MULTILABEL CLASSIFICATION OF DOCUMENTS WITH MAPREDUCE

P.Malarvizhi [#1] Ramachandra V.Pujeri [*2]

# Research Scholor, Department of Computer Science and Engineering, Karpagam university , Coimbatore, Tamilnadu, India

* Vice Principal, KGISL Institute of Technoloy, Coimbatore, Tamilnadu, India

1 malarvizhi_s@yahoo.co.in

2 sriramu_vp@kggroup.com

***Abstract*—Multilabel classification is the problem of assigning a set of positive labels to an instance and recently it is highly required in applications like protein function classification, music categorization, gene classification and document classification for easy identification and retrieving of information. Labeling the documents of the web manually is a time consuming and a difficult task due to the size of the web which is a huge information resource and to overcome this difficulty, we propose an algorithm of MapReduce for classifying labels to the documents of the web. MapReduce is a framework of parallel programming model with the functions map and reduce and meets a number of varieties of applications. In our approach, the documents of the web are given to the MapReduce framework and the MapReduce framework assigns the set of positive labels to the documents of the web using binary classification of binary classifier. On experimentation, our proposed approach satisfactorily classifies the labels to the documents of the web.**

Keyword-Multilabel classification, MapReduce, Problem transformation, Binary classifier, Binary classification

I. INTRODUCTION

Recently, the need for multilabel classification is highly increasing in applications like text categorization, functional genomics and scene classification in which several examples may belong to more than one labels simultaneously. There are many interesting problems that require multilabel classification [1] in which, the examples are associated with a set of labels [2]. Multilabel classification was primarily motivated by the emerging need for automatic text categorization and medical diagnosis [3]. Text documents are considered as natural multilabel problems which belong to more than one conceptual class [4] and the textual data, such as documents and web pages, are frequently associated with more than a single label. For example, a news paper article with the reactions of the Christian church to the release of the “Da Vinci Code” film can be labeled as both religion and movies and the classification of textual data is the dominant multilabel application [5].

Problem transformation and algorithm adaptation are the methods existing for handling multilabel classification and the two problem transformation methods are Label Powerset (LP) and Binary Relevance (BR) in which the problem transformation splits the multilabel learning problem into one or more singlelabel problems[6]. In general, the multilabel classification is performed by Problem Transformation (PT) method which turns the multilabeled training data into a singlelabel to train one or more singlelabel classifiers and the singlelabel classifiers output is then combined to have a multilabel representation [7]. The most well known problem transformation method is the binary relevance method (BM) [2], [8], [9] which transforms any multilabel problem into one binary problem for each label where it trains |L| binary classifiers C1, · · · ,C|L| in which each classifier Cj predicts the 0/1 association for each corresponding label ljεL [10]. A straight forward approach for addressing multilabel classification is to model each class independently. In the binary relevance problem transformation method, one binary classifier is trained independently for each possible label, in which all training examples for which the label is relevant are used as positives examples and all other examples as negative examples [11].

In our approach, we have used the mapreduce framework for classifying labels to the documents and the mapreduce framework can be used for a varieties of data intensive and compute intensive applications [12]. The Google’s mapreduce programming model is quite suitable for processing large data sets [13] where the computation is splited into small tasks to run in parallel on several machines and to scale easily on very large clusters of low cost commodity computers [14]. The abstraction of the mapreduce framework allows the application developers to focus on their application [15] in which when a problem is specified in a mapreduce form, it is easy to parallelise the computation, distribute data to the processors and to load balance between them. The details concerning all these issues are hidden from the user and opportunities for task level and instruction level parallelization are easily identified [16]. The map function processes the input data to generate a set of intermediate key/value pairs which are then merged by the reduce function for the same key. The computation is parallelised automatically by the mapreduce by running multiple map and/or reduce tasks in parallel over disjoined portions of the input or intermediate data [17].

In the proposed approach, the documents to be classified are given to the map function of the mapreduce framework which includes the functions map and reduce and the map function generates a (key, value) pair for each document and is given to the reduce function. The reduce function the binary classifier equals the number of labels, performs the binary classification for each document and outputs a binary value of 0 or 1 and the proposed approach assigns only the positive value labels to the documents.

The rest of the paper is structured as follows: Section 2 gives a brief review of the related task while Section 3 describes an overview of mapreduce and multilabel classification . Section 4 introduces the proposed approach and Section 5 shows the experimental results. Finally, Section 6 concludes the paper.

II. RELATED WORK

Bishan Yang et al. [18] have proposed a novel multilabel active learning method to minimize the human labeling efforts which reduces the required labeled data without sacrificing the classification accuracy. In single label problems, each data is associated with one label which is handled by traditional active learning algorithms. Their approach considers the multilabel information and focus to label data which optimizes the expected loss reduction in multilabel information, where they have optimized the reduction rate of the size of version space with Support Vector Machines (SVM). They have also designed an effective method to predict possible labels for each unlabeled data and the expected loss is approximated by summing up losses on all labels according to the most confident result of label prediction. Experiments on seven real world data sets shows that their approach provides better classification result with much fewer labeled data than state of the art methods.

Akinori Fujino et al. [19] have designed a multilabel classification system for classification of patent retrieved at NTCIR-6 with the combination of binary classifications in which there is a binary classifier per Fterm that determines the assignment of F-term to patent documents. They have also constructed hybrid classifiers by combining the component generative models with weights based on the maximum entropy principle as binary classifiers to effectively use the multiple components of patent documents. They have confirmed that their system provides good ranking of F-terms in assigning them to patent documents with a test collection of Japanese patent documents.

Grigorios Tsoumakas et al. [20] have contributed a novel algorithm for effective and computationally efficient multilabel classification in domains with large label sets L. Hierarchy of Multilabel classifiers was constructed by the HOMER algorithm where each one deals with a much smaller set of labels compared to L and a more balanced example distribution which improves predictive performance along with linear training and logarithmic testing complexities with respect to |L| .

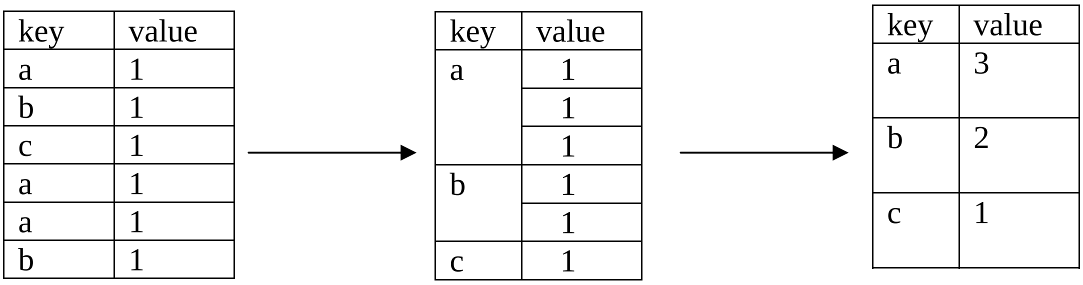
Tamer Elsayed *et al.* [21] have presented a MapReduce algorithm for computing pairwise document similarity in large document collections that permits to separate the inner products involved in computing document similarity into separate multiplication and summation stages in a way that is well suited for efficient disk access patterns across several machines and their algorithm shows linear growth in running time and space in terms of the number of documents with the collection of approximately 900,000 newswire articles.

III. OVERVIEW

This section presents a brief overview of mapreduce and multilabel classification.

A. Mapreduce

Mapreduce was developed by Google for processing of huge amounts of raw data for example, crawled documents or web request logs that must be distributed over multiples of machines for processing with in a reasonable time and in this distribution the parallel computing the same computations are performed on each cpu with a different data set. The distributed nature and the abstraction of the mapreduce allows the computation task simple and hides the details of parallelization, data distribution, load balancing and fault tolerance [22] . The input key/value pairs are processed by map function to generate intermediate key/values which are then merged by reduce function to generate output for the same key [23] is shown in figure 1 .



key	value
a	1
b	1
c	1
a	1
a	1
b	1

key	value
a	1
	1
	1
b	1
	1
c	1

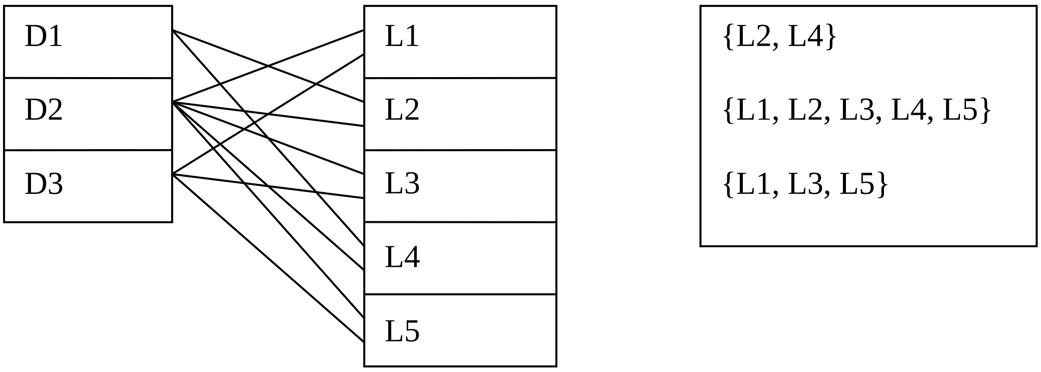
key	value
a	3
b	2
c	1

Map (key, value) intermediate(key, value) reduce (key, value)

Fig 1. Example of Mapreduce

B. *Multilabel classification*

Multilabel classification is the process of assigning an instance simultaneously to one or more classes in which a binary classifier is learned independently for each class to assign a test instance all of the class labels for which the corresponding classifier says 'yes' [24]. The aim of multilabel classification is to have simultaneously a collection of binary classifications in which the positive classes are the relevant labels for the instances and the methods used to handle multilabel classification tasks fall into two groups [2], [5] in which the first group transforms the learning tasks into a set of singlelabel of binary or multiclass classification tasks. Binary Relevance (BR) method is the most simple and very effective common transformation strategy where each label is classified as relevant or irrelevant without any relation with the other labels [25] and is shown in figure 2 .



Examples Labels Set of labels

Fig 2. Example of multilabel classification

IV. MULTILABEL CLASSIFICATION OF DOCUMENTS WITH MAPREDUCE

MapReduce meets wide varieties of applications and the proposed approach is executed in the distributed environment using MapReduce framework. Under this MapReduce programming model, an application is executed as a sequence of mapreduce operations of a Map phase and a Reduce phase which process a large number of independent data items where the system supports automatic parallelization, distribution of computations, task management and fault tolerance [12]. The map function of the mapreduce programming model takes the (key, value) as input and generates an intermediate (key, value list) as output. The reduce function of the mapreduce programming model takes the intermediate (key, value list) as input and lists the final value as output is shown in figure 3 and the example process of mapreduce framework for the proposed approach is shown in figure 4.

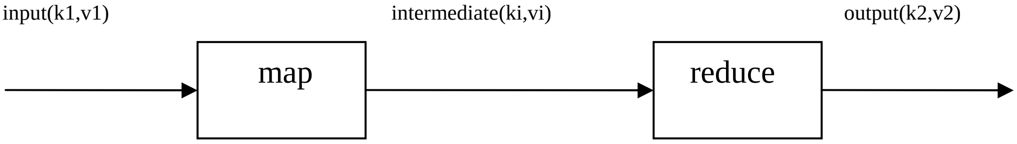


Fig.3. Mapreduce framework

Map: (k_1,v_1) -> $(k_i, v_i$ list)
Reduce: $(k_i, v_i$ list) -> (k_2,v_2)

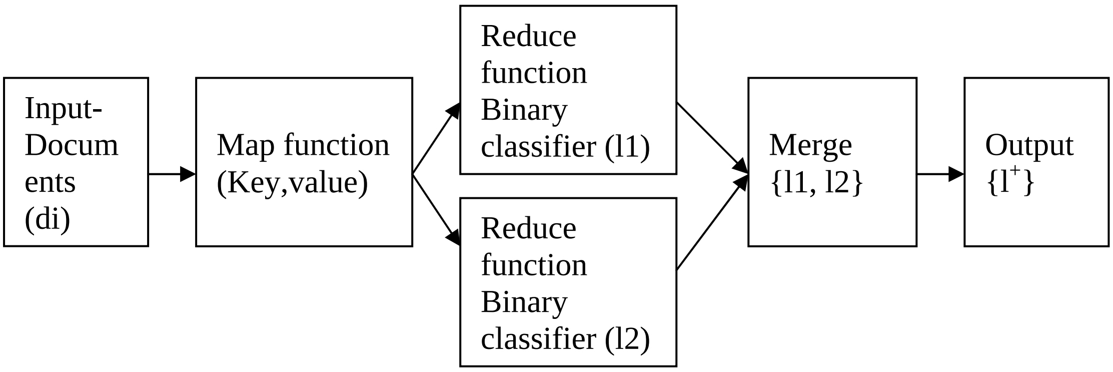

Input-Documents (di)
Map function (Key,value)
Reduce function Binary classifier (l1)
Reduce function Binary classifier (l2)
Merge {l1, l2}
Output {l+}

Fig.4. Example process of mapreduce framework for the proposed approach

In this figure, we have used one map function and two reduce functions for the classification of labels to the documents. The map function converts the documents d_i to be classified into (key, value) pair where the key, value refers the keyword and frequency count of the keyword and is given to the reduce functions the binary classifiers which computes the binary value 0/1 for each label. The merge function combines the binary value of all the binary classifiers for each document and assigns only the positive value labels to the documents. The distributed nature of the mapreduce framework reduces the computation task and the processing time for the classification of labels to the documents where the positive labels are assigned to the documents. The documents for classification are collected from the web using webcrawler and the collected documents are given to the proposed approach for classification of labels.

In the proposed approach, the documents d_i collected from the web $D=\{d_i \mid i=1\ldots M\}$ are represented as a set X the feature vector which contains a set of keywords $X=\{x_j \varepsilon D \mid j=1\ldots m\}$ and a label vector L which contains a set of labels represented as $L=\{l_k \mid k=1\ldots N\}$. For classification of labels to the documents, the binary classifier h performs the binary classification between the feature vector X and the predefined label vector L which contains N number of sets of labels $L=\{l_k \mid k=1\ldots N\}$ where each set contains a set of predefined label related keywords p_i with a weightage q_i, which provides a better classification results $l_k=p_i.q_i$; $i=1\ldots n$. The number of binary classifier h is the same as the N number of labels of L and for each label l_k one binary classifier h is constructed. Each binary classifer h_k predicts the binary value 1/0 of positive/non positive for each label l_k by comparing the label related keywords p_i of label l_k of label vector L with the keywords of X and

$h_k^+ = (\Sigma qi.S_{pi}^{(X)}$; $i=1\ldots n)\neq 0$. If pi similar with keywords of X, 1 is assigned for $S_{pi}^{(X)}$,

otherwise 0 will be assigned. The binary value 0/1 from all binary classifiers h_k are combined to determine the multilabels for the documents d_i and only the positive value labels l_k^+ are assigned to the documents $d_{i=}U\{ l_k^+ \}$ and multilabel classifier $H(d_i) =U\{ h_k^+\}$.

V. EXPERIMENTATION AND RESULTS

The documents are crawled from the web using webcrawler for classification and the web crawler WebSPHINX (Website-Specific Processors for HTML INformation eXtraction) is used to collect web pages from the web and are stored in a local repository. Crawling begins with by feeding the web crawler a set of seed pages which are a list of uniform resource locators (URLs) will start the crawling process [26]. The crawler parses the web page to extract the hyperlinks of incoming and outgoing for further crawling and then stores the crawled web pages into the local repository. Standard URL normalization is performed on extracted hyperlinks the URLs to identify equivalent URLs which link to the same web pages and will not be included in the to-crawl list of URLs for further crawling and this process is continued till the stopping criteria of the web crawler are met which is the number of web pages downloaded or the total file size will be used as the indicators to stop crawling.[27]

MapReduce framework is constructed based on the observation that several tasks have the same structure in which a computation is mapped over a large number of records like documents to generate partial results which are then aggregated in some order. The MapReduce framework meets more number of varieties of applications where the per-record computation and aggregation vary by task and the basic structure remains the same[21]. For the proposed approach, we have used four map functions and three reduce functions which run in thread parallelism and was implemented by using java. We have collected 120 documents of management, biometrics and image process from the web as test data by using the customizable webcrawler websphinx and 20 documents of management, image process, biometrics (d1-d7, d8-d14, d15-20) were taken for computing the sample results. Stemming the terms and removing the stopwords are performed for each document and the documents are given to the map functions to compute the (key, value) pair for each document. The map function computes the (key, value) pair for each document and is given to all the reduce functions the binary classifiers each of which have distinct predefined label related keywords from any one of the predefined labels to predict a binary value 0/1. The binary classifiers equals the number of predefined labels computes the binary value 1/0 and the combiner the merge function combines the binary value from all binary classifiers for each document and only the positive labels are assigned to the documents. Table 1 shows the sample results of the proposed approach during the classification of labels to the documents and Table 2 gives the intermediate results of the proposed approach which assigns the set of positive labels to the documents. Table 3 gives the label cardinality LC and label density LD of the sample results of the proposed approach.

Label cardinality is the average number of labels per example

$$LC=1/n \sum_{i=1}^{n} |Y_i| \quad \text{and}$$

Label density is the label cardinality / number of labels

$$LD=LC / |L|$$

TABLE I. SAMPLE RESULTS OF CLASSIFYING LABELS TO THE DOCUMENTS

Documents	Binary classifiers			labels
	h1	h2	h3	l1,l2,l3
d1	1	1	0	{1,1,0}
d2	0	1	0	{0,1,0}
d3	0	1	0	{0,1,0}
d4	0	1	0	{0,1,0}
d5	1	1	0	{1,1,0}
d6	0	1	0	{0,1,0}
d7	0	1	1	{0,1,1}
d8	1	1	0	{1,1,0}
d9	1	1	0	{1,1,0}
d10	1	1	0	{1,1,0}
d11	1	1	0	{1,1,0}
d12	1	1	0	{1,1,0}
d13	1	1	0	{1,1,0}
d14	1	0	0	{1,0,0}
d15	1	1	1	{1,1,1}
d16	1	1	1	{1,1,1}
d17	1	1	1	{1,1,1}
d18	0	1	1	{0,1,1}
d19	1	1	1	{1,1,1}
d20	1	1	1	{1,1,1}

TABLE II. SAMPLE RESULTS OF POSITIVE LABELS OF THE DOCUMENTS

Documents	Positive labels
d1	{l1,l2}
d2	{l2}
d3	{l2}
d4	{l2}
d5	{l1,l2}
d6	{l2}
d7	{l2,l3}
d8	{l1, l2}
d9	{l1, l2}
d10	{l1, l2}
d11	{l1, l2}
d12	{l1, l2}
d13	{l1, l2}
d14	{l1}
d15	{l1,l2,l3}
d16	{l1,l2,l3}
d17	{l1,l2,l3}
d18	{l2,l3}
d19	{l1,l2,l3}
d20	{l1,l2,l3}

TABLE III. LABEL CARDINALITY AND LABEL DENSITY OF THE SAMPLE RESULTS

documents	LC	LD
management	1.43	0.48
imageprocess	1.86	0.62
biometrics	2.83	0.94

The proposed approach was evaluated on test data by using the label based evaluation metrics of precision and recall in which the evaluation task is decomposed into separate evaluations for each label [5] . To evaluate recall and precision for each label, L be the set of labels, Y_d be the set of true labels for example d and P_d be the set of predicted labels

from classifier h where H_d^l =1 if $l \varepsilon Y_d$ and $l \varepsilon P_d$, otherwise 0; P_d=1 if $l \varepsilon P_d$, otherwise 0; Y_d=1 if $l \varepsilon Y_d$, otherwise 0 and the label based recall, precision measure on data set D [28] is calculated and given in table 4.

$$\text{Precision}(l) = \frac{\Sigma_{d\varepsilon D} H_d^l}{\Sigma_{d\varepsilon D} P_d^l} \qquad \text{Recall}(l) = \frac{\Sigma_{d\varepsilon D} H_d^l}{\Sigma_{d\varepsilon D} Y_d^l}$$

TABLE IV. PRECISION AND RECALL OF THE PROPOSED APPROACH

Evaluation metrics	Management (l1)	Imageprocess (l2)	Biometrics (l3)
Precision	0.92	0.90	0.92
Recall	0.80	0.95	0.96

The results are plotted as chart in figure 5 and the chart shows the proposed approach is more accurate in classifying labels to the documents.

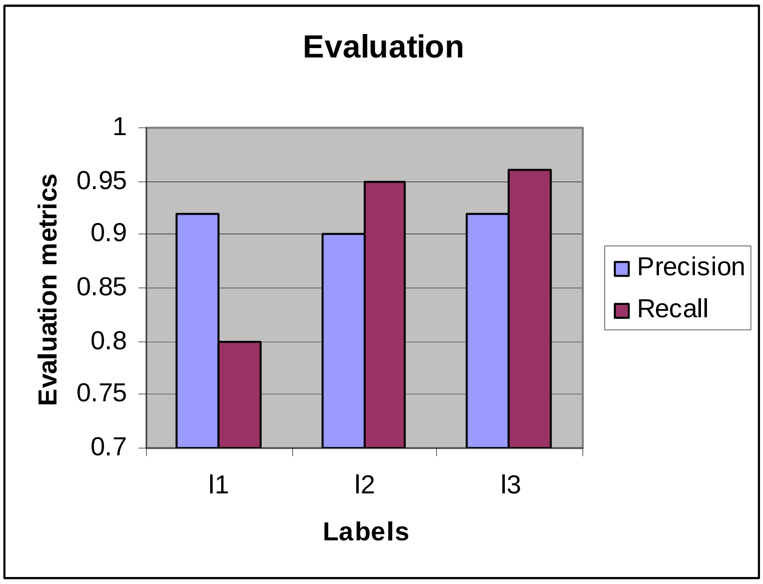


Fig. 5 Performance of the proposed approach

VI. CONCLUSION

Our proposed approach uses MapReduce framework for classifying labels to the documents of the web in which the binary classification is performed by binary classifiers. MapReduce framework the parallel programming model used in our proposed approach accurately classifies the positive labels of the documents using binary classifiers which equals the number of labels with less computation task.

REFERENCES

[1] Derrall Heath, Andrew Zitzelberger, Christophe G. Giraud-Carrier, *A Multiple Domain Comparison of Multi-label Classification Methods*, Working Notes of the 2nd International Workshop on Learning from Multi-Label Data, Haifa, Israel, 2010.

[2] Grigorios Tsoumakas, Ioannis Katakis , *Multi-Label Classification: AnOverview*, in International Jouranl of Data Warehousing and Mining, 2007.

[3] Mohammad S Sorower, *A Literature Survey on Algorithms for Multi-label Learning*, Department of Computer Science, Oregon State University, Corvallis, OR 97330, December 2010.

[4] Araken M Santos, Anne M P Canuto and Antonino Feitosa Neto , *A Comparative Analysis of Classification Methods to Multi-label Tasks in Different Application Domains*, International Journal of Computer Information Systems and Industrial Management Applications, Vol.3, pp. 218-227, 2011.

[5] Grigorios Tsoumakas, Ioannis Katakis and Ioannis Vlahavas, *Mining multilabel data*, in Maimon O, Rokach L (ed) Data mining and knowledge discovery handbook, Springer, pp. 667–685, 2010.

[6] Xiatian Zhang, Quan Yuan, Shiwan Zhao, Wei Fan, Wentao Zheng, Zhong Wangz, *Multilabel Classification without the Multi-label Cost*, in Proceedings of the 10th SIAM International Conference on Data Mining, 2010.

[7] Jesse Read, *A Pruned Problem Transformation Method for Multi-label Classification*, in Proceedings of the New Zealand computer science research student conference, Christchurch, New Zealand, pp. 143–150, 2008.

[8] Shantanu Godbole and Sunita Sarawagi, *Discriminative methods for multi-labeled Classification*, In PAKDD ’04: 8th Pacific- Asia Conference on Knowledge Discovery and Data Mining, pages 22–30. Springer, 2004.

[9] Min-Ling Zhang and Zhi-Hua Zhou. , *A k-nearest neighbor based algorithm for multi-Label classification*, in GnC ’05: IEEE International Conference on Granular Computing, pages 718–721. IEEE, 2005.

[10] Jesse Read, Bernhard Pfahringer, Geoff Holmes, Eibe Frank, *Classifier Chains for Multi-label Classification*, in Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pp. 254-269, 2009.

[11] Sang-Hyeun Park and Johannes F¨urnkranz, *Multi-Label Classification with Label Constraints*, International Journal of Computer Information Systems and Industrial Management Applications , Vol. 3, pp. 218-227, 2011.

[12] Shimin Chen, Steven W. Schlosser, *Map-Reduce Meets Wider Varieties of Applications*, Intel Research Pittsburgh Tech Report, IRP-TR-08-05, May, 2008.

[13] Ralf Lammel, *Google's MapReduce programming model — Revisited*, Science and Computer Programming, Vol 68, Issue 3, pp. 208-237, 2007

[14] Matei Zaharia, Andy Konwinski, Anthony D. Joseph, Randy Katz, Ion Stoica, *Improving MapReduce Performance in Heterogeneous Environments*, Technical Report No. UCB/EECS-2008-99.

[15] Bryan Catanzaro , Narayanan Sundaram and Kurt Keutzer, *A mapreduce framework for Programming graphics processors*, in 3rd Workshop on Software Tools for MultiCore Systems (STMCS), 2008.

[16] Jackson H.C. Yeung, C.C. Tsang1, K.H. Tsoi, Bill S.H. Kwan, Chris C.C. Cheung, Anthony P.C. Chan and Philip H.W. Leong, *Map-reduce as a Programming Model for Custom Computing Machines*, 16th International Symposium on Field-Programmable Custom Computing Machines, pp. 149 -159, 2008.

[17] Colby Ranger, Ramanan Raghuraman, Arun Penmetsa, Gary Bradski, Christos Kozyrakis, *Evaluating MapReduce for Multi-core and Multiprocessor Systems*, in Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture, pp.13-24, 2007.

[18] Bishan Yang, Jian-Tao Sun, Tengjiao Wang, Zheng Chen, *Effective Multi-Label Active Learning for Text Classification*, in the 15th ACM SIGKDD Conference On Knowledge Discovery and Data Mining, 2009.

[19] Akinori Fujino and Hideki Isozaki, *Multi-label Patent Classification at NTT Communication Science Laboratories* , in Proceedings of NTCIR-6 Workshop Meeting, May 15-18, 2007, Tokyo, Japan.

[20] Grigorios Tsoumakas, Ioannis Katakis and Ioannis Vlahavas, *Effective and Efficient Multilabel Classification in Domains with Large Number of Labels*, ECML/PKDD 2008 Work-shop on Mining Multidimensional Data, 2008.

[21] Tamer Elsayed, Jimmy Lin and Douglas W.Oard, *Pairwise Document Similarity in Large Collections with MapReduce,* in Proceedings of ACL-08, pp. 265-268, 2008.

[22] *Introduction to parallel programming and mapreduce*, http:/code.google.com/edu/parallel/mapreduce-tutorial. html.

[23] Hung-chih Yang, Ali Dasdan, Ruey-Lung Hsiao, D. Stott Parker, *Map-Reduce-Merge: Simplified Relational Data Processing on Large Clusters*, in Proceedings of the 2007 ACM SIGMOD international conference on Management of data , pp. 1029-1040, 2007.

[24] Nadia Ghamrawi , Andrew McCallum, *Collective Multi-Label Classification*, in Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pp. 195-200, New York, NY, USA, 2005.

[25] Gerardo Lastra, Oscar Luaces, Jose Ramon Quevedo, Antonio Bahamonde, *Graphical Feature Selection for Multilabel Classification Tasks*, in Proceedings of Intelligent Data Analysis, IDA 2011.

[26] Gautam Pant, Padmini Srinivasan, Filippo Menczer, *Crawling the Web*, Web Dynamics 2004, pp. 153 - 178.

[27] Lay-Ki Soon, Yee-Ern Ku , *Web Crawler with URL Signature – A Performance Study*, in 4th Conference on Data Mining and Optimization (DMO), Langkawi, Malaysia, 02-04 September 2012.

[28] Matthew R. Boutell, Jiebo Luo, Xipeng Shen, Christopher M.Brown, *Learning multi- labelscene classification*, Pattern Recognition, vol..37, no.9, pp. 1757-1771, 2004.