Combining ILP with Semi-supervised Learning for Web Page Categorization

Nuanwan Soonthornphisaj and Boonserm Kijsirikul

Abstract**—**This paper presents a semi-supervised learning algorithm called Iterative-Cross Training (ICT) to solve the Web pages classification problems. We apply Inductive logic programming (ILP) as a strong learner in ICT. The objective of this research is to evaluate the potential of the strong learner in order to boost the performance of the weak learner of ICT. We compare the result with the supervised Naive Bayes, which is the well-known algorithm for the text classification problem. The performance of our learning algorithm is also compare with other semi-supervised learning algorithms which are Co-Training and EM. The experimental results show that ICT algorithm outperforms those algorithms and the performance of the weak learner can be enhanced by ILP system.

Keywords**—**Inductive Logic Programming, Semi-supervised Learning, Web Page Categorization.

I. Introduction

THE Web page categorization task is a challenging problem since the growth rate of the Web page is very high. Nevertheless, there is no common style of these Web pages in each category. Therefore we need human experts to categorize these Web pages into categories which is a time consuming and expensive task.

Many researches have been done to find the efficient algorithm which can classify these documents automatically. These algorithms use machine learning concept such as neural networks, naive Bayes and *k*-nearest neighbors [1], [2], [3] in order to find the general model of each Web's category based on a set of labeled documents. Unfortunately, the efficient algorithms still need a high portion of labeled documents to construct the model.

In order to relax the problem of the big amount of these labeled documents, the semi-supervised learning algorithms are developed. These algorithms need a small amount of labeled data, after that the algorithm will try to label the unlabeled documents and accumulate the newly labeled data during the learning process.

This paper aims to combine the Inductive Logic Programming (ILP) with the iterative cross-training algorithm (ICT). ICT is the semi-supervised learning algorithm which consists of two learners, the strong and the weak learner.

The strategy of ICT is to find the strong learner that can induce the ability of the weak learner and use the weak learner with the knowledge supplied from the strong learner to do its job in the real world application.

Our assumption is that the ILP is the strong learner since it is normally supplied with the domain knowledge during the learning process, hence it has high abilty in making the decision of the Web's category. The reason that supports this idea comes from the experiments which were done on the Thai-non Thai Web page classification [4]. In that problem domain, ICT use a word-segmentation with the domain knowledge in the form of dictionary as a strong learner. We found that the word-segmentation classifier could enhance the performance of the simple naïve Bayes classifier of ICT.

This paper is organized as follows, section 2 of this paper gives the concept and the detail of ICT. Other algorithm used in comparison with ICT will be given in section 3. The experimental results and the conclusion part will be described in section 4 and 5.

II. Iterative Cross-training

A. The Architure of ICT

In this section, we present the framework of the ICT, which consists of two learners. Each learner gets a small amount of labeled data. The strong learner (*classifier1*) starts the learning process from the labeled data and classify unlabeled data (*TrainingData2*). The weak learner (*classifier2*) uses these newly labeled data to learn and classify *TrainingData1*. The detail of ICT algoritm will be given in Table I.

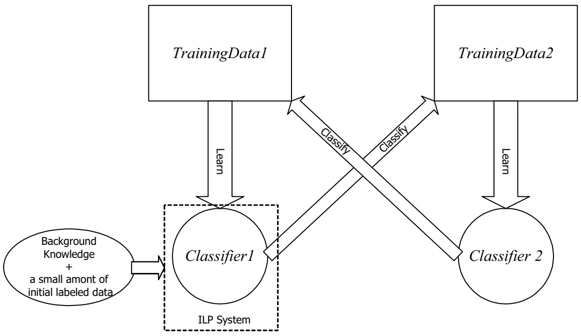

Figure. 1. The Architure of Iterative Cross-Training

This work was supported by the Thailand Research Fund under Grant no. MRG4680156

Nuanwan Soonthornphisaj is with the Department of Computer Science, Faculty of Science, Kasetsart University, Bangkok, Thailand; e-mail fscinws@ ku.ac.th

Boonserm Kijsirikul is with the Computer Engineering Department, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand; e-mail boonserm.k@ chula.ac.th

TABLE I
THE LEARNING ALGORITHM OF ICT

Given:
- Two training sets *TrainingData1* for the strong learner and *TrainingData2* for the weak learner (*TrainingData1* and *TrainingData2* both contain *U* labeled examples).
- – Use labeled data in *TrainingData1* to estimate the parameter set θ_s of the strong learner.
- – Use labeled data in *TrainingData2* to estimate the parameter set θ_w of the weak learner.
- – Loop until all data are labeled.
- Use the strong classifier with current θ_s to classify *TrainingData2* into categories.
  - Train the weak learner by the labeled examples in *TrainingData2* to estimate the parameter set θ_w of the classifier.
- Use the weak classifier with current θ_w to classify *TrainingData1* into categories.
  - Train the strong classifier by the labeled examples in *TrainingData1* to estimate the parameter set θ_s of the classifier.

B. The Strong Learner

The ILP system is embeded in the strong learner of ICT. Many ILP systems have been developed such as GOLEM [5], FOIL [6], PROGOL [7]. We choose the PROGOL system which uses a technique called inverse entailment to generate the single most specific hypothesis that, together with the background knowledge entails the observe data [8]. PROGOL uses the sequential covering algorithm to learn a set of rules from the hypothesis space. It employs A^* search along the way to find a set of rules that represent the concept of the class. Many researchers point out that PROGOL is seen as a standard ILP learner and is often used as a benchmark when new ILP systems are introduced.

Our ILP system is supplied with two set of examples, i.e., a small amount of initial labeled data and unlabeled data. The ILP system makes use of background knowledge about the categories of the Web pages together with a set of initial labeled data to induce a set of rules (θ_s). Then the system classifies unlabeled examples using the rule set and feeds the newly labeled examples to the weak learner. The weak learner starts its process with the accumulated labeled data to estimated θ_w and classifies *Trainingdata1* into categories. The ILP system continues using its background knowledge and *Trainingdata1* as labeled examples to induce a new rule set. This process is repeated until the system is converged.

The feature sets used by the ILP system are in predicate forms. We extract three feature sets from each Web page as follows.

1) A title predicate, *has_title*(*p,word*), is created using words appearing in the title of the Web page, *p*.
2) A heading predicate, *has_head(p,word)*, is created using words appearing in all headings of the Web page, *p*.
3) A hyperlink predicate, *has_link(p,word)*, is created using words appearing in all hyperlinks of the Web page, *p*.

For the background knowledge which is an important part of the ILP system, we supply the knowledge for each Web page category. This knowledge is also written in predicate form. Table II and III give example lists of background knowledge for the DrugUsage and WebKb dataset.

TABLE II
A SET OF BACKGROUND KNOWLEDGE FOR DRUGUSAGE DATASET

DrugUsage dataset		
class name	**adverse**	
adverse(adverse)	symtom(hematology)	symtom(acute)
adverse(reaction)	symtom(vascular)	symtom(liver)
adverse(interaction)	symtom(cardiovascul)	symtom(metabolism)
symtom(sleepy)	symtom(digestion)	symtom(hepatitis)
symtom(nervous)	symtom(allergy)	symtom(urinary)
class name	**overdose**	
overdose(overdosage)	effect(vomit)	contraindicate (hypoclycemia)
overdose (contraindicate)	contraindicate (contraindicate)	contraindicate(heart)
effect(fatal)	contraindicate (hypersensitivity)	contraindicate (hypertension)
effect(toxic)	contraindicate(peptic)	contraindicate (allergic)
effect(coma)	contraindicate(ulcer)	contraindicate (hypertrophy)
class name	**warning**	
warning(warn)	targetpeople(nurse)	targetpeople(dilivery)
warning(precaution)	targetpeople(pediatric	targetpeople(maternal)
targetpeople(pregnancy)	targetpeople(labor)	targetpeople(animal)
targetpeople(mother)		
class name	**patient information**	
patientinfo(patient)	physician(physician)	usage(room)
information(inform)	physician(doctor)	usage(shake)
information(product)	patientinfo(take)	usage(breath)
information (prescription)	usage(temperature)	usage(instruction)
class name	**clinical pharm**	
phamacology (phamacology)	clinical(clinic)	druganalysis(negative)
phamacology (phamacodynamic)	druganalysis(gram)	dilution(dilution)
phamacology (pharmacokinetic)	druganalysis(positive	dilution(technique)

C. The Weak Learner

For the weak learner, we employ the naive Bayes algorithm which makes its prediction based on the probability obtained from the Bayes theorem. Note that there are 2 assumptions concerning with naïve Bayes. These assumptions are (1) The presence of each word is conditionally independent of all other words in the document given the class label and (2) an assumption that the position of a word is unimportant, e.g. encountering the word “subject” at the beginning of a document is the same as encountering it at the end. The classification result (l^*) obtained from naïve Bayes can be found in Equation 1 and 2.

$$l^* = \underset{l_j}{argmax}\ Pr(l_j) \prod_{i=1}^{n} Pr(w_i \mid l_j, w_1, \ldots, w_{i-1}) \qquad (1)$$

$$= \underset{l_j}{argmax}\ Pr(l_j) \prod_{i=1}^{n} Pr(w_i \mid l_j) \qquad (2)$$

TABLE III
A SET OF BACKGROUND KNOWLEDGE FOR WEBKB DATASET

Class name			
course	**student**	**faculty**	**project**
subject(cs)	sport(soccer)	academic(faculty)	project_def (project)
subject(cse)	sport(hockey)	Academic (institute)	project_def (mission)
subject(ee)	sport(softball)	academic (university)	project_def (objective)
assignment(assign)	sport(golf)	academic (department)	project_def (propose)
assignment(solution)	sport(ski)	teach(course)	project_group (group)
assignment (homework)	hobby(travel)	teach(subject)	project_group (member)
assignment(problem)	hobby(movie)	teach(student)	project_group (researcher)
assignment(question)	hobby(game)	interest(research)	project_group (researcher)
assignment(quiz)	hobby(cook)	interest(paper)	project_group (people)
class(course)	hobby(cat)	interest (publication)	project roup (manager)
class(lecture)	hobby(dog)	interest(subject)	project_group (alumni)
class(lab)	relative(wife)	job(lecture)	place (laboratory)
semester(fall)	relative(friend)	job(teach)	place(lab)
semester(winter)	relative(father)	job(course)	
semester(spring)	relative(son)	activity(member)	
semester(authum)	relative (daughter)	activity(acm)	
material(handout)	relative(family)	activity(ieee)	
material(syllabus)	personal(resume	Activity (commitee)	
material(textbook)	personal(life)	activity (conference)	

The concept of naive Bayes can be found in [4]. Note that the parameter θ_w of the naive Bayes are the probabilities, $Pr(l_j)$ and $Pr(w_i|l_j)$ which are estimated from the training data. The prior probability, $Pr(l_j)$, is estimated from the ratio between the number of examples belonging to class l_j, and the number of all examples. The value of $Pr(w_i|l_j)$ is the conditional probability of seeing word w_i given class label l_j.

III. ALGORITHMS USED IN COMPARISON

A. ICT-NB Algorithm

For ICT-NB algorithm, we implement two classifiers of ICT using naive Bayes. The different between these two classifiers is the feature set. The first classifier is supplied with the words appearing on the heading of the Web page as the feature set, whereas the second classifier uses the words appearing on the content as the feature set. The learning mechanism is the same as the ICT algorithm.

B. Supervised Naive Bayes Algorithm

This algorithm is a single classifier which gets a high portion of labeled documents during the learning process. We test the performance of supervised naive Bayes using two feature sets which are heading and content feature.

C. Co-training Algorithm

The Co-Training algorithm was first introduced by Blum and Mitchell in 1998 [9]. The concept of the algorithm is based on the boosting technique.

That means, the algorithm learns from a small number of initial labeled data, and then it will incrementally classify unlabeled data into categories. The basic assumption of Co-Training is that the instance distribution is compatible with the target function. It requires that, for most examples, the target functions over each feature set predict the same label. For example, in the Web page domain, the class of the instance should be identifiable using either the text appearing on the hyperlink or the text appearing in the page content. The second assumption is that the features in one set of an instance are conditionally independent of the features in the second set, given the class of the instance.

D. Expectation-Maximization Algorithm

Another boosting style algorithm is Expectation-Maximization (EM). This algorithm was first introduced by Dempster et al. [10]. It is an iterative algorithm for maximum likelihood estimation in problems with incomplete data.

Given a model of data generation, and data with some missing values, EM iteratively uses the current model to estimate the missing values, and then uses the missing value estimates to improve the model. Using all of the available data, EM will locally maximize the likelihood of the parameters and give estimates for the missing values. Therefore, the class labels of the unlabeled data are treated as the missing values. EM has two steps, which are the E-step and M-step, respectively. The E-step calculates probabilistically weighted class labels for every document using the classifier. For the M-step, it estimates new classifier parameters using all documents. In Nigam, et al.'s work [11], they combined EM with a naive Bayes classifier to solve the text classification problem. The algorithm has shown to be able to significantly increase text classification accuracy when given limited amounts of labeled data and large amounts of unlabeled data.

IV. EXPERIMENTAL RESULT

The Web page datasets, we use for our experiments are the WebKb dataset [13] and the Drug-Usage dataset [12]. The performance evaluation is done using the standard precision (P), recall (R) and F_1-measure (F_1). These measurements are defined as follows.

$$P = \frac{\text{no. of correctly predicted examples in the target class}}{\text{no. of predicted examples in the target class}} \tag{3}$$

$$R = \frac{\text{no. of correctly predicted examples in the target class}}{\text{no. of all examples in the target class}} \tag{4}$$

$$F_1 = \frac{2PR}{P+R} \tag{5}$$

The experimental set up for each dataset is as follows.

A. The WebKb Dataset

For ICT, Co-Training and EM, we randomly selected 30% of all examples from each category to be initial labeled data. The unlabeled training data consisted of 30% of all examples and 40% of all examples were used as a test set. The experiments were conducted using 5-fold cross-validation. Table IV shows the results of all experiments conducted on the WebKb dataset. In the table, ICT-ILP stands for the performance of ICT which combines the ILP system in one of the classifiers. ICT-NB is ICT which combines two naive Bayes classifers, each of which learns from different feature sets. Co-Training stands for the Co-Training algorithm, S-Bayes stands for the supervised naive Bayes algorithm. Note that the *Classifier1* of ICT-ILP is the *Progol* system. For other algorithms, *Classifier1* means the heading-based classifier. The *Classifier2* is the content-based classifier for all of the algorithms.

Considering the two versions of ICT (ICT-ILP and ICT-NB) in Table IV, we found that ICT-ILP's performance measured by F_1 was increased from 78.25% to 80.90% on *Classifier1*. Moreover, *Classifier1* was able to boost the performance of *Classifier2*. The *Classifier2*'s performance was enhanced from 71.76% to 84.44%. Compared to the supervised naive Bayes algorithm, ICT-ILP outperformed S-Bayes on *Classifier1*. The reason that ICT-ILP got the highest performance came from the contribution of the strong learner (the Progol system).

TABLE IV
THE PERFORMANCE OF CLASSIFIERS ON THE WEBKB DATASET

	Classifier1			Classifier2		
Algorithm	P	R	F_1	P	R	F_1
ICT-ILP	80.00	81.82	80.90	82.61	86.36	84.44
ICT-NB	71.85	94.39	78.25	67.23	86.73	71.76
Co-Training	73.95	84.69	75.64	79.69	60.72	66.14
S-Bayes	74.99	87.24	79.91	76.95	84.18	79.60
EM	68.62	91.64	75.98	76.78	74.28	70.70

B. DrugUsage Dataset

For ICT, Co-Training and EM, we selected 33% of all examples to be initial labeled data. The training set consisted of 33% and the remaining 34% was a test set. For the supervised naive Bayes classifier, we selected 66% of all examples to be labeled data. The test set consisted of 34% of all examples. All experiments were conducted using 3-fold cross validation.

For the performance of *Classifier1* (as shown in Table V), ICT-ILP got the highest F_1. ICT-NB's performance was increased from 69.17% to 89.90%. This means that the ILP system had contributed 30% of performance enhancement to ICT-NB. For *Classifier2*, ICT-ILP got 65.39% measured by F_1, which was higher than that of ICT-NB. The overall learning process of ICT-ILP took 1 hour to converge.

The learning process of ICT-ILP took more time than the ICT-NB, since the strong learner, Progol, needed time to do a general-to-specific search to get the optimum set of rules. In each iteration of ICT, the Progol took about 1 hour to generate the rules, therefore it took 3 hours for ICT-ILP to converge.

TABLE V
THE PERFORMANCE OF CLASSIFIERS ON THE DRUGUSAGE DATASET.

	Classifier1			Classifier2		
Algorithm	P	R	F_1	P	R	F_1
ICT-ILP	82.37	98.32	89.9	56.03	88.2	65.39
ICT-NB	60.54	80.66	69.17	57.14	70.3	63.04
Co-Training	55.51	62.52	58.81	50.45	77.56	61.14
S-Bayes	75.74	92.67	83.35	68.81	87.12	76.89
EM	72.41	87.5	79.25	33.33	95.83	49.46

V. CONCLUSION

We have presented the enhancement version of ICT using the ILP system. We found that the induced rules have more efficiency in classify unlabeled examples. This evidence can be seen from all experimental results. The benefit of the ILP system can be seen clearly when all categories in the dataset are closely related. The reason is that most of the words in the closely related categories are likely to be equally distributed. Thus using the statistical approach like the naive Bayes learner might not perform well enough to distinguish the difference between categories. The representation of the rule sets, on the other hand, can point out the specific location in each Web page that can be used as a standard prototype of the categories.

REFERENCES

[1] R. A. Calvo and H. A. Ceccatto, "Intelligent document classification," *Intelligent Data Analysis*, vol. 4, no.5, 2000.

[2] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys* (CSUR), vol.34, no. 1, pp. 1- 47, 2002.

[3] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proc. 22nd Annu. Int. SIGIR*, Berkley, 1999, pp. 42-49.

[4] B. Kijsirikul, P. Sasipongpairoege, N. Soonthornphisaj and S. Meknavin, "Supervised and unsupervised learning algorithms for Thai Web page identification," In *Proc. Pacific Rim Int. Conf. on Artificial Intelligence*, Australia, 2000, pp. 690-700.

[5] S. Muggleton, S. and C. Feng, "Efficient induction of logic programs," In *Proc. 1st Conf. Algorithmic Learning Theory*. 1990.

[6] J.R. Quinlan, "Learning logical definitions from relations," *Machine Learning,* vol. 5, no. 3, pp.239-266, 1990.

[7] S. Muggleton, "Inverse entailment and progol," *New Generation Computing,* vol. 13, pp. 245-286, 1995.

[8] T.M.Mitchell, *Machine Learning*. New York: McGraw-Hill, 1997, pp. 180-184.

[9] A. Blum and T.M. Mitchell, "Combining labeled and unlabeled data with co-training," In *Proc. 11th Annu. Conf. Computational Learning Theory,* 1998.

[10] A.P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society* Series B vol. 39, pp. 1-38, 1977.

[11] K. Nigam, A. McCallum, S. Thrun and T.M. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine Learning,* vol. 9 no. 2, pp.103-134, 1999.

[12] DrugUsage. 2001. Data set. http://www.kindcu.siit.ac.th, Phatumtani, Thailand.

[13] WebKb. 2000. Data set. http://www.cs.cmu.edu/afs/cs.cmu.edu Carnegie Mellon University, U.S.A.