TOWARD MULTI-LABEL CLASSIFICATION USING AN ONTOLOGY FOR WEB PAGE CLASSIFICATION

Yaya Traoré[1] and Sadouanouan Malo[2] and Bassolé Didier[1] and Séré Abdoulaye[2]

[1]University Joseph KI-ZERBO, Ouagadougou, BURKINA FASO
[2]University Nazi Boni, Bobo-Dioulasso, BURKINA FASO

ABSTRACT

Automatic categorization of web pages has become more significant to help the search engines to provide users with relevant and quick retrieval results. In this paper, we propose a method based on Multi-label Classification (ML) using an ontology which allows the prediction of the categories of a new web page created and tagged. It uses the ontology in the learning phase as well as in the prediction phase. In the learning phase, the ontology is used to build the training set. In the prediction phase, the ontology is used to place the new pages tagged in the most specific categories. The experiment evaluation demonstrates that our proposal shows the substantial results.

KEYWORDS

Multi-label classification (ML), ontology, categorization, prediction.

1. INTRODUCTION

Nowadays, many web platforms are used to allow collaboration between users of a community for creating and sharing knowledge. The web pages are semantically annotated. The number of web pages are continuously growing and can cover almost any information needed. However, the huge amount of web pages and the organization of these pages make the retrieval of precise and exact information more and more difficult for a user. So an efficient and accurate method for classifying this huge amount of data is very essential if the web pages are to be exploited to its full potential. There doesn't exist any specific method to automate this task. We deal with this problem as a Multi-label (ML) classification problem [1], [12] consisting in predicting the categories of a new page according to its tags. In our context, categories are looked upon as text labels.

In order to use the label relationships to build the training data, we associate ML method with ontology. An ontology [2] is used to present the domain knowledge. In this paper, we propose a novel method that uses a method of ML based on ontology to predict the categories of a new web page. Experiments are implemented to evaluate the performance of the proposed approach on the datasets of the uniprot[1] web site. The experimental results indicate that the approach has a better performance.

Natarajan Meghanathan et al. (Eds) : ICCSEA, WiMoA, SCAI, SPPR, InWeS, NECO - 2019
pp. 183-191, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91815

The remainder of the paper is organized as follows : Section 2 presents Multi-label classification problem for web page. Section 3 presents an external Ontology used to annotate the page of semantic web platform. Section 4 presents related work. In Section 5, we describe and give details on the proposed approach Multi-label classification using an ontology for web page classification. Section 6 specifies primary experiment results. Finally, Section 6 ends with a conclusion and perspectives.

2. MULTI-LABEL CLASSIFICATION (ML)

Traditional classification tasks deal with assigning instances to a single label. In Multi-label classification (ML), the task is to find the set of labels that an instance can belong to rather than assigning a single label to a given instance. In this case, each instance may belong to many classes simultaneously and when an instance is labeled with a certain class. A multi-label classification for web page in Semantic web platform deals with a situation where web page can belong to more than one category. Semantic web platform is basically defined by a set of categories and pages. Each page is assigned to one or more categories and includes a set of tags. Formally, semantic web platform web is defined by :

- P is the finite set of web pages, let n the number of pages,

- Let T the finite set of tags and $R_T \subseteq P{\times}T$ a binary relation between P and T, let m the number of tags. We denote : $\forall\ i \in [0,n], \forall\ j \in [0,m],\ \forall\ P_i \in P\ , \forall\ T_j \in T, R_T(\ Pi,\ Tj) = t_{ij}$ with $t_{ij}=1$ if the page $P_i \in P$ is tagged by $T_j \in T$ and 0 otherwise,

- C the finite set of categories and $RC \subseteq P{\times}C$ a binary relation between P and C, let k the number of categories. We denote : $\forall\ i \in [0,n], \forall\ j \in [0,k],\ \forall\ P_i \in P\ , \forall\ C_j \in C,\ R_C(\ P_i,\ C_j) = c_{ij}$ with $c_{ij}=1$ if the page $P_i \in P$ is categorized by $C_j \in C$ and 0 otherwise. We define the function g which allows to obtain all the pages associated with a category as follows:

- g: $C \rightarrow P$ such that, $\forall c \in C,\ g\ (c) = \{p\ /\ p \in P\ and\ RC\ (p,c) = 1\}$.

The Table 1. shows the training data for multi-label classification for web page classification.

Table 1. Training data of multi-label classification for web page classification

	T_1	....	T_m	C_1	...	C_k
P_1	t_{11}	....	t_{1m}	c_{11}	...	c_{1k}
....	....	....	....	....	....	....
P_n	t_{n1}	....	t_{nm}	c_{n1}	....	c_{nk}
P_{n+1}	$t_{(n+1)1}$	....	$t_{(n+1)m}$	?	?	?

There are multiple approaches to deal the multi-label classification problem. The first approach is Binary Relevance (BR) method. In this approach we can use k independent binary classifiers corresponding to k classes (categories) in our data. Let h the classifier of ML. BR decomposes the learning of h into a set of binary classification tasks, one per label, where each single model h_l is learned independently, using only the information of that particular label l and ignoring the information of all other labels. Hence, $h_l(x)$ =1, if the label l is predicted for the instance x. $h(x)$ is the set of relevant labels predicted by h for the object x. Thus, for an new instance x, BR outputs the union of the labels predicted by the k classifiers. Table 2. show BR method of ML problem with training data for n=5, m=3 and k=3.

Table 2. Transformed data sets produced by Binary Relevance (BR) method.

	T_1	T_2	T_3	C_1
P_1	t_{11}	t_{12}	t_{13}	c_{11}
P_2	t_{21}	t_{22}	t_{23}	c_{21}
P_3	t_{31}	t_{32}	t_{33}	c_{31}
P_4	t_{41}	t_{42}	t_{43}	c_{41}
P_5	t_{51}	t_{52}	t_{53}	c_{51}

	T_1	T_2	T_3	C_2
P_1	t_{11}	t_{12}	t_{13}	c_{12}
P_2	t_{21}	t_{22}	t_{23}	c_{22}
P_3	t_{31}	t_{32}	t_{33}	c_{32}
P_4	t_{41}	t_{42}	t_{43}	c_{42}
P_5	t_{51}	t_{52}	t_{53}	c_{52}

	T_1	T_2	T_3	C_3
P_1	t_{11}	t_{12}	t_{13}	c_{13}
P_2	t_{21}	t_{22}	t_{23}	c_{23}
P_3	t_{31}	t_{32}	t_{33}	c_{33}
P_4	t_{41}	t_{42}	t_{43}	c_{43}
P_5	t_{51}	t_{52}	t_{53}	c_{53}

3. ONTOLOGY

Ontology [3], [2] represents the relevant concepts (classes) of a domain. Each concept is defined by a set of consensual terms that is not specific to an individual but accepted by a community of users. Specifically, all the defining terms are organized in hierarchy. In semantic web platform, the classes of ontology are used by the experts to annotate the new page created by the users. Uniprot.org web site is an example of semantic web platform where web pages are annotated by keywords (tags) and classes (categories) of gene ontology (GO).

Let Co the set of classes of an external ontology and HC the hierarchy of classes of the ontology. $\forall\ c \in Co$, we denote:

- $ch(c)$: the set of children classes of c in HC,
- $desc(c)$: the set of descendant classes of c in HC,
- $sib(c)$: the set of sibling classes of c in HC.

4. RELATED WORK

Many similar studies have been realized about web page categorization. The studies differed with the methods used and the use of different machine learning algorithms. [6] proposed a web page classifier that is based on an adoption of k-Nearest Neighbor (k-NN) approach. To improve performance of k-NN approach, authors supplemented the k-NN approach with a feature selection method and a term weighting scheme using markup tags. [7] proposed a method to classify web pages using the Naïve Bayesian algorithm. The research considered ten categories to be classified and the NB algorithm had an accuracy of 89.05%. It is also observed that the classification accuracy of the classifier is proportional to number of training documents. The results are quite encouraging. [8] proposed a system to classify web pages using Neural Networks. [10] used a combined approach of Page Rank and Feature Selection. [11] proposed a method that made use of other information in a web page such as images, audio and video. In this paper, we propose a novel method that use Binary relevance (BR) method of Multi-label Classification (ML) and an ontology to categorize a new web page in a semantic web platform used to share knowledge from different communities.

The mult-label classification problem has been studied in the work of [1], [4], [5], [12], [13], [14], [15]. The diverse applications of multi-label classification are studied in several domains, such as text categorization. In our context, we apply the ML to predict the categories of a new web page. In the literature, Multi-label classification approaches can be divided into transformation and adaptation methods. A comprehensive review on multi-label classification algorithms is given in [13]. We focus on transformation method in this paper. Transformation methods decompose the multilabel problem into a set of binary classification problems. The most popular method is called Binary (BR), which trains a binary classifier for each class (against the others), inherently assuming independence between the classes. In this paper, we apply a Binary relevance method to categorize web page.

BR is a naturally multi-label classification approach. While BR has been used in many practical applications, it has been widely criticized for its implicit assumption of label independence which might not hold in the data. BR + algorithm [16], an extension of the BR algorithm, considers the relationship between labels, and constructs binary classification problems, similarly to BR. The differences are its descriptor attributes, which merge all original attributes as well as all labels, except the own label to be predicted. Classifier Chains [5] (CC) arrange the local classifiers in a chain where the outcome of a classifier is used as a feature on the next classifiers in the chain, allowing some dependency between labels to be modeled. In this paper, we propose to reduce the number of instance use in the training set of each label and to use the relationship between labels. An external ontology is used to define the set of positive and negative instances (pages) for each category (existing in an ontology) in the learning phase. Our contribution is described in the section 5.

5. PROPOSED METHODOLOGY

In this section, we give details on the methodology used to predict the categories of a new page. The proposed methodology contains two steps. Its main processes are described as follows. The first step is the learning phase. In this step, training data is partitioned into $|cl|$ subsets, where $|cl|$ means the total number of categories. Then, $|cl|$ binary classifiers are built. The second step is the prediction phase. For a new page created and tagged, this step uses for each category cl, his binary classifier to predict if cl is affected.

5.1. Step 1: Learning Phase

The first phase is selecting sibling classes, descendant classes for each category by using the hierarchy of ontology and built the training set of cl. For each category cl, we use an ontology to select these sibling ($sib(cl)$) and descendant ($desc(cl)$) classes. For the training set of the category cl, instances that belong to cl or the descendants ($desc(cl)$) $of\ cl$ are chosen as the positive instances, and other belonging to the siblings of cl are selected as negative instances. In the case that the class cl has no correspondent class in an ontolgoy, positive instances of cl are calculated by $g(cl)$ and negative instances are the instances which are not selected by $g(cl)$. The set of positive $Tr^{+}(cl)$ and negative $Tr^{-}(cl)$ instances are given as follows :

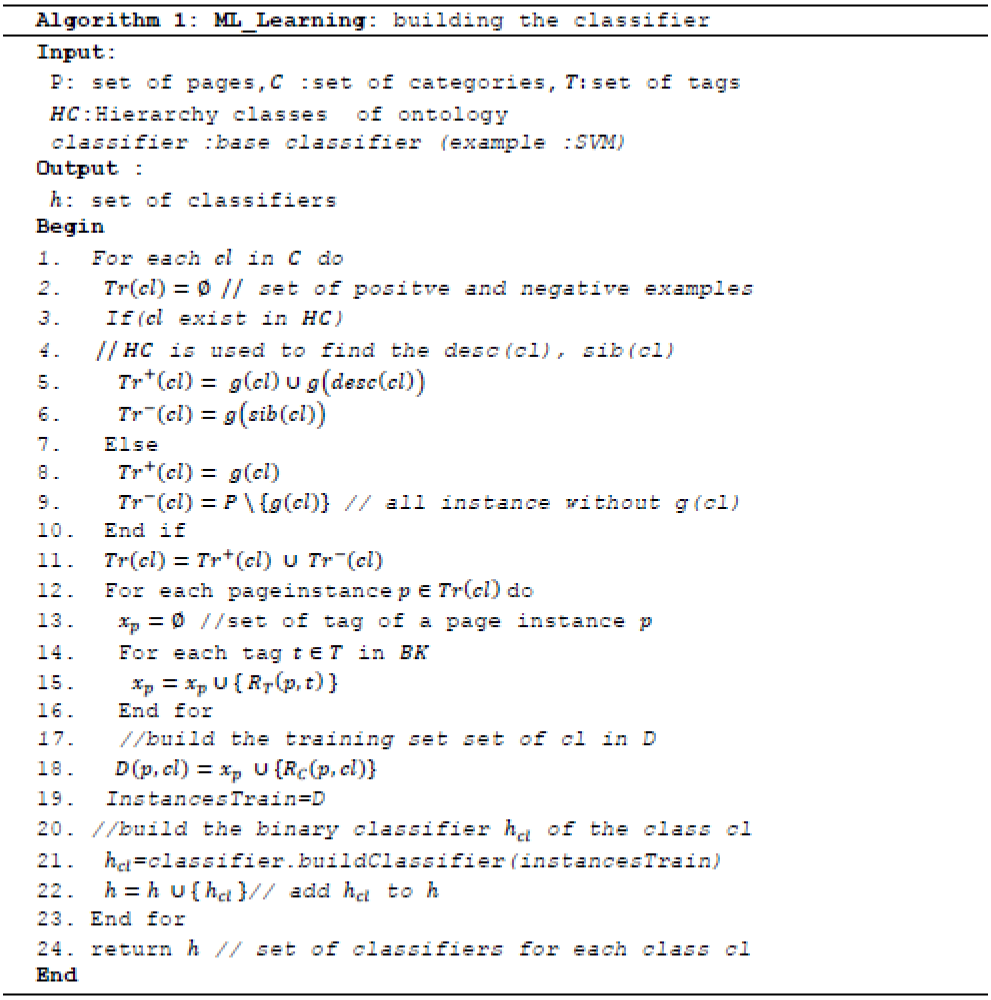
If $cl\ exist\ in\ HC\ then$:

- $Tr{+}(cl) = g(cl) \cup g(desc(cl))$
- $Tr{-}(cl) = g(sib(cl))$

Else :

- $Tr+(cl)= g(cl)$
- $Tr-(cl)=P \setminus \{g(cl)\}$

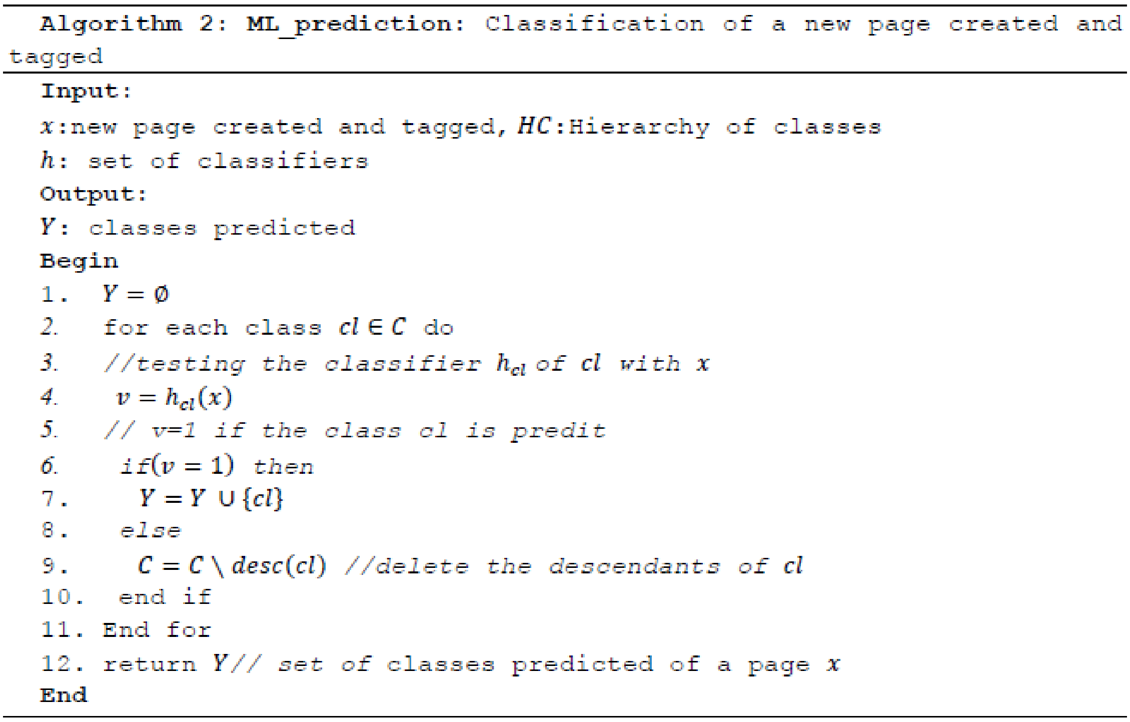
The last phase of this step, is the choice of the base classifier. For each category cl a base classifier [19] (for example : SVM, NaiveBayes, J48,…) is trained by using the training data set at this category. We propose the **Algorithm 1** (***ML_learning)***, to build the set of classifiers of categories. This algorithm takes as input, P,C,T, HC the hierarchy of classes of ontology and the base classifier. Ontology is used to select the descendant ($desc(cl)$), sibling ($sib(cl)$) categories of cl. The Algorithm generates all the set of positive and negative examples for the class cl (line 11). For each page examples $p \in Tr(cl)$, it generates the rows (line 15) of the page instance $p \in Tr(cl)$ of the training dataset of cl. it generates the training dataset (line 18) of the class cl. Next, it builds the binary classifier (line 21) of a class cl. h is used to save all the binary classifiers (line 22), of all classes. Finally, the algorithm returns the set of classifiers (line 24).



```
Algorithm 1: ML_Learning: building the classifier
Input:
 P: set of pages,C :set of categories, T:set of tags
 HC:Hierarchy classes  of ontology
 classifier :base classifier (example :SVM)
Output :
 h: set of classifiers
Begin
1.  For each cl in C do
2.   $Tr(cl) = \emptyset$ // set of positve and negative examples
3.   If(cl exist in HC)
4.  //HC is used to find the desc(cl), sib(cl)
5.    $Tr^{+}(cl) = g(cl) \cup g\big(desc(cl)\big)$
6.    $Tr^{-}(cl) = g\big(sib(cl)\big)$
7.   Else
8.    $Tr^{+}(cl) = g(cl)$
9.    $Tr^{-}(cl) = P \setminus \{g(cl)\}$ // all instance without g(cl)
10.  End if
11.  $Tr(cl) = Tr^{+}(cl) \cup Tr^{-}(cl)$
12.  For each pageinstance $p \in Tr(cl)$ do
13.   $x_p = \emptyset$ //set of tag of a page instance p
14.   For each tag $t \in T$ in BK
15.    $x_p = x_p \cup \{ R_T(p,t) \}$
16.   End for
17.   //build the training set set of cl in D
18.   $D(p,cl) = x_p \cup \{R_C(p,cl)\}$
19.  InstancesTrain=D
20. //build the binary classifier $h_{cl}$ of the class cl
21.  $h_{cl}$=classifier.buildClassifier(instancesTrain)
22.  $h = h \cup \{ h_{cl} \}$// add $h_{cl}$ to h
23. End for
24. return h // set of classifiers for each class cl
End
```

5.2. Step 2: Prediction Phase

This step uses the built classifiers with the algorithm 1 (***ML_learning())*** to predict the categories of a new page tagged. The algorithm 2, ***ML_prediction()*** is used to perform the prediction. This algorithm takes as input the new page created and tagged, the set of classifiers h, HC the hierarchy of classes. The algorithm traverses the set of categories and for each class, the corresponding classifier hcl is invoked (line 4). For improving the prediction phase, if a class is not predicted, then these descendants (line 10) are pruned in the list of classes. Finally, the algorithm returns the set of classes predicted (line 13) for a new page x.



```
Algorithm 2: ML_prediction: Classification of a new page created and tagged
Input:
$x$:new page created and tagged, $HC$:Hierarchy of classes
$h$: set of classifiers
Output:
$Y$: classes predicted
Begin
1.  $Y = \emptyset$
2.  for each class $cl \in C$ do
3.  //testing the classifier $h_{cl}$ of $cl$ with $x$
4.   $v = h_{cl}(x)$
5.  // v=1 if the class cl is predit
6.   if$(v = 1)$ then
7.     $Y = Y \cup \{cl\}$
8.   else
9.     $C = C \setminus desc(cl)$ //delete the descendants of $cl$
10.  end if
11. End for
12. return $Y$// set of classes predicted of a page $x$
End
```

6. EXPERIMENTS

6.1. Dataset And Experimental Setup

Platform : The experiments are implemented under macOS 10.13.4 (17E199), with Intel Core i5 @ 2,5 GHz and 8 Go RAM. The code is implemented in Java and used the weka library.

Dataset : As an application area we have chosen bioinformatics[2] in view of the fact that it is an important, that has many pages annotated by some keywords and classes of GO (Gene Ontology). First we extract the pageID, Keywords and Gene ontology IDs to create the pages of web platform. Fig.2. illustrates the whole process of collecting training data initial (a) and a fraction of the GO taxonomy (b) called hierarchy of categories HC.

The characteristics of the experimental dataset, such as pages number, categories number and tags number of each data set, are summarized:

- Training pages number : 5199
- Testing pages number : 3985

- Categories number : 630
- Tag number : 2815

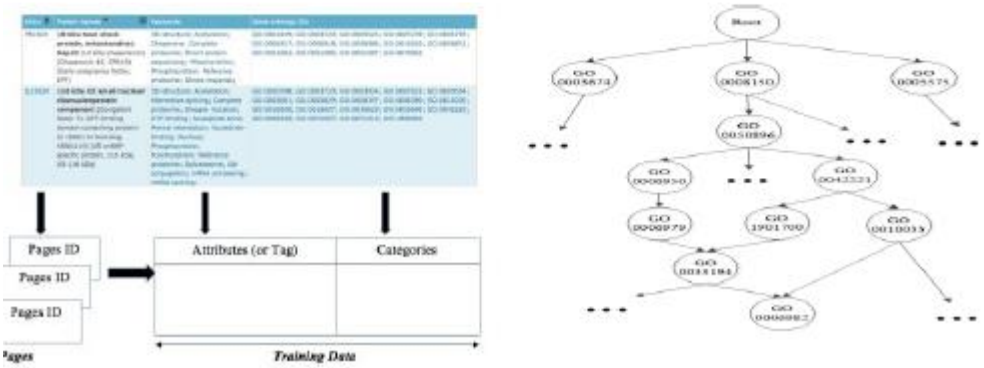


Fig. 2. (a) the whole process of collecting training data (b) a fraction of the Gene Ontology

6.2. Evaluation Metrics

The metrics precision (P), recall (R) and Fmeasure (F1) are proposed to evaluate our method. The precision, recall and F1 for the i –th example are defined as :

$$P_i = \frac{|Y_i \cap Z_i|}{|Y_i|}, \quad R_i = \frac{|Y_i \cap Z_i|}{|Z_i|}, \quad F1_i = \frac{2 * P_i * R_i}{P_i + R_i}$$

where, for an example i, Y_i is the set containing all of the predicted classes, and Z_i the set including all of its true classes. There are two methods to combine the performance of all instances to evaluate the results measured on a dataset with n instances labeled: the micro-averaging version and the macro-averaging version. We use the macro-averaging version, the precision P, recall R and $F1$ are first computed for each instance and then averaged.

$$P = \frac{\sum_{i=1}^{n} P_i}{n}, \quad R = \frac{\sum_{i=1}^{n} R_i}{n}, \quad F1 = \frac{\sum_{i=1}^{n} F1_i}{n}$$

6.3. Experiments Results And Analysis

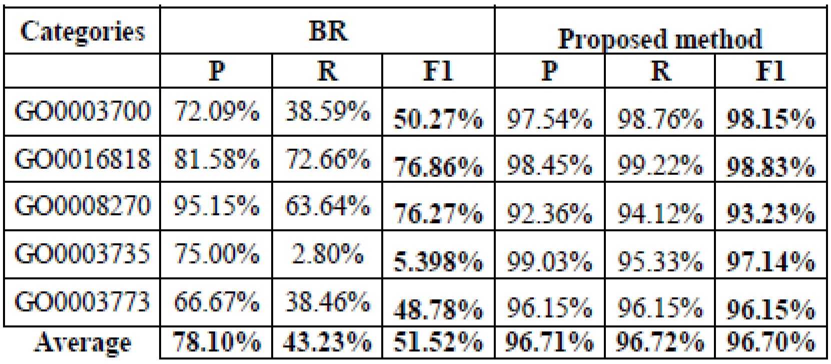
To observe the performance of the method proposed to place the pages in the good categories, we use some base classifiers: SVM, NaiveBayes, J48. The results (Table 3.) show that the method gives the good performance and proposed method +SVM gives the better performance.

Table 3. The experimental results of proposed method with some base classifiers

	P	***R***	***F1***
Proposed method + SVM	76.85%	77.67%	76.96%
Proposed method + NaiveBayes	61.25%	69.94%	65.08%
Proposed method + J48	74.81%	73.76%	74.25%

The performance of the Proposed method compares with BR are shown in Table 4. Table 4 shows the performance of the Proposed method and BR method from few categories. Proposed method has the best performance in Precision, Recall and Fmeasure than BR. The macro-averaging $F1$ for Proposed method (76.96%) is better than macro-averaging $F1$ of BR(26.28%). This best performance is due firstly to the use of the ontology. Thus, we can come to a conclusion that our method proposed improve the performance of the method BR of Multi-label classification.

Table 4. BR method vs Proposed method for five categories.



Categories	BR			Proposed method		
	P	R	F1	P	R	F1
GO0003700	72.09%	38.59%	**50.27%**	97.54%	98.76%	**98.15%**
GO0016818	81.58%	72.66%	**76.86%**	98.45%	99.22%	**98.83%**
GO0008270	95.15%	63.64%	**76.27%**	92.36%	94.12%	**93.23%**
GO0003735	75.00%	2.80%	**5.398%**	99.03%	95.33%	**97.14%**
GO0003773	66.67%	38.46%	**48.78%**	96.15%	96.15%	**96.15%**
Average	**78.10%**	**43.23%**	**51.52%**	**96.71%**	**96.72%**	**96.70%**

7. CONCLUSIONS

In this paper, we propose a novel method based on Multi-label Classification (ML) using an ontology for web page classification. In the learning phase, the ontology is used to select the set of positive and negative intances for learning and building the training set. In the prediction phase, ontology is used to improve the execution time of the method of Mutli-label Classification. Experiment results on a data downloaded from the biological database Uniprot (www.uniprot.org) show that proposed method improve BR method of ML. Due to its good performance, the method proposed is expected to be a potential approach to solve the automatic web page classification in a semantic web platform.

In the future work, we will test our approach in more methods of Multi-label classification and more datasets. Furthermore, we can optimize our method by changing the base classifier, or trying to use different classifiers for different nodes.

REFERENCES

1. Zhang, M., Zhou, Z.: A review on multi-label learning algorithms. IEEE Trans. Knowl.Data Eng. 26(8), 1819{1837 (2014). DOI10.1109/TKDE.2013.39. URL http://dx.doi.org/10.1109/TKDE.2013.39

2. Gomez Perez A. Ontological engineering : A state of the art. Expert update 2. Technique et Science Informatiques 28, page 81233–126,1999.

3. T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. in : International Journal of Human-Computer Studies - Special issue : the role of formal ontology in the information technology, V, pages 907 – 928, 1993.

4. Madjarov G., Kocev D., Gjorgjevikj D., and Dzeroski S. An extensive experimental comparison of methods for multi-label learning. Pattern recognition, 45(9) :3084–3104., 2012.

5. W. Bi and J. T. Kwok. Multi-label classification on tree-and DAG-structured hierarchies in Proc. 28th Int. Conf. Mach. Learn., Bellevue, WA, USA, 2011, pp. 17–24.

6. O. W. Kwon and J. H. Lee, Web page classification based on knearest neighbor approach, IRAL, 2003.

7. O. W. Kwon and J. H. Lee, Web page classification based on knearest neighbor approach, Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages, 2000, Hong Kong, China, September 30 - October 01, 2000

8. Ajay S. Patil, B.V. Pawar. Automated Classification of Web Sites using Naive Bayesian Algorithm, Proceedings of the International Multi Conference of Engineers and Computer Scientists 2012 IMECS 2012, 14-16 March, 2012, Hong Kong

9. Q. S. Mahdy and K. Qader. Web page classification by using neural networks. Journal of Pure Applied Sciences, 2011.

10. S. Shibu, A. Vishwakarma, and N. Bhargava, A combination approach for web page classification using page rank and feature selection technique, International Journal of Computer Theory and Engineering, 2010.

11. A. P. Asirvatham and K. K. Ravi, Web Page Categorization Based on Document Structure, 2002.

12. F. HERRERA, F. CHARTE, A.J. RIVERA, M.J. DEL JESUS, Multi-label Classification Problem Analysis. Metrics and Techniques. Springer International Publishing, ISBN 78-3-319-41111-8, 2016.

13. Tsoumakas G. and Katakis I. Multi-label classification: An overview. International Journal of Data Warehousing and Mining (IJDWM), 3(3) :1–13, 2007.

14. Tsoumakas G., Katakis I., and Vlahavas I. Mining multi-label data. In Data mining and knowledge discovery handbook, pages 667–685. Springer, 2010.

15. Tsoumakas G. and Vlahavas I. Random k-labelsets : An ensemble method for multilabel classification. In Proceedings of the 18th European conference on Machine Learning, 2007.

16. Cherman EA, Metz J, Monard MC. Métodos multirrótulo independentes de algoritmo: um estudo de caso. In: Anais da XXXVI Conferencia Latinoamericana de Informática (CLEI). Asuncion, Paraguay; 2010. p. 1–14.

17. Clare A, King RD. Knowledge discovery in multi-label phenotype data. Lect Notes Comp Sci 2001:42–53.

18. Suzuki E, Gotoh M, Choki Y. Bloomy decision tree for multi-objective classification. Princ Data Min Knowl Discov 2001:436–47. http://dx.doi.org/ 10.1007/3-540-44794-6_36.

19. Sun, A.; Lim, E.P.; Liu, Y. On strategies for imbalanced text classification using SVM: A comparative study. Decis. Support Syst. 2009, 48, 191–201.