volume 15, 2021

A Filtering Algorithm of Main Word Frequency for Online Commodity Subject Classification in E-Commerce

Zhenfeng Wei[1], Xiaohua Zhang[2]

[1]School of International Business, Zhejiang Industry and Trade Vocational College, Zhejiang, Wenzhou325003, China

[2]School of Management, Zhejiang Dongfang Polytechnic, Zhejiang, Wenzhou325000, China

Received: January 31, 2021. Revised: February 25, 2021. Accepted: March 10, 2021. Published: March 30, 2021.

***Abstract*—Based on the traditional classification of plain text in E-Commerce, this article has put forward a processing method in accordance with semi-structured data and main information in web pages, which enhances the accuracy of the product distribution. On the basis of the traditional text-mining, combined with the structure and links of web page, this article has proposed an improved web page text representation model in E-Commerce based on supporting vector machines and web text classification algorithm, but there are still a lot of shortcomings waiting for further improvement. According to the data contrast in precision ratio, recall ratio and F-measure, the effect of the improved experiment with LDF-IDF is comprehensively better than that of tf-idf. The precision rate in certain classification can reach 100%, but there is low precision rate caused by items with fewer samples or samples fuzziness. Therefore, the classification of the correct category will directly affect the effect of classification.**

***Keywords*—Term Frequency, Text classification, Web subject classification, Filtering Algorithm**

I. INTRODUCTION

THE rapid development of Internet leads the dramatic growth of the network information resources of online goods in E-commerce. Therefore, the information overload of online goods in E-commerce has become a urgent issue in the E-commerce era. It's important that the useless information need to be rejected and the right information of online goods in E-commerce need to be searched quickly on the correct and effective classification of different Web pages.

Currently, many scholars have worked on the research about webpage classification of online goods in E-commerce and made some progress. The common classification methods mainly include decision tree method[1-2], naive Bayesian method[3], Support Vector Machine [4-6], k-Nearest Neighbor[7-8], Rocchio's algorithm [9] and so on. Herrouz, A. proposed a method [10] for web text classification based on SVM to perform separate single value decomposition for local LSA.

All of these methods have made some positive progress in improving the webpage classification[11-12]. Although the current classification methods have classified the Web pages quite well, there are still some problem needs improving. Finally, this paper adopted the SVM as the text sorting algorithms, through comparing with TF-IDF, it proves the improved LTF-IDF has better effect on web page classification.

II. IMPROVED CLASSIFICATION ALGORITHM

Web page subject classification is a typical application of text classification in the field of Internet[13]. Its core technology is consistent with the text classification, but needs to consider the difference between it and the ordinary text during the classification process. TF-IDF is one of the key algorithms for text classification and provides the basis for better formulation. Document presentation empowers the computer to recognize a classified page, and common methods include Boolean model and vector space model [14-15]. Boolean model sets all the characters of Web pages as a binary parameter, 1 representing existence and 0 representing inexistence. Boolean model is simple but too rough to suit this application due to that it can't distinguish between feature item's weighting in a web page about web page subject classifications. Currently the VSM(Vector Space Model) is mainly adopted to present text in the text classification.

Through presenting documents as feature vector, the elements of VSM not only described a feature's existence in the document, but also described the weights of features in the document. VSM is stated as $V(d)$,

$V(d)=(t_1,w_1(d),...t_n,w_n(d))$. $V(d)$ is a feature vector of document, $t_i(i=1,...n)$ is a group without repeated features, $w_i(d)(i=1,...n)$ is the weight of feature t_i in documents, with a definition of frequent function in a document. At present, this function adopts TF-IDF (term frequency inverse document frequency). The importance of word rises in direct ratio of its frequency of appearance, and the weight calculation formula of words is shown as below

$$w_i(d)=tf_i(d)*\log(\frac{N}{n_i}+0.01) \quad (1)$$

Where $w_i(d)$ shows the word *i* weight in the text d , $tf_i(d)$ is the occurrence frequency of word *i*, in documentd, *N* is the total number of the word in one document, n_i is the total number of documents that contain the word.

Classification function is based on VSM, TF-IDF's results to realize the calculation, on the basis of consistent presentation mode and the weight function is essential which directly affects the classification results. But TF-IDF is based on non-structural design of plain text without considering characteristics of semi-structured web data. Based on arithmetic of plain text TF-IDF, this article has provided improved solution combined with the semi-structured feature of web page.

A. Analysis of Web Page Structure

The web page has a semi-structured feature based on HTML (Hypertext Markup Language) technology consisting of HTML tags and text descriptions with its structure shown below

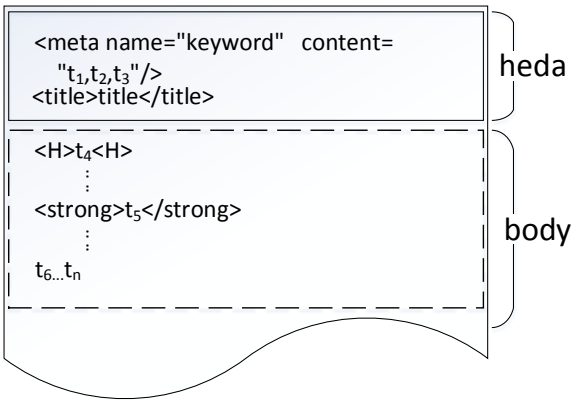

Fig. 1 Structure of web page

Web page consists of two big parts of head and body, marked respectively through <head></head> and <body></body>. The head section of a web page contains a relatively fixed structure including the page's keywords, in the position of <head><meta name = "keyword" >; the title of the page, locates in position of <head><title>, and words appeared in the positions of head-meta\head-title is suitable with the page theme and needs to take weighting into consideration during the frequency weighting calculation. The body part contains main text and all content, where the appeared words needs to be stress through different modifier such as through standard markers of <h1>\<h2>\ \<B>, and may extend the unique marker of theme through CSS. Words appear in these locations are always in deep relationship with the main theme and information needed more user concern.

Different from unstructured text documents, data constituting the web page has not only visible parts directly facing readers, but also invisible parts facing search engines. At the same time, the web page can display multimedia data, including text, image, audio and video. How to combine features of web page's structure to improve classification results is the problem researched in this article.

B. LTF-IDF Weighting Algorithm based on Location Gain

This paper describes the location of web pages in a way similar to xpath, and uses "\" to link different levels of tags to a path to define the page location. The Web page path diagram is as follows.

```
<html>
    <head>
        <meta name="keyword" contant="t1,t2,t3"/>
        <title>t4</title>
    </head>
    <body>
        <table>...<h1>t5</h1>
        <talbe>...<h2>t6</h2>
        ...t7,...tn
</body>
</html>
```

Fig. 2 Web page path diagram

As shown in the above, there are web location path as follow:

l1 = html\head\meta_keyword

l2 = html\head\title

l3 = html\body\table\...\h1

l4 = html\body\table\...\h2

…

lm = html\body\other

L1position is defined as keyword in the page of above figure, and L2 position is defined as title, then we can get a conclusion that words in these two positions have very high correlation with the web page's theme. L3\L4 positions also have a high correlation. Now we use collection L to present ($L = \{l_1, l_2, \ldots l_n\}$) the location in the web page, gain influence factor of each position to word frequency is $K = \{k_1, k_2, \ldots k_n\}$. Describe the web page location-gain as vectors $g(d) = \{l_1 : g_1(d), \ldots l_n : g_n(d)\}$.

$g_j(d)$ is gain of position l_n to words may be a mapping function from a place to gain, let $g: l_j \rightarrow k_j$

Assuming a frequency $tf_i(d, l_j)$ of the feature i appeared in the page location l_j, and then the calculation formula of feature i frequency is

$$tf_i(d) = \sum_{j=1}^{n} tf_i(d, l_j) = \sum_{j=1}^{n} tf_i(d, j) * g_j(d) = \sum_{j=1}^{n} tf_i(d, j) * k_j \quad (2)$$

$tf_i(d)$ is the gain frequency in the page position; $tf_i(d, j)$ is the word frequency in the location l_i, $tf_i(d, l_j)$ is the word frequency after gaining in the page position, after determining to map function, its value is $tf_i(d, j) * k_j$. Then the calculation formula of word's gain weight in the position is

$$W_i(d) = \sum_{j=1}^{n} tf_i(d, j) * k_j * \log(\frac{N}{n_i} + 0.01) \quad (3)$$

The normalization formula is shown as below

$$W_i(d) = \frac{\left(\sum_{j=1}^{n} tf_i(d, j) * k_j\right) * \log(\frac{N}{n_i} + 0.01)}{\sqrt{\sum_{i=1}^{m} (\sum_{j=1}^{n} tf_i(d * k_j) * \log(\frac{N}{n_i} + 0.01))^2}} \quad (4)$$

C. Complexity Assessment

The modified TF-IDF algorithm is basically the same with that of LTF-IDF algorithm, with an only addition of a location to influence factor, so there is no change of TF-IDF's time complexity.

III. Experiment Procedure

A. Selection of classification algorithm

This article selects one from algorithm of KNN(k-Nearest Neighbor)[16-18], support vector machine as a verified classification algorithm[19-20]. In the VSM expression, it will calculate the similarity of unspecified classified and training samples to find *K* nearest neighbors similar to the unspecified samples and determine their affiliation in accordance with the category of them. Use the similarity representing unspecified documents and training text, and the similarity can be presented through Euclidean distance (formula 5) or the cosine similarity (formula 6)

$$sim(d, d_i) = \frac{1}{|d - d_i|} = \frac{1}{\sum_{j=1}^{n} (x_j - x_{i,j})^2} \quad (5)$$

$$sim(d, d_i) = \cos(\theta) = \frac{\sum_{j=1}^{n} \left(x_j * x_{i,j}\right)}{\sqrt{\sum_{j=1}^{n} \left(x_j\right)^2} * \sqrt{\sum_{j=1}^{n} \left(x_{i,j}\right)^2}} \quad (6)$$

The cosine similarity will eliminate standard differences caused by scales and other factors, and will be adopted for the experiment. After obtain K nearest positions, weighting through calculation for unspecified text and classification

$$p(d, c_i) = \sum_{j=1}^{k} sim(d, d_i) y(d_j, c_i) \quad (7)$$

$y(d_j, c_i)$ is function of category property, when d_j belongs to the classification, the value is 1, otherwise it is 0. Determine document's classification according to the degree of scores at last.

In the case of inseparable linearly, supporting vector machine will finish calculation in the low dimensional space, mapping the input space to high-dimensional feature space through the kernel, building the optimal separating hyperplane in the high-dimensional feature space to separate nonlinear data which is not easy to separate in plane. As shown in Figure 2, Data in a two-dimensional space can't be divided, so it has to map into three dimensional space for division. Function realizing the map is called kernel.

Theory of KNN's classification has a high accuracy, but a large calculation. When the vector dimensions and sample size are larger, KNN can't work effectively, but the SVM is superior to KNN on the classification effect and efficiency. After verified comparison, this paper chooses the SVM as classifier to verify the classification effect of LTF-IDF.

B. Experimental Procedure

This article designs the following processes to verify the effect of improved algorithm, and the experiments have two tests, one is calculating feature weight by the traditional TF-IDF, another is calculating feature weight by LTF-IDF, and then compared effects of classification to determine the effect of improved algorithm.

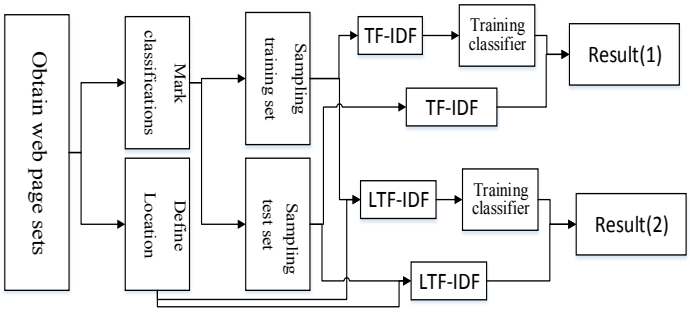

Fig. 3 process diagram

Step 1. Obtain the collection of Web pages for experiments, the plan is designed to automatically extract from the catering web site through a web crawler;

Step 2. Mark obtained pages and complete the location definition with the features of web pages;

Step 3. Complete the annotated document, extract 80% of each category from the marked documents as the training set, and the rest as test set A;

Step 4. Adopt traditional weight calculation for TF-IDF classification tests, and record effect of classification;

Step 5. Adopt improved weighting algorithm for LTF-IDF classification tests, and record effect of classification;

Step 6. Compare from three aspects in precision ratio, recall ratio and f-comparison to complete the experiment.

Experimental procedure is as follows.

Step 1. Page preprocessing

Use Beautiful soup library of Python to preprocess web page text and access to the tree structure of document.

Firstly, analyze the text structure of web page, define HTML structure irrelevant to the text subject, and then delete the extraneous content. For example: (this is an advertisement position).

Delete some ultrahigh frequent words. For example, in the navigation page on the head of web <div the page id = "header" > , and information of friend link at the tail information within, as well as description of the recommendation dish's price (Yuan/dish), these words will lead to useless calculation in the collection.

Delete numbers in the text. Number of the web page text is the mathematical description of food prices and telephone number, which has no contribution to the subject classification.

Delete some low frequent words. Some words like only exit in one or two texts, reservation of this kind of words will lead to a higher vectorial dimension making the calculation complex and no contribution to the subject classification.

Preprocess the web pages including main segmentation of the web page, removing irrelevant noisy information and stop words for classification of web page. For example, the body content of a web page is that " This is a restaurant ", the result got after word segmentation is series of phrases as "this is a Hunan cuisine ", and the result after removing noise and common words is "Hunan cuisine". By introducing language databases such as the Sogou lab's common beverage words, China's North-South ham-food name, eight big cuisines menu list and the catering complete corpus, it effectively reduces the phenomena that classics word is separated by segmentation system. Chinese pretreatment module calls jieba segmentation system to processing, below is a segmentation show of a typed text.

Original text: recommended dish: flavor radish (12 Yuan/dish); delicious black fungus (12 Yuan/dish); health big pot (48 Yuan/dish); Hunan water organic live fish (78 Yuan/dish); garlic maotai-flavor elbow (69 Yuan/dish); garlic spicy crab (98 Yuan/dish); fence fragrance wings (38 Yuan/dish); steamed fish head with diced hot red peppers (88 Yuan/dish);

Text after segmentation: recommended food/ flavors/ radish/ black fungus / health/ big port/Hunan water/organic fish/ garlic/ maotai-flavor elbow/ garlic / spicy crab/ fence/ fragrance wings/ steamed fish head with diced hot red peppers.

The original text of "Yuan/dish" and figures number such as "12" have no sense to help with the subject classification, so add these words as stop words in the stop word list. If the word after segmentation still appears in the stop word list, then it should be deleted. Due to the introduction of the sogou lab common beverage words, the word of "steamed fish head with diced hot red peppers" is not divided into "chopped chili pepper" and "fish heads".

Combines location features of feature items to calculate the feature item's frequency in every web page of the training set. Count appearance times of feature items in the web page of training set, if the feature item appears in anchor text, then multiply 2 times the calculated number; If the feature is on <title></title>, and<h1></h1>, <h2></h2>, and , then multiply 1.5 times the word frequency; if the feature time is in body text, then multiply 1 times the word frequency, if the features item appeared in other places, then multiply 0.5 times the feature item word frequency.

This article's probability estimated method is based on improved TF-IDF algorithm, considering location ' s weighting, such as the features word "spicy" has appeared totally in 100 texts (total amount of text is 4000), and has appeared 1 time in the anchor text of text A, 1 time in the subject, 2 times in body descriptions and 3 times in other place

such as the example comments, then the TF (Term Frequency) value of the features word "spicy" in text A is: 1*2+1* 1.5+2*1+3*0.5=7,IDF (Inverse Document Frequency) value is: then the TF-IDF value of feature word "spicy" is: 11.215179859653.

Step2. Feature dimension reduction

Select the first 10 feature items of high weight in each web page of the training set and set their weight as text feature vector form the web page, LiLi fishing port (east labour road branch), improved feature weight extract algorithm to get 10 words as: fishing port, wild, duck's cry, freshwater fishes, mandarin fish, cut meat, Celebrity Retreat, sky peak, griddle, Huidu fish. Consolidated all page text feature vectors on the training web pages and listed according to their weights from large to small, select top 1000 feature items and feature vector constituted by its weight and start training classifier by the supporting vector machine.

Step 3. Calculation test sets and text vector quantity

Use 1000 items of features as dictionary, calculate the initial text vector of the test text, and calculate the final test set text vector (each vectorial dimension is 1000-dimension) based on new text representation model proposed in the third chapter.

Step 4. Classification by SVM

Adopt LIBSVM software package of Python of which the kernel function applies RBF(Gaussian radial basis function), and the gamma value of the parameter is set to 0.111111, the parameter c is set to 0.01.

C. Simulation of Experiment

This article adopts pages of restaurants as a data source and obtains Web pages and classifies according to the style of cooking which is divided into 9 kinds of Hunan cuisine, Home dish, Farm food, Chafing dish, Sichuan cuisine, Cantonese cuisine, Barbecue, Seafood and the Private cuisine, with a total collection of 3,994 page documents, training set consisting of random 70% of each web page, and test set consisting of the rest 30% part, 2,794 web page documents of training set, and 1200 web page documents of test set. Compare by 3 different evaluation criteria (recall ratio, precision ratio, and f-measure) and a typical feature selection method. The programming language used Python. LIBSVMis a general SVM package designed Dr. Chih JenLin from Taiwan University which is easy to operate and use, fast and efficient. Python version of LIBSVM is adopted in this experiment.

IV. Experiment result

We have compared algorithms of tf-idf and ltf-idf using the same data set and their precision ratio, recall ratio and F-measure values are given as Table 1, Table 2 and Table 3.

Table 1. Comparison of recall ratio

Category	**tf-idf**	**ltf-idf**
HN	0.900900901	0.930930931
FF	0.867256637	0.898230088
H	0.828571429	0.914285714
CF	0.884816754	0.921052632
S	0.733333333	0.866666667
C	0.722222222	0.888888889
BBQ	0.833333333	1
SF	0.765957447	0.85106383
P	0.894495413	0.926605505
Average	**0.825654163**	**0.910858251**

Table 2. Comparison of precision ratio

Category	**tf-idf**	**ltf-idf**
FF	0.9375	0.959752322
H	0.827004219	0.871244635
CF	0.899224806	0.977099237
S	0.965714286	0.988700565
C	0.323529412	0.481481481
BBQ	0.448275862	0.551724138
SF	0.833333333	1
P	0.923076923	0.952380952
FF	0.866666667	0.89380531
Average	**0.780480612**	**0.852909849**

Table 3. Comparison of F-measure value

Category	**tf-idf**	**ltf-idf**
FF	0.918836141	0.945121951
H	0.846652268	0.88453159
CF	0.862453532	0.944649446
S	0.923497268	0.953678474
C	0.448979592	0.619047619
BBQ	0.553191489	0.680851064

SF	0.833333333	1
P	0.837209302	0.898876404
FF	0.880361174	0.90990991
Average	0.789390455	0.870740718

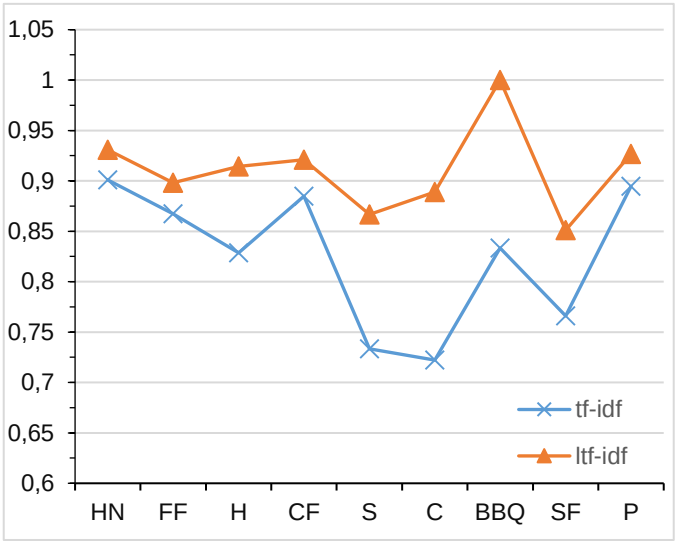

Fig. 4 Comparison of recall ratio

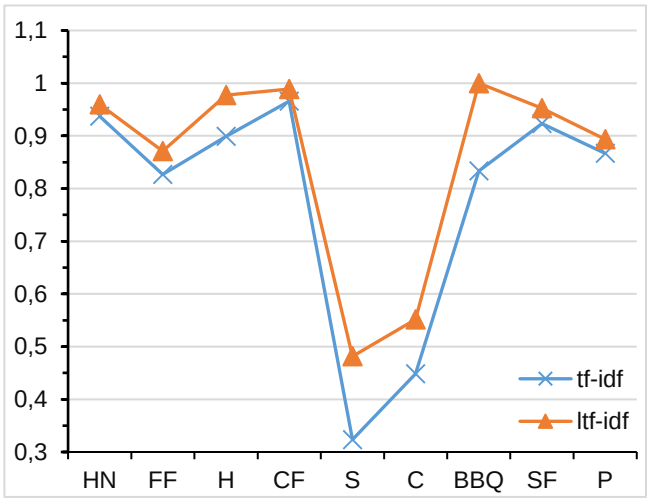

Fig. 5 Comparison of precision ratio

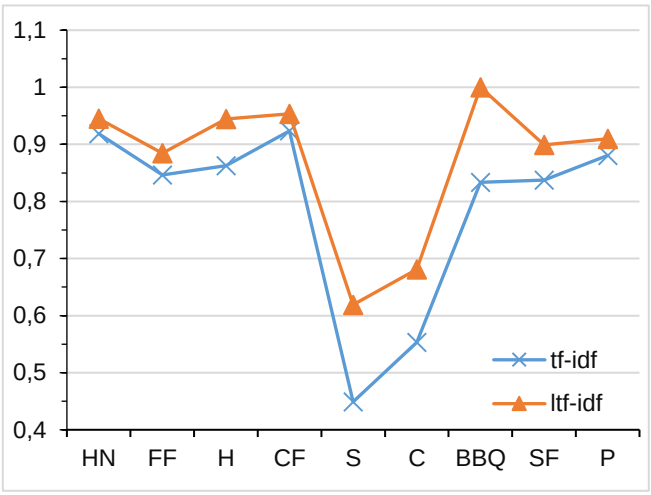

Fig. 6 Comparison of F-measure value

According to the data contrast in precision ratio, recall ratio and F-measure, the effect of the improved experiment with ltf-idf is comprehensively better than that of tf-idf. The precision rate in certain classification can reach 100%, but there is low precision rate caused by items with fewer samples or samples fuzziness. Therefore, the classification of the correct category will directly affect the effect of classification.

V. Conclusion

The Internet has become the world's largest open public data sources and the web classification directly affects the user experience and effects of using the Internet for information in E-Commerce, but with the development of related technologies, studies will have also continued to emerge. A novel algorithm is proposed for the classification accuracy with semi-structured Web page. Based on feature dimension reduction process does not take into account the semantic influence on feature selection. How to make full use of knowledge to improve the classification results in natural language understanding, and this is one that requires further study. The effect of Web text classification depends not only on classification algorithms but also under the influence of information technology, such as segmentation, feature extraction, and so on. How to effectively and accurately combine topic characteristics to segment is one of the next study directions.

ACKNOWLEDGEMENTS

This work is supported by the General Scientific Research Projects of Zhejiang Provincial Department of Education in 2019. "*Research on the integration of ideological and political education in online courses of colleges and universities in Zhejiang Province under the background of big data*." (NO.Y201942816)

REFERENCES

[1] Lewis, D. D., Schapire, R. E., Callan, J. P., & Papka, R. (1996, August). Training algorithms for linear text classifiers. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 298-306). ACM.

[2] Baykan, E., Henzinger, M., Marian, L., & Weber, I. (2011). A comprehensive study of features and algorithms for url-based topic classification. ACM Transactions on the Web (TWEB), 5(3), 15.

[3] Eyheramendy, S., Lewis, D. D., & Madigan, D. (2003). On the naive bayes model for text categorization.

[4] Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. Neural Networks, IEEE Transactions on, 13(2), 415-425.

[5] Saraç, E., & Ozel, S. A. (2013, June). Web page classification using firefly optimization. In Innovations in Intelligent Systems and Applications (INISTA), 2013 IEEE International Symposium on (pp. 1-5). IEEE.

[6] Herrouz, A., Khentout, C., & Djoudi, M. (2013). Overview of web content mining tools. arXiv preprint arXiv:1307.1024.

[7] Radinsky, K., & Horvitz, E. (2013, February). Mining the web to predict future events. In Proceedings of the sixth ACM international conference on Web search and data mining (pp. 255-264). ACM.

[8] Yadav, M., & Mittal, M. P. (2013). Web Mining: An Introduction. International Journal of Advanced Research in Computer Science and Software Engineering, 3(3), 683-687.

[9] Thi Mai Le, Shu-yi Liaw, My-Trinh Bui, The Role of Perceived Risk and Trust Propensity in the Relationship between Negative Perceptions of Applying Big Data Analytics and Consumers?Responses, WSEAS Transactions on Business and Economics, Volume 17, 2020, Art. #41, pp. 426-435.

[10] Manchanda, P., Gupta, S., & Bhatia, K. K. (2012). On The Automated Classification of Web Pages Using Artificial Neural Network. IOSR Journal of Computer Engineering (IOSRJCE), 4(1), 20-25.

[11] Vasily S. Starostin, Veronika Yu. Chernova, Elena A. Fedorenko, Potential of Export-Oriented Import Substitution in the Eurasian Economic Union: the Case Study of the Agro-Industrial Complex, WSEAS Transactions on Business and Economics, Volume 16, 2019, Art. #17, pp. 145-152.

[12] Özel, S. A. (2011, June). A genetic algorithm based optimal feature selection for web page classification. In Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on (pp. 282-286). IEEE.

[13] Mangai, J. A., & Kumar, V. S. (2011). A novel approach for web page classification using optimum. IJCSNS, 11(5), 252.

[14] Kamal, A. (2013). Subjectivity Classification using Machine Learning Techniques for Mining Feature-Opinion Pairs from Web Opinion Sources. arXiv preprint arXiv:1312.6962.

[15] Golub, K. (2011). Automated subject classification of textual documents in the context of Web-based hierarchical browsing. Knowledge organization, 3(38), 230-244.

[16] Patil, G., & Patil, A. Web Information Extraction and classification using Vector Space Model Algorithm. International Journal of Emerging Technology and Advanced Engineering, ISSN, 2250-2459.

[17] Thijs, B., Zhang, L., & Glänzel, W. (2013, July). Bibliographic Coupling and Hierarchical Clustering for the validation and improvement of subject-classification schemes. In 14th international conference on scientometrics and informetrics (pp. 237-250). International Society of Scientometrics and Informetrics Vienna.

[18] Jain, Y. K., & Wadekar, S. (2011). Classification-based Retrieval Methods to Enhance Information Discovery on the Web. International Journal of Managing Information Technology (IJMIT) Vol, 3.

[19] Tan, S. (2006). An effective refinement strategy for KNN text classifier. Expert Systems with Applications, 30(2), 290-298.

[20] Shin, K., Abraham, A., & Han, S. Y. (2006). Improving kNN text categorization by removing outliers from training set. In Computational Linguistics and Intelligent Text Processing (pp. 563-566). Springer Berlin Heidelberg.