Journal of Korea Multimedia Society Vol. 13, No. 12, December 2010(pp. 1786-1797)

Machine Learning Based Automatic Categorization Model for Text Lines in Invoice Documents

Hyunkyung Shin†

ABSTRACT

Automatic understanding of contents in document image is a very hard problem due to involvement with mathematically challenging problems originated mainly from the over-determined system induced by document segmentation process. In both academic and industrial areas, there have been incessant and various efforts to improve core parts of content retrieval technologies by the means of separating out segmentation related issues using semi-structured document, e.g., invoice,. In this paper we proposed classification models for text lines on invoice document in which text lines were clustered into the five categories in accordance with their contents: purchase order header, invoice header, summary header, surcharge header, purchase items. Our investigation was concentrated on the performance of machine learning based models in aspect of linear-discriminant-analysis (LDA) and non-LDA (logic based). In the group of LDA, naïve baysian, k-nearest neighbor, and SVM were used, in the group of non LDA, decision tree, random forest, and boost were used. We described the details of feature vector construction and the selection processes of the model and the parameter including training and validation. We also presented the experimental results of comparison on training/classification error levels for the models employed.

Key words: Text classification, document image analysis, document image understanding, information retrieval, machine learning, CART (classification and regression tree), automatic invoice document processing.

1. INTRODUCTION

Classification of queries, texts, and documents is one of the major subjects in information science area [1]. The common purpose of these subjects is to understand the contents contained in electronic documents. Contents retrieval for document image, also known as document image understanding, is collection of inter-dependent processes of page payout decomposition, logical component labeling, search-based information retrieval, and OCR. However, there are no reliable methods applicable across the broad range of document types [2]. This is due to the fact that no reliable document segmentation methods have been established, which is well admitted as mathematically challenging problem [3].

For the cases of formatted (or semi-formatted) documents, document segmentation can be omitted (or minimally used). As an example, a typical invoice is formed with the structured tables and it can be easily formatted into the document-units without applying the complicated segmentation processes. As a consequence, one can concentrates on the study of contents retrieval algorithms from document images without burden of tasks for document segmentation.

In aspect of the contents retrieval, the target information to be extracted from an invoice are the PO (purchase order) number, the customer num-

※ Corresponding Author : Hyunkyung Shin, Address 65 Bokjung Dong, Sujenog Gu, Seongnam City, Gyeonggi Do, TEL :+82-31-750-8674, FAX : +82-31-750-8864, E-mail : hyunkyung@ kyungwon.ac.kr
Receipt date : Oct. 5, 2010, Revision date : Dec. 29, 2010
Approval date : Jan. 5, 2011
† Dept. of Mathematics & Information, Kyungwon University (hyunkyung@kyungwon.ac.kr)
※ This research was supported by the Kyungwon University Research Fund in 2010.

ber, the invoice date, the total amount, the freight, and the detailed list of purchased items consisted of the fields such as the unit cost, the quantity, the discount, and the total extension. The conventional way is to apply KWIC (key word in context). In order to define context around the keyword, the text line is a good candidate. For the study of this paper we obtained the text lines using a simple document segmentation method as described in the section 3.

In this paper we propose a CART based decision tree model [4,5] for an essential stage of invoice recognition system development, classification of the category of text lines in invoice document. We also present the result of comparison study with various machine-learning-based classification methods. Successful classification of the text lines is important since it provides the visual spatial intelligence for the target information to be etxracted. For the comprehensive study, we compared with LDA based machine learning methods such as SVM, naïve Baysean, and k-nearest neighbor [6] since our method is a non-LDA method.

The rest of this paper is organized as follows: in section 2 some of the related research works were introduced, in section 3 a procedure of invoice document processing was explained, in section 4 details of model and parameter selection was described, in section 5 analysis on the training and classification errors was presented, in section 6 a brief discussion on this project was placed.

2. RELATED WORKS

Previous researches on invoice document understanding are roughly divided into the three parts: text classification, machine learning techniques for document classification, and invoice recognition system.

Text classification is basic technology for general document analysis and understanding as well as invoice. [1,2] and [7] are part of researches on this subject. For classification techniques, Belaïd proposed morphological tagging approach for invoice document analysis, which is bottom-up without a-priori template [8]. Nielson and Barrett presented a template based layout zoning method [9] and Kotsiantis considered induction classification algorithms [4], concluded that SVM and MLP were superior to logic based (non-LDA) tree methods when dealing with multi-dimensional continuous (ordinal) features. While in this paper we will present on the contrary - for intermediate complexity of feature vector, non-LDA based tree classifier were superior to SVM. Cesarini, et. al. introduced case dependent domain knowledge and applied to invoice document as a case-study [10]. Shin developed fast and robust text line segmentation as a pre-processing stage for invoice recognition [11].

Among the researches on invoice recognition system, Ming et. al. proposed whole block moving method for slant image to deal with chinese invoice, the pre-processed invoice is processed using pre-compiled template library [12]. Hamza et. al. developed case based reasoning for document invoice analysis which is basically an auto-templating method [13]. Sako et. al. studied form data identification problem with the target ROI extraction using keyword matching, knowledge base character string recognition [14]. Chen and Blostein reported survey of document image classification emphasizing three components: problem statements, classifier architecture, and performance evaluation [15].

3. INVOICE RECOGNITION PROCEDURE

Commonly, invoice recognition system identifies a field info, purchase order (P.O.), for labeling of the invoice which should be matched with electronic document system. Once P.O. field is successfully identified, it carries out the page segmen-

tation (zoning) in order to detect the product details inside of the invoice. This paper describes the latter process which consists in 5 stages: 1) text line segmentation, 2) pre-validation, 3) text line classification, 4) validation, 5) information parsing.

For the first stage, we applied the text line segmentation using the 3 primary AC coefficients of 8x8 DCT (discrete coefficient table) blocks, which represents strong edges, to find the white strips between two text lines. It is basically projection method with consideration of periodicity of the projection profile. The underlying idea is the Markov modeling to recover missing white strips and avoid multiple selections of maxima in the density estimation [11]. We used the mean shift for finding the maximum of density function.

Once the white spaces were detected between the text lines, we registered the coordinate information in form of B-tree with depth of 4, i.e. a tree with the same depth and with the inconsistent number of child nodes. In the B-tree the root node represents a whole page of document, its first child is paragraph node. A paragraph node contains the line nodes and the order of child nodes is arranged in terms of y-position. Having constructed the B-tree, we merged the OCR (optical character recognition) text information provided as input into the tree structure. Using the texts, we created word node as the child of the line node and we arranged the node order in terms of x- coordinate position of word bounding boxes. In this way we had semi-structured document layout of the page so that we could easily transverse a document page.

At the second stage, we applied pre-validation to check whether the input document is an invoice or not. The prior information was that one of the text lines should contain product information, P.O. number. If we validated this configuration then the process continued or stopped.

At the third stage, we applied information retrieval technique using decision tree type non-linear discriminant analysis to search the header of the table and the summary of table. As the results the header and summary information provided the table regions. the text lines within the table region could be considered with high probability as the product details. At the forth stage, the results of the third stage were subject to be validated with the a-priori formula relation, unit cost * quantity = amount.

At the final stage, based on the validated lines, using correlation of word bounding boxes, we could divide the line into the group of the columns. We found the three columns which represent unit cost, quantity and total extension amount.

In [Figure 1] illustrates the invoice recognition system. The real lines represent the flow direction of information data, and the dotted lines represent the system interruption routine.

4. MODEL DESCRIPTION

In this section we explain the details of the methods employed to the stage 3 of invoice text line classification. For the classification process we used CART based decision tree technique [5]. Construction of training-data is explained as below.

4.1 Feature vector construction

As seen in [Figure 2] below, the output of text line segmentation, line node, contains the list of words, word node.

For the study of this paper the feature vector was taken as an array of raw text words from a text line. In the figure, the green color represents bounding box of line node and the blue color represents word bounding box. However most of the machine-learning engines require the fixed size array as input. We defined a constant value of array size as 16, MAX_FT_SIZE = 16. If the number of text words exceeded MAX_FT_SIZE, we cut off the rest and if the number of text words fell behind

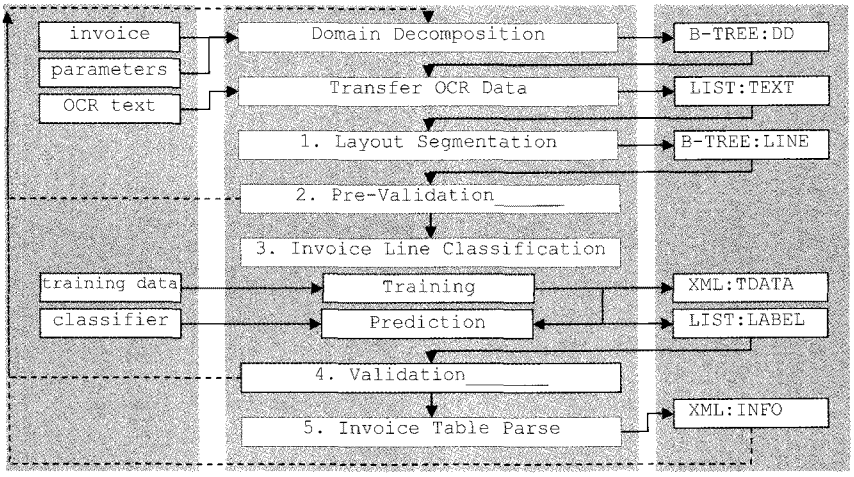


Fig. 1. diagram view of invoice recognition system.

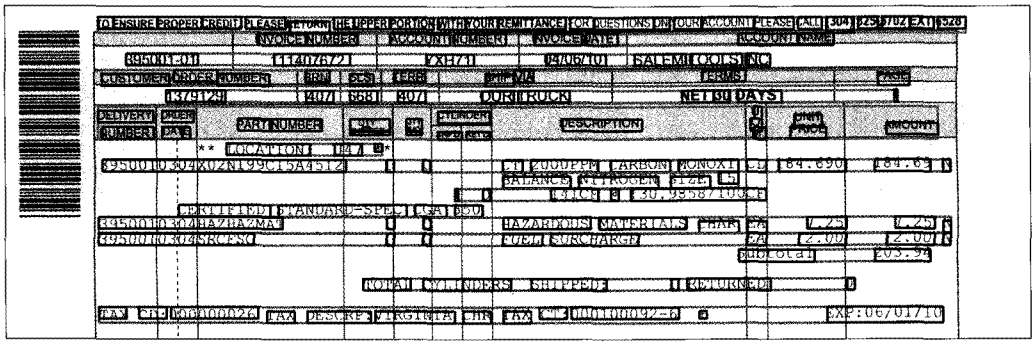


Fig. 2. an output of text line segmentation for invoice.

MAX_FT_SIZE, we filled with empty strings. [Figure 3] visualizes the output of feature vector extraction process. For an example, given the four text lines as seen at the upper part, the four feature vectors were created as seen at the lower part.

4.2 Training data construction

A training data is a feature vector with label. In this paper, for the labeling of training data, we classified the text lines into the six classes: 1) invoice PO header, 2) invoice table header, 3) summary header, 4) invoice surcharge header, 5) invoice contents, and 6) the rest. There is no continuity among the values of class labels, which requires the label should be categorical variable. [Figure 4] illustrates output of training data. In the

Output of the text line segmentation

shipped	backordered	description	unit	price	net	price	total
0	0	0	0	0	straight	flute	0
0	0	note	UPS	COLLECT	0	0	0
0	0	note	shipped	from	factory	0	0

Formation of feature vector from the text line

Fig. 3. example of feature vector construction from text lines.

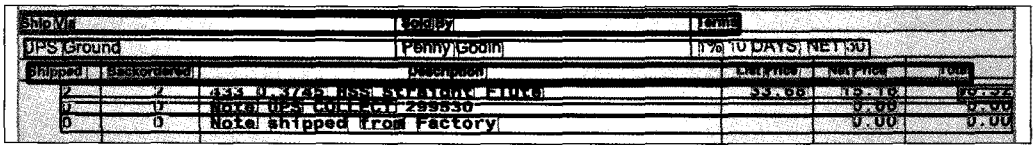


Fig. 4. an example of training data-color coded.

figure the red color indicates the case 1 - invoice table header, the brown color indicates the case 3 - invoice PO header, and the blue color indicates the default.

4.3 Classification model selection

For the implementation of text lines classification, in this paper, we employed the decision tree model for the process of text line classification. Some details on the model parameter selection will be described in this section. In addition to analysis on the decision tree, as seen in the [Figure 5] we also, for the comparison study of classification performance level through various machine learning models, utilized the five more classifier models such as naïve bayesean classifier, k-nearest neighbor, SVM, random tree, and boost.

5. EXPERIMENTS

5.1 sample data acquisition

For the quantitative assessment of statistical estimation of classification error, we randomly sampled out the number 1,420 of invoice documents from an archive. The maximum number of the invoices from the same vendor (same type of invoice) is less than 10 so that the total number of distinct invoices is greater than 150. The total number of text lines from the invoice data set is approximately 30,000.

5.2 measure of fit and cross validation

In order for unbiased error analysis to apply generally accepted 10-fold cross-validation, we divided the invoices into the 10 groups [16,17]. For

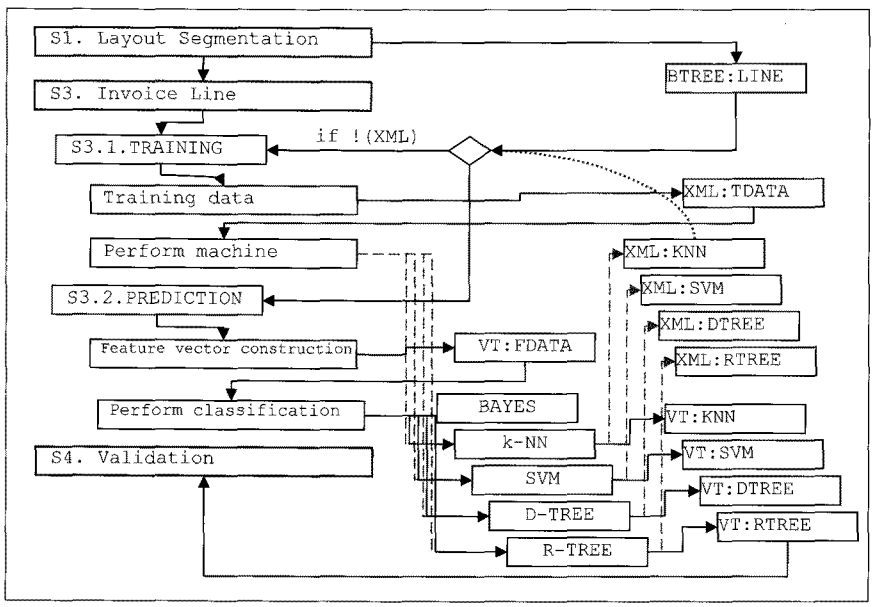


Fig. 5. a brief diagram view of text line classification performance comparison module using multiple machine learning techniques.

the measures of fit, we used the mean square error (MSE) where we utilized two kinds of errors defined as follows:

PageError(page)	= 0 if the table headers in an invoice were identified correctly
	= 1 otherwise
LineError(line)	= 0 if the classified label of text line was same with training data
	= 1 otherwise

For each type of errors defined above, estimation of the MSE is as follows:

MSE-page = $\sum_i$(PageError(i))^2/N, where _i = 0, ···, N (number of pages tested)
MSE-line = $\sum_i$(LineError(i))^2/M, where _i = 0, ···, M (number of lines tested)

With the measures of fit, two types of error estimation were defined: precision and recall rates. Consider a two-category truth table with TT, TF, FT, and FF. TT indicates the count of samples identified as T where the ground truth value is T, TF indicates the count of samples identified as T where the ground truth value is F, FT the count of samples identified as F while the ground truth value is T. In this case, the precision is TT/(TT+FT) and the recall rates is (1-TF/(TT+TF)).

5.3 error analysis

Both errors occurred at training stage (training errors) and classification stage (classification errors) were summarized in Table 1 and Table 2, respectively, in which the error values were pre-

Table 1. Training errors: 1-precision and recall rates in terms of size of training data

#data	1,558	4,438	5,122	6,749	10,000	12,745	16,015	22,764
DTREE	0.162	0.126	0.128	0.046	0.125	0.051	0.091	0.076
RTREE	0.132	0.170	0.140	0.141	0.201	0.181	0.107	0.140
BOOST	0.015	0.039	0.064	0.016	0.075	0.027	0.038	0.035
SVM	0.029	0.017	0.055	0.003	0.198	0.022	0.038	0.030
KNN	0.191	0.183	0.132	0.177	0.168	0.148	0.109	0.112
BAYES	0.338	0.417	0.345	0.267	0.384	0.384	0.410	0.394
1-precision (MSE-line)								
#data	1,558	4,438	5,122	6,749	10,000	12,745	16,015	22,764
DTREE	0.574	0.594	0.66	0.574	0.575	0.576	0.594	0.611
RTREE	0.843	0.884	0.860	0.876	0.851	0.877	0.901	0.914
BOOST	0.519	0.646	0.746	0.432	0.758	0.736	0.849	0.741
SVM	1.000	0.983	1.000	0.997	0.996	0.990	0.995	0.994
KNN	0.965	0.935	0.944	0.903	0.929	0.945	0.948	0.952
BAYES	0.652	0.757	0.851	0.874	0.760	0.729	0.801	0.817
recall rates (MSE-line)								
#data	1,558	5,122	6,749	10,000	16,768	21,206	22,764	
DTREE	0.811	0.811	0.811	0.811	0.767	0.767	0.767	
RTREE	0.791	0.842	0.814	0.814	0.932	0.908	0.926	
BOOST	0.664	0.854	0.909	0.892	0.950	0.963	0.968	
SVM	0.072	0.203	0.239	0.324	0.368	0.417	0.424	
KNN	0.700	0.871	0.833	0.854	0.897	0.921	0.928	
BAYES	0.722	0.716	0.711	0.712	0.712	0.711	0.709	
1-precision (MSE-page)								
#data	1,558	5,122	6,749	10,000	16,768	21,206	22,764	
DTREE	0.659	0.659	0.659	0.659	0.660	0.654	0.654	
RTREE	0.732	0.731	0.763	0.828	0.831	0.871	0.862	
BOOST	0.424	0.691	0.377	0.730	0.800	0.694	0.733	
SVM	0.965	0.981	0.963	0.946	0.969	0.966	0.967	
KNN	0.864	0.844	0.874	0.908	0.902	0.915	0.903	
BAYES	0.641	0.736	0.804	0.720	0.841	0.835	0.838	
recall rates (MSE-page)								

Table. 2 precision and recall rates of line classification in terms of the size of training data

#data	1,558	5,122	6,749	10,000	16,768	21,206	22,764
DTREE	0.811	0.889	0.842	0.858	0.767	0.854	0.887
RTREE	0.791	0.842	0.814	0.814	0.932	0.908	0.926
BOOST	0.664	0.854	0.909	0.892	0.950	0.963	0.968
SVM	0.072	0.203	0.239	0.324	0.368	0.417	0.424
KNN	0.700	0.871	0.833	0.854	0.897	0.921	0.928
BAYES	0.722	0.716	0.711	0.712	0.712	0.711	0.709
precision (MSE-line)							
#data	1,558	5,122	6,749	10,000	16,768	21,206	22,764
DTREE	0.759	0.759	0.759	0.759	0.766	0.754	0.754
RTREE	0.732	0.731	0.763	0.828	0.831	0.871	0.862
BOOST	0.424	0.691	0.377	0.730	0.800	0.694	0.733
SVM	0.965	0.981	0.963	0.946	0.969	0.966	0.967
KNN	0.864	0.844	0.874	0.908	0.902	0.915	0.903
BAYES	0.641	0.736	0.804	0.720	0.841	0.835	0.838
recall rates (MSE-line)							

sented in terms of the size of training data. The range of size of training data (text line) was from about 1,500 to 23,000.

5.3.1 training error analysis

Training errors were evaluated by 1 - precision and recall rates. Higher level of training errors typically indicates that the underlying model surpasses the stage of saturation and reaches to over-fitting state. As a whole purpose of employing cross validation, the less training error does not guarantee the less classification error. In this section we tried to see capacity of model complexity.

As seen in the Table 1, SVM outperformed the others with the lowest 1-precision and the highest recall rates. Boost performed as good as SVM in precision but performed poor (60% range) in recall rates. Decision tree performed little worse than boost. Random forest and kNN had similar precision and recall which were lesser than decision tree. Naïve bayesean classifier had the lowest precision and poor recall.

[Figure 6] illustrated the correlation between size of training data and training error level. . What we observed was that size-to-error correlation was less significant than model-to-error relation. SVM and boost had excellent precision followed by decision tree

5.3.2 classification error analysis

As seen in the Table 2, SVM performed classification process with the worst precision (but still the best recall rates). As an interesting behavior of SVM, the precision was improving linearly as the size of training data, implied that model complexity of SVM seems much higher than the other competitors. Boost also showed linear correlation in precision with the size of training data but not significantly as seen in SVM. However in recall rates, boost showed poor rates when the size of training data is small (less than 10, 000). For the rest of classifiers, the precision and the recalls were steady in terms of training data size. These implied that SVM and boost required larger training data set (22,000 or more) while the rest of classifiers needed smaller set (around 5,000) to reach their full capacity.

5.3.3 complexity of feature vector

Relationship between feature vector dimension and classification was investigated and summarized in Table 3. SVM showed fine performance for smaller feature vector dimension but very low performance for larger feature vector dimension,

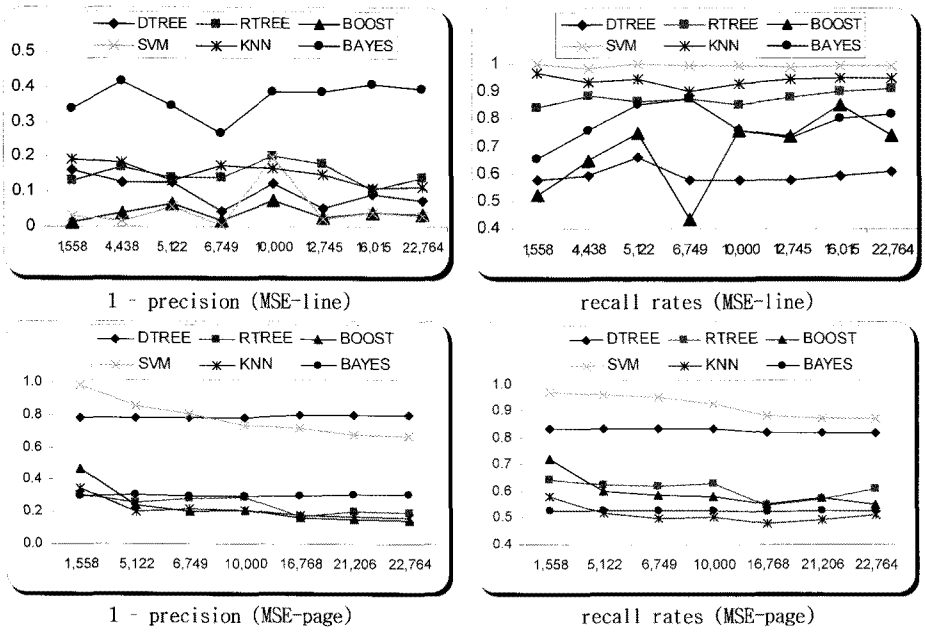


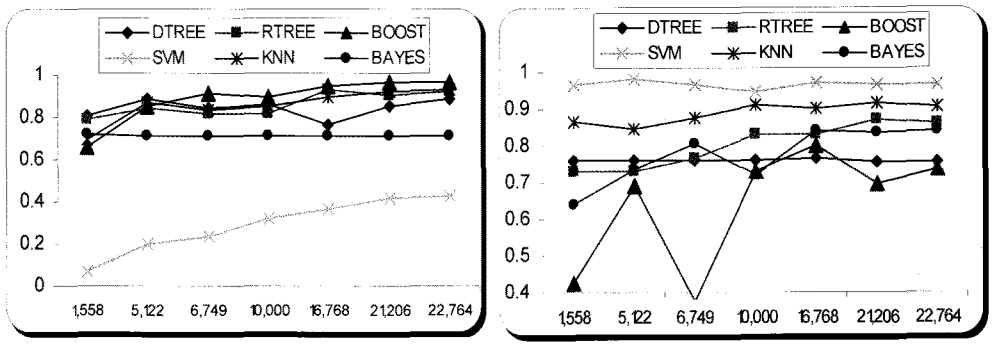
Fig. 6. training error and recall rates in terms of training data size.

Table 3. precision and recall rates of line classification in terms of the size of feature vector

#FV dim	4	6	8	10	12	14	16
DTREE	0.833	0.912	0.893	0.956	0.895	0.893	0.811
RTREE	0.780	0.866	0.786	0.816	0.842	0.851	0.863
BOOST	0.837	0.921	0.922	0.922	0.930	0.938	0.942
SVM	0.775	0.589	0.421	0.384	0.383	0.378	0.378
KNN	0.824	0.900	0.875	0.883	0.882	0.871	0.872
BAYES	0.353	0.582	0.676	0.704	0.714	0.774	0.768
precision (MSE-line)							
#FV dim	4	6	8	10	12	14	16
DTREE	0.719	0.719	0.719	0.719	0.719	0.719	0.719
RTREE	0.851	0.874	0.885	0.835	0.892	0.870	0.853
BOOST	0.906	0.897	0.848	0.886	0.730	0.732	0.736
SVM	0.948	0.952	0.963	0.970	0.972	0.972	0.972
KNN	0.930	0.905	0.952	0.938	0.929	0.929	0.929
BAYES	0.949	0.958	0.960	0.911	0.739	0.509	0.746
recall rates (MSE-line)							

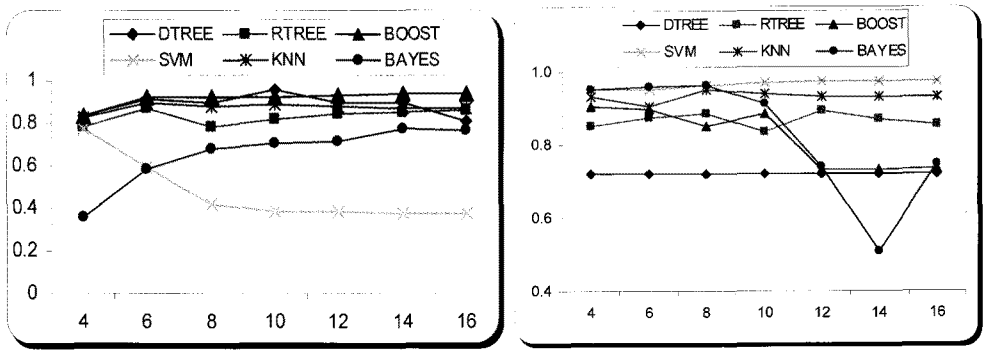
which implied that it required larger training data set to achieve its full classification performance for this type of problem. If we looked at CART based classifiers (decision tree, random forest, and boost), boost and random forest were improved as the size of feature vector grew, but decision tree had its maximum performance for medium size feature vector (10). This clearly proved that ensemble of decision trees was actually effective to complexity of feature vector. However, decision tree had the best overall accuracy over random tree and boost for this type of problem.

As seen in [Figure 8], naïve Baysean, random tree and boost had linear dependency between accuracy and dimension of feature vector. The dependency was stronger in Baysean and was weak-



Precision (MSE-line) recall rates (MSE-line)

Fig. 7. classification precision and recall rates in terms of the size of training data.



precision (MSE-line) recall rates (MSE-line)

Fig. 8. precision and recall rates of line classification in terms of the size of feature vector.

er in random tree and boost. The rest of classifier models did not get benefits from complexity of feature vectors. Except for the case of naïve Baysean, recall rates were almost statistically independent on complexity of feature vector. Naïve Baysean suffered with lower recall rates as the dimension of feature vector increased. This implied that higher degree of feature vector dimension was not reflected into the dimension of decision boundary of Baysean classifier.

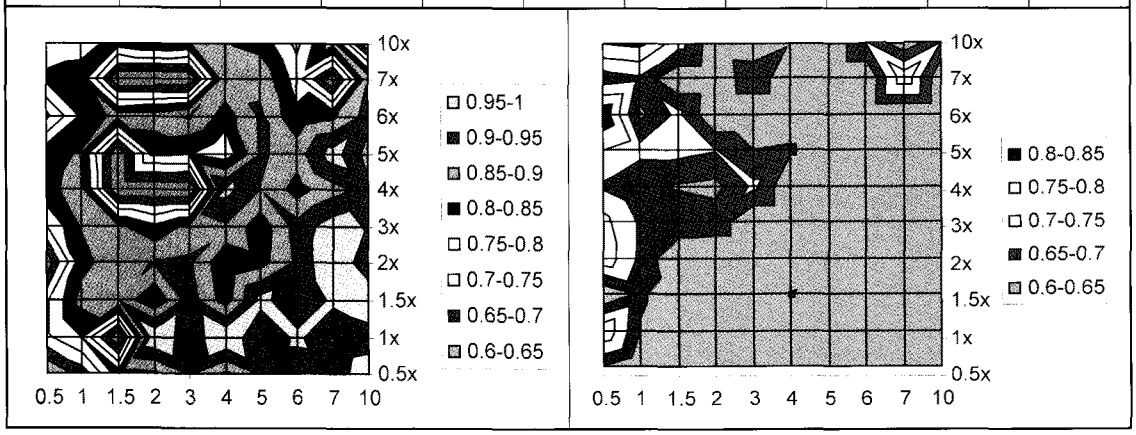
5.3.4 model parameter selection for decision tree

As emerged in the previous section, among the line of non-LDA and CART based classifiers, there were no significant frontrunner. This was because the underlying problem of text line classification was not complicated enough to see benefits from ensemble of weak classifier trees. In this section, we investigated model parameters used in decision tree which were also core elements of random forest and boost.

The prior vector as a part of parameters of decision tree model is highly sensitive to its selection, requiring heavy amount of training process. For the study of this paper, we employed four category problems which requires four-dimensional vector for the priors- In Table 4 we presented the precision and the recalls in terms of the second and fourth elements of the priors. At the upper panel of the table, we presented a occurrence matrix between the second and the fourth feauture vector elements in the way of the following. At default, the prior value for each element of feature vector is [1, 1, 1, 1]. For the purpose of studying on the effect of the prior, we use [1, 0.5, 1, 0.5], [1, 1, 1, 0.5], [1, 1.5, 1, 0.5], ···, [1, 10, 1, 10]. The column

Table 4. effect of priors to performance of decision tree

	0.5	1	1.5	2	3	4	5	6	7	10
0.5x	0.396	0.674	0.826	0.837	0.904	0.834	0.908	0.910	0.916	1.000
1x	0.738	0.800	0.365	1.000	0.918	1.000	0.898	0.964	0.894	0.918
1.5x	0.738	0.868	0.896	0.800	0.924	0.823	0.969	0.902	0.963	0.998
2x	0.607	0.837	0.874	0.902	0.910	0.806	0.831	0.907	0.998	0.910
3x	0.752	0.854	0.862	0.811	0.868	0.891	0.821	0.919	0.980	0.943
4x	0.840	0.866	0.391	0.385	0.391	0.967	0.898	0.828	0.882	0.916
5x	0.834	0.854	0.385	0.812	0.803	0.758	0.907	0.877	0.977	0.876
6x	0.683	0.807	0.870	0.877	0.905	0.826	0.907	0.812	0.842	0.921
7x	0.815	0.840	0.396	0.394	0.387	0.887	0.890	0.776	0.388	0.899
10x	0.756	0.766	0.829	0.854	0.849	0.852	0.832	0.736	0.874	0.385



indicates the prior of the second element while the row indicates that of the fourth element. At the lower panel of the table, the the matrix was illustrated by the two dimensional graphs. We should mention that our study showed that the first and the third element had less significant dependence the second and the fourth. As can it be seen in the left graphic panel at the bottom of table, the decision tree had higher precision when the second element and the fourth element have the range from 5 to 10 and from 0.5 to 2, repsectively. At the right graphic panel we presented the recalls. The matrix showed that performance rates of decision tree is quite dependent on the priors as the magnititude of discrepancy is about 0.6. From this data, we concluded that assignment of appropriate prior values must be scrutinized as an important training stage.

The most important model parameter for CART based classifier is the criterion on splitting node. In this study, we employed both entropy and minimum number of samples in a node. In [Figure 9] the precision and the recalls were plotted in terms of the size of minimum samples. As can it be seen, the precision is almost steady but the recalls were significantly worse as the size became larger. For the purpose of clarification, we added the trend lines using moving average of length 2 (2 per. Mov. Avg).

6. DISCUSSION

In this paper various machine learning methods were investigated for the case of non-uniform sized feature input vectors. In order to avoid the bias due to data mining process, we intentionally used the raw texts for the construction of feature vector instead of using multiple layered data mining processes for feature extraction. The experiments showed that CART based methods were

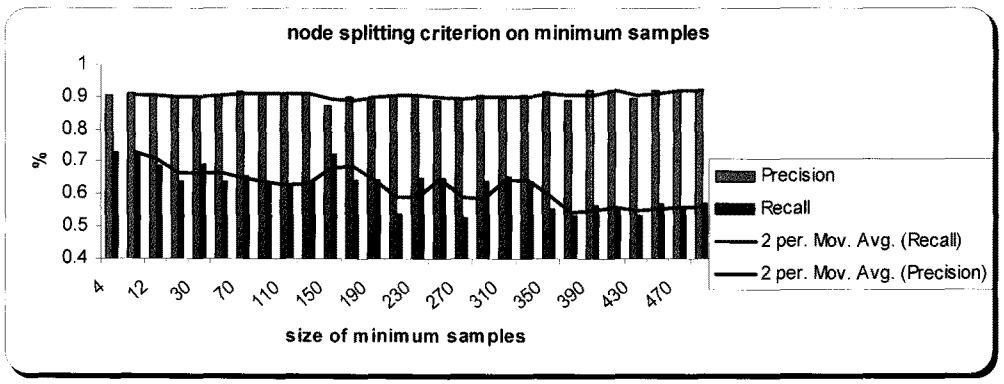


Fig. 9. effect of node splitting criterion on selection of minimum samples.

more consistent with precision and recalls than LDA (linear discriminant analysis) based methods.

It is well known that k-NN, the simplest machine learning technology, gives wrong impression with very low training error level. The cross validation usually elucidates this illusion. Our study showed that CART based methods were superior to the LDA based method - SVM, naïve Baysean -and the memory association based method k-NN. This result may be originated from the data mining strategy adopted in our study [4].

This study is a part of invoice recognition system development project. We concentrated on the text line classification. Once the text line is classified into the pre-defined categories, to extract the target information from the invoice, the KWIC (keyword in context) will be applied to the subset of text lines by the categories, which will have improved overall performance.

REFERENCE

[1] S. Büttcher, C. L. A. Clarke, and G. V. Cormack."*Information Retrieval: Implementing and Evaluating Search Engines*," MIT Press, Cambridge, MA, 2010.

[2] H. Baird, D. Lopresti, B. Davison, and W. Pottenger, "Robust document image understanding technologies," Proc. of ACM HDP Workshop, USA, pp. 9-14, 2004.

[3] I. Witten, A. Moffat, and T. C. Bell, "*Managing Gigabytes: Compressing and Indexing Documents and Images,*" Second Edition, Morgan Kaufnann Publishiers, New York, NY, 1999.

[4] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques," *Informatica*, Vol. 31, pp. 249-268, 2007.

[5] L. Breiman, J. H. Friedman, R. A. Olshen, and C.J. Stone, "*Classification and regression trees,*" Monterey, CA: Wadsworth & Brooks/ Cole Advanced Books & Software, New York, NY, 1984.

[6] S. Haykin, "*Neural Networks-A Comprehensive Foundation,*"second ed. Prentice-Hall Inc., Upper Saddle River, NJ, 1998.

[7] Y. Ishitani. "Model-based information extraction method tolerant of OCR errors for document images." *Int. J. Comput. Proc. Oriental Lang.*, vol. 15(2) pp. 165 - 186, 2002.

[8] Y. Belaïd and A. Belaïd, "Morphological Tagging Approach in Document Analysis of Invoices," Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), 2004.

[9] H. E. Nielson and W. A. Barrett, "Consensus-Based Table Form Recognition," ICDAR, Edinburgh (Scotland), pp. 906-910 , 2003.

[10] F. Cesarini, E. Francesconi, M. Gori and G. Soda, "Analysis and Understanding of

Multi-Class Invoices," *IJDAR*, 2003.

[11] H. Shin, "Fast Text Line Segmentation Model Based On DCT For Color Image," *KIPS*, Volume 17-D, Issues 6, 2010.

[12] D. Ming, J. Liu, and J. Tian, "Research on Chinese financial invoice recognition technology," *Pattern Recognition Letters*, Vol. 24, Issues 1-3, pp. 489-497, 2003.

[13] H. Hamza, Y. Belaid and A. Belaid, *"Case-Based Reasoning for Invoice Analysis and Recognition,"* LECTURE NOTES IN COMPUTER SCIENCE, N.o.4626, pp. 404-418, 2007.

[14] H. Sako, M. Seki, N. Furukawa, H. Ikeda and A. Imaizumi, "Form Reading based on Form-type Identification and Formdata Recognition," In International Conference on Document Analysis and Recognition, Edinburgh (Scotland), pp. 926-930, 2003.

[15] N. Chen and D. Blostein "A survey of document image classification: problem statement, clas-sifier architecture and performance evaluation," *IJDAR*, vol.10, pp.1-16, 2007.

[16] R. R. Picard and R. Dennis Cook, "Cross-Validation of Regression Models," *Journal of the American Statistical Association* 79 (387): pp. 575-583, 1984.

[17] D. J. Hand, H. Mannila, and P. Smyth, *"Principles of Data Mining,"* MIT Press, Cambridge, MA, 2001.

Hyunkyung Shin

received her Ph. D in applied mathematics and statistics from State University of New York at Stony Brook. Since 2007 she joined in department of mathematics and information of Kyungwon Universtiy as an assistant professor. Her research interests are theory of computation, image processing, computer vision, and machine learning based on neural networks.