University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 4, Issue 1, January 2020

ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

An Effective Way to Enhance Classifications for the Semi-Structured Research Articles

Ejaz Ahmed, Sumbal Ashraf, Waseem Shahzad

FAST National University of Computer and Emerging Sciences, Pakistan
ejaz.ahmed@nu.edu.pk, sumbal.ashraf@yahoo.com, waseem.shahzad@nu.edu.pk

Abstract: Due to the drastic increase in the research publications, numerous research articles are available electronically on different online digital libraries. Some research articles or papers are not retrieved during online searches due to their classification issues. The adequately structured research articles are relatively easily approachable as compared to semi-structured and unstructured research articles, and sometimes the reader does not get accurate results on different digital libraries as the research articles are not classified properly. Neglecting the semi-structured and unstructured published research not only causes gap deficiency but also affects the results of the proposed techniques and citations for other articles. Usually, researchers missed semi-structured and unstructured research articles during their online search. Classification techniques have been applied to structured articles and no significant work has been performed towards the classification of semi-structured and unstructured research articles. Therefore, this research focuses on the classification of semi-structured research articles using different supervised classification techniques so that the most accurate and large amount of relevant research results will be achieved. For experimentation, a labeled dataset was used for the classification of semi-structured papers. The dataset we used for experimentation is comprised of manually gathered research articles from Santos repository dataset and labeling them accordingly. The current study used four different supervised classification techniques such as Support Vector Machine (SVM) classifier, Naïve Based classifier, K Nearest Neighbor classifier, and Decision Tree classifier. The comparison was performed between these supervised classification techniques to see which classifier gives better accuracy. The unit of measures or parameters selected to compare these classifiers are: accuracy, recall, precision, and f-score. The evaluation was performed on the basis of results and comparison in the experimentation. Experimental results of classifiers K-neighbor are better than other classifiers SVM, Decision Trees and Naive Bayes.

Keywords: Structured research articles, semi-structured research articles, classifiers, supervised technique, keywords, Santos, Gensim, TFID

I. INTRODUCTION

Research means mining new facts through a systematic way [1]. All types of research articles are mined for our related research topic and categorized in a systematic manner. Good and continuous research in a particular domain not only opens new opportunities for researchers but also matures the domain. Nowadays, extensive amount of research is being performed in almost every state-of-the-art domains such as Software Engineering [2, 3], Electrical Engineering [4, 5], Medical [6], Computer Science [7, 8], Art, Bioinformatics, Big Data [9, 10, 11] with the Data Mining of various disciplines such as Paleontology, Cancer Detection, Civil [12], Aerodynamics and Mechanics etc. Domain is normally well defined from the review of research articles. The research articles can be defined in terms of manuscript, document, research paper, conference paper, journal paper etc. Published research documents or articles are then made available on the internet for further study and are being reviewed by the researchers and students. Researchers can search these documents via search engines databases such as Google Scholar, IEEE explorer, ACM and other digital libraries [13]. Search engines answer the user's queries after classifying the indexed documents or articles on the basis of metadata elements viz. title, keywords, abstract, authors and matching of words from full text.

Some unstructured and semi-structured research articles are published and retrieved through search engines but do not contain metadata elements keywords, abstract or some other standard sections [7, 6, 11, 14]. Sample unstructured and semi-structured papers articles are enlisted in references [15, 16]. A large number of articles are generally structured so their classification becomes relatively much easier as compared to unstructured or semi-structured articles [11]. Whereas, semi-structure articles are those papers which have no proper format means articles do not have keywords, abstract is normally not explicitly well written and may have issues including missing pages and venues etc. [15]. Unstructured articles are those articles in which various sections are missing or written without any proper section heading. This does not give the clarity of the concept [16].

Keywords of research articles always give significant concept to the related document [10]. Keywords help to determine the research work in a particular domain such as data mining. The keywords for this domain can be text classification, face detection, pattern reorganization, and emotion detection. Firstly, missing keywords deflate the search results of the extracted papers. Secondly, due to the semi-structured papers either search engine exclude those papers while searching them or if found then those papers affect the result of their proposed techniques (especially in the text classification). This is due to the lack of proper classification for semi-structured research articles. If proper classification of those papers is managed, it will not only optimize the search but also help to distinguish and classify the right domain. Different parameters and techniques are used for classifying articles for example some have used single feature, described in [14], and some have used double features [9] but still have not achieved the better accuracy in their results due to deficiencies of semi-structured articles.

A tremendous work has been managed in categorizing of structured research articles but semi-structured research still needs plenty of contribution and attention. The studies have shown that 70% of research from academic labs is not fully described because of invariability of the dataset [2]. Dataset has billion of published research articles in their repositories containing semi-structured, structured and unstructured nature of articles. Therefore this research only emphasizes on the classification of semi-structured research articles as they have no proper abstract and keywords.

We are using “Santos” dataset and managing manually by extracting semi-structured research articles. We only used semi-structured papers to extract the keywords for our experiments. We further categorized those extracted keywords by assigning the labels instead of using domains for training the dataset. The problem encountered was extracting all keywords which are not mentioned in the research paper so, for that we have used four different ways. (1) Read complete text of an article and extract all possible keywords, (2) Calculate the weightage score of every word then create our own keywords repository and match the extracted keywords with them, (3) Create a vector of whole paper and according to the weightage score the whole paper and (4) Manually read the whole paper and label it with class.

Categorization of articles in domains is another issue that needs to be addressed and we are keep in mind that some research articles only belongs to one of the state-of-the-art domain. In our research work, we categorized the keywords in five state-of-the-art domains i.e. computer science, software engineering, medical, electrical and others. Our dataset is small for the purpose of experiment and adding up more domains may affect the results. For enhancing the accuracy and precision with better results we reduced the scope and decreased the categories up to two domains that are computer science and software engineering.

In our scenario, we used supervised learning technique and different classifiers to validate that which classifier performed well in our proposed approach. Multiple classification techniques were used by researchers for structured research articles on selective features [14]. We have also used multiple techniques for cleaning our data of semi-structured articles and extracting different features for example we used word to vector for converting the text into vectors, TF-IDF (Term Frequency and Inverse Document Frequency) [9] for providing the weight score frequency to each word, word tokenizer so that word is passed token wise, pos tag, stop words for removing grammatical words. Results are evaluated on the basis of metrics parameters of experimentation such as accuracy score, recall score, precision score and F1 score. We are focusing on semi-structured articles because researchers exclude those kinds of papers for achieving a better metric score of their proposed algorithms and approaches.

Similarly many research papers which are not in the proper format but still published in well-rated conferences. Researchers may need those semi-structured papers for their studies and research. When their proposed approaches or techniques especially in data science such as text, image classification affected by these semi-structured articles then facing issue of low accuracy score of their proposed algorithms. So, they have to exclude that semi-structured and unstructured research papers. If the proper classification of those research articles could be introduced then we can not only optimize the search but also distinguish and classify the domain which will be beneficial for researchers. In this paper, we will manage which of the supervised machine learning classifier techniques gives better result in classifying the semi-structured research papers? We will apply the supervised technique for semi-structured research articles and compares the result with the cluster of semi-structured research articles.

In this paper, background studies and related work of classifications of research articles are discussed in section 2. Section 3 presents framework of classification of our research work. Section 4 presents how our framework is implemented under experimental setup. We finally conclude by summarizing the main achievements and future work.

II. Related Work

Through this research, we could extract that what is the optimal way to classify the missing semi-structured research articles? In this section, we have summarized the related work as follows:

Guo et al., [2] proposed a framework by using a rule-based approach for extracting auto-metadata from the papers. Hua et al., [9] have used the technique for classifying the structures paper. They proposed the algorithm for classifying the document on the basis of multi-label and multi-classification. Restriction should be applied to word list so that unnecessary and meaningless words can be avoided. Lines repetition and other filters (like removing acronyms, special helping words, project names etc.) should be applied to narrow down the scope of classification. Woodson et al., [13] is also based on classification of papers based on Requirement Engineering papers that has been empirically evaluated. Zhu et al., [18] proposed a framework of matrix factorization used for content information and link structure classification through a semi-supervised learning technique. The reproducibility of an academic research paper is managed using requirement engineering paper [20].

III. FRAME WORK OF CLASSIFYING AND CLUSTERING OF SEMI AND UNSTRUCTURED ARTICLES

A. Problem Identification

Generally, a dataset contain multiple research articles as explained in figure 1. The research articles lie in multiple domains (such as computer science, electrical engineering, electrical and medical etc.) and are of different type (i.e. structured, semi-structured and non-structured). Every domain is distinguished by its defined keywords. When we categories one paper let say from title and keywords of the paper, it lies in multiple domains but could not classified properly. This problem occurs due to missing of keywords in semi-structured and unstructured research article. However, classifying structured article is way easier than the semi structured and unstructured articles, that's why we are only focusing on the semi structured articles. The semi-structured articles lack keywords, introduction, abstract, venues and can have improper format of citation but we are highly focusing on missing keywords research articles.

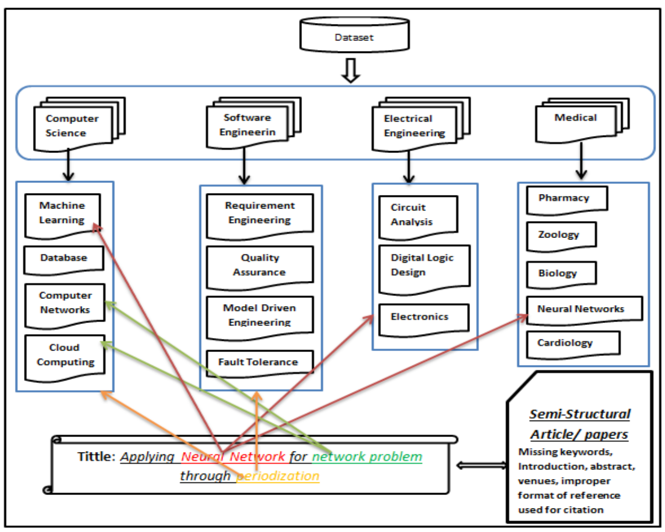

Figure 1. Problem identification mapping for semi-structured articles

B. Working Process of our Methodology

Keywords of each paper are loaded into vectors and those vectors are passed into the classifiers for calculating the accuracy, precision, recall, and f-score. For each paper, we have given two labels. One is computer science and other is software engineering for supervised classification techniques. In semi-structured research articles, we have concerned with the keywords as they are not defined in the manuscript so we match the extracted keywords (which are in the form of vector) from our defined repository keywords as shown in figure 2.

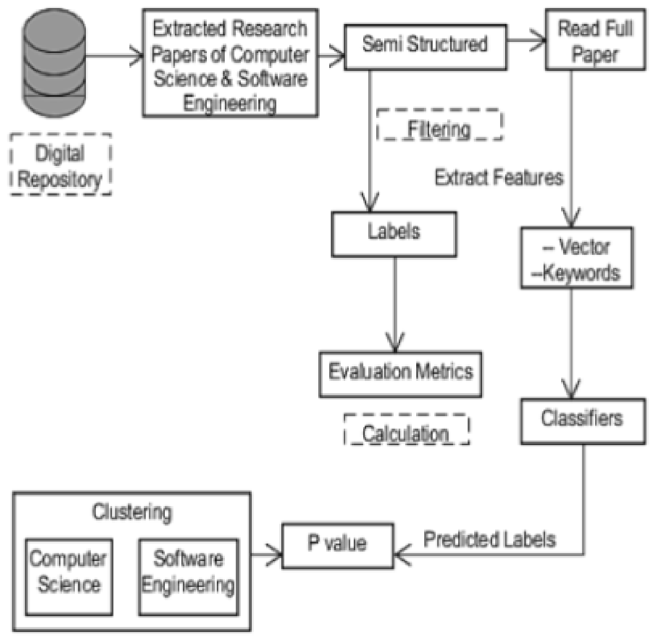

Figure 2. Working process of our methodology

First, we need to understand the working process of our methodology, explained in figure 3. We have digital repository named as "Santos dataset" which contains all types of structured, semi-structured and unstructured research articles. We focused on specific dataset of semi-structured known as Santos dataset. We filtered around 200 semi-structured articles with two domains computer science and software engineering and then manually downloaded the research papers for performing our experiment. For extracting keywords from the papers, we used different python libraries. Each paper keywords are loaded into string of words and then converted into vectors; those vectors are passed into classifiers for calculating the metrics value such as accuracy, precision, recall and f-score. The classifier is applied on semi-structured as well as on structured research articles for comparisons and verifications purposes.

C. Deep flow of the Process using clustering

On the basis of accuracy, we have allocated the papers in their particular category or domain. Accuracy of weight, frequency will be evaluated on different classifiers. Deep flow of the process is shown in figure 3.

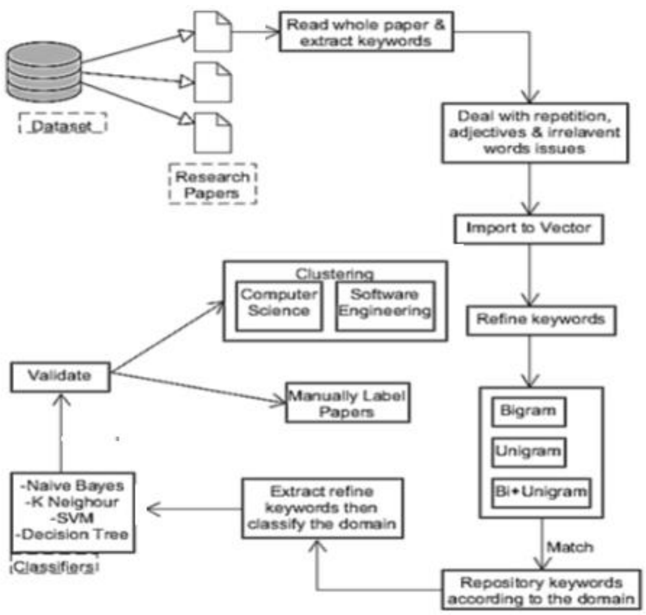

Figure 3. Deep flow of the Process using clustering

In the deep flow process, we explain the working technique for semi-structured articles in figure 3. First we remove the unnecessary words, repetition, adjectives and helping verbs, and finally create the list of keywords. We match the keywords through the process of unigram, bi gram and unigram + bigram techniques. The technique will further match the keywords from our repository keywords which are labeled domain wise. Classifiers use those labels and run on their trained model. Through classifiers, we came to know how efficiently the paper is classified. Further validation is managed by two ways: One is through cluster of two domains and second verifies our manual labeled data. The cluster is used on both structured and semi-structured research articles.

D. Equations

As the semi-structured papers have the issues of section heads and keywords so we need to read the whole paper through PDFPy for extracting the appropriate keywords. Preprocessing of data is performed in which repletion, adjective and irrelevant words issues are resolved through different techniques such as stop words, lemmatize, POS tag etc. These refined keywords of a paper are then imported in a vector so that all papers are stored in the vectors for matching the keywords. Refining the keywords in a proper way, we follow the equation from paper [19] given below

$$\text{Trigram} < \text{Bigram} < \text{Unigram} < \text{Unigram} + \text{Bigram} < \text{Unigram} + \text{Bigram} + \text{Trigram} \quad (1)$$

We only focus on initial part of above equation that is: Bigram < Unigram < Unigram+Bigram. In this way, the most accurate and one and two words keywords are extracted and then we matches these keywords from our repository keywords. Domains are classified on the basis of extracted and matched keywords from the research papers (labeled in the dataset) as shown in the figure 4.

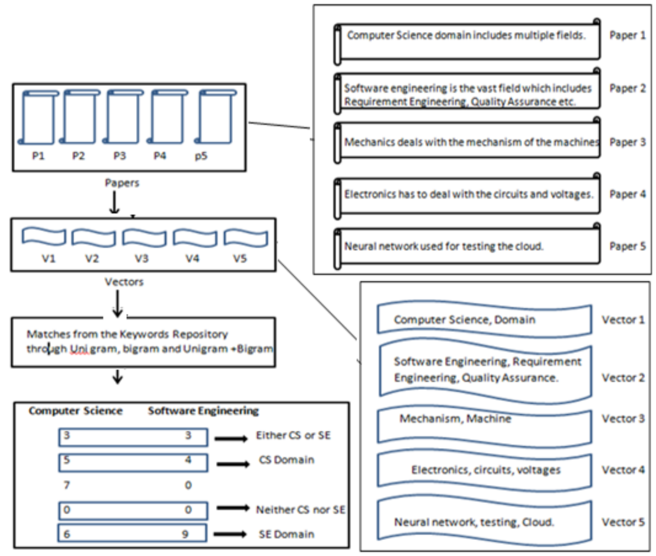

Figure 4. Domain Classification through string vector

E. Keywords Extraction

Classifiers are applied to check whether the classifier gives better results on our technique. The classifier's results are compared at the end on different datasets. Python library Scikit-Learn is used for the extraction of keywords from labels; selected result is shown in the figure 5.

```
n [194]: str1=dataExcel['text_final'][0]
         str_test=clean(str1)
         predict(str_test)


         Matched Words in CS:

         structure
         physical
         machine
         machine
         hardware
         table
         physical


         Matched Words in SE:

         negative
         condition
         branch
         condition
         design
         condition
         Matching in Computer Science= 7
         Matching in Software Eng= 6
```

Figure 5. Keyword extraction matches with labels

Validation is managed through two ways: One is the manual and second is through clustering. In a cluster, we define the two domains Computer Science and Software Engineering then matched the result data which was extracted on the basis of trained data. The outlook of the clustering is shown figure 6. The probability value is calculated in a cluster.

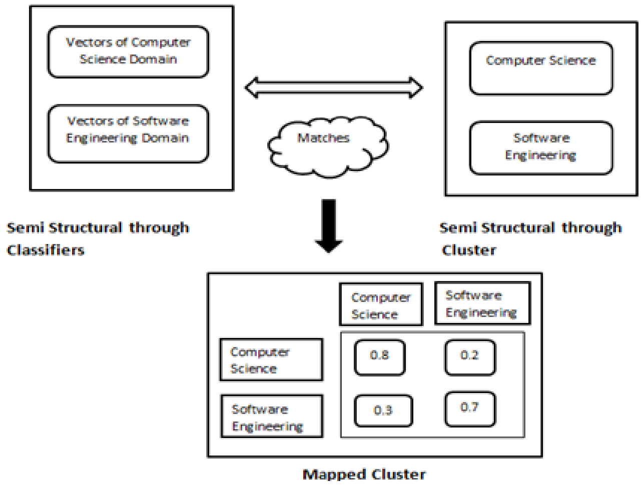

Figure 6. Clustering for Semi-structured Articles or Papers

IV. Experimental Setup

In this research work, we are using digital repository Santos dataset. This repository contains all types of structured, semi-structured and unstructured research articles. We have used different python libraries for extracting keywords from these research papers and mapped with our own keywords repository. We have used multiple techniques such as TF-IDF, Gensim, nltx, textract and pyPDF2 etc., to clean up the data and extract the variables from the PDF. The model is used for evaluating the techniques applied on different classifiers. The evaluation of the experiment is based on the metrics parameters which are defined in section 3. We have used Scikit-Learn, built in libraries for the implementation of classifiers. 70% of the data is used for training the model and 30% of the data is used for testing the model. The comparison is performed on different four classifiers for measuring the efficiency, accuracy and presentence of the classifiers.

We followed the steps containing data preprocessing and cleansing, training, model creation and experimentation. In data preprocessing and cleansing with training part, we create spreadsheet file by putting keywords extracted from each PDF. We used numpy and pandas libraries for this purpose and also used pyPDF2, nltk, stop words, lemmatizer and texttract. We trained it for matching the extracted keywords from the repository keywords file by assigning the labels and distinguishing the category in which the paper lies through test data. Model creation emphasizes on the integration of categorizing phase and the extraction phase. While extracting keywords we faced the issue of unstructured manuscript then we have to exclude those papers. In our model, we used different techniques for getting the accuracy of the results like we have used TF-IDF and Gensim for extracting the results. In experimentation, we used different classifiers to detect the accuracy. We have used two types of variables: independent variables are keywords and dependent variables are semi-structured research articles. This means change in independent variables keywords reflects the direct impact over the research articles which are dependent variables.

We executed our process on different state-of-the-art four classifiers and compared their results with efficiency on the basis of metrics parameters. Keep in mind that as we increased the dataset in the form of number of articles and their score varied and we performed experiment on computers with different hardware specifications. First, we calculate the score with respect to classifiers and at the end we calculate it relatively. Each keyword in a string vector of label has its own specific TF-IDF a numeric identifier. Numeric identifiers are used in the classification by classifiers and clustering. As a sample, we have shown the readings for K-neighbor Tree classifier in figure 7.

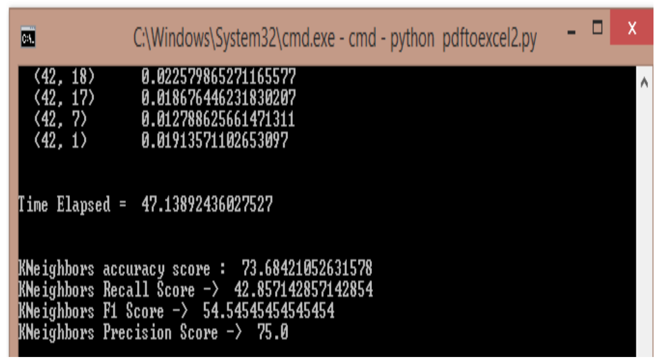

Figure 7. k-neighbour's Result

A. *Metrics Score*

Our metric parameters include Accuracy indicates how accurately proposed technique maps on the scenario and how accurately the results produced because it shows the ratio of correct prediction and input samples. Precision shows only the percentage of results which are relevant. Recall shows percentage of all the results. The f-score shows the harmonic average of the recall and precision. We calculated the score for all state-of-the-art classifiers as shown in table 1.

TABLE I. Classifier's Metric Score on Different Datasets

Dataset	**Metric Parameters**	**Classifiers**			
		SVM	**Naïve Bayes**	**K-neighbor**	**Decision Tree**
50 Papers	**Accuracy Score**	60.00%	73.33%	50%	68.42%
	Recall Score	53.33%	73.33%	56.36%	66.66%
	f-score	74.99%	84.61%	70.00%	70.58%
	Precision Score	100%	100%	66.66%	75%
100 Paper	**Accuracy Score**	56.25%	53.00%	62%	59.00%
	Recall Score	61.53%	44.44%	50.00%	76.90%
	f-score	53.33%	42.00%	70.00%	50.00%
	Precision Score	47.05%	60.00%	64.28%	60%
150 Papers	**Accuracy Score**	53.33%	46.66%	60%	53.33%

	Recall Score	53.84%	50.00%	58.00%	50.00%
	f-score	44.44%	47.82%	42.00%	48.78%
	Precision Score	40.00%	48.00%	57.00%	48%
200 Papers	**Accuracy Score**	55.55%	51.00%	74%	57.40%
	Recall Score	33.33%	42.84%	68.18%	46.51%
	f-score	28.57%	46.15%	60.00%	50%
	Precision Score	60%	50.00%	78.94%	43.47%

B. Classifiers Time Elapse

Time elapse is also calculated with respect to every classifier to check the efficiency and time consumption. It shows the time duration taken by each specific classifier against dataset. We set the time elapsed in each classifiers and their readings are shown in table 2.

TABLE II. TIME ELAPSE OF CLASSIFIERS ON DATASETS. (TIME IN SEC)

Classifiers/ Dataset	SVM	Naïve Bayes	K-neighbor	Decision Tree
50 papers	6.58	6.76	6.68	6.78
100 papers	10.87	10.93	11.06	10.90
150 papers	14.25	14.26	13.93	14.08
200 papers	15.63	15.57	15.91	15.65

C. Clusters Performance Result

As we have mentioned that dataset is the dependent variable so when we change the size of the dataset their results also vary as shown in the table 3. The two clusters are used for comparing the results of the classifiers. In clusters, we cannot mention the names so we assigned “0” for “Computer Science” and “1” for Software Engineering”.

TABLE III. CLUSTER RESULTS ON DIFFERENT CLASSIFIERS WITH PERFORMANCE

Cluster Scores		
Classifiers	**Score 1**	**Score 2**
SVM	0.3777	0.4444
Naïve Bayes	0.3703	0.6111
K- Neighbor	0.3888	0.3703
Decision Tree	0.574	0.42599

D. Classification Score

All the classifier’s results are now in the one graph. It also portrays that which classifier performed better under different parameters of selection of papers. It will also help us in future if we want to propose any new or altered classifier then we should work in which specific parameters to enhance the efficiency and performance of the proposed technique. Figure 8 shows the comparison of different performance measures on the different number of papers and the percentage of performance on different classifiers.

Classifier K-neighbor shows distinct behavior as compared to other classifiers when papers are 200. Classifiers SVM and Naive Bayes are showing 100% performance when papers are 50.

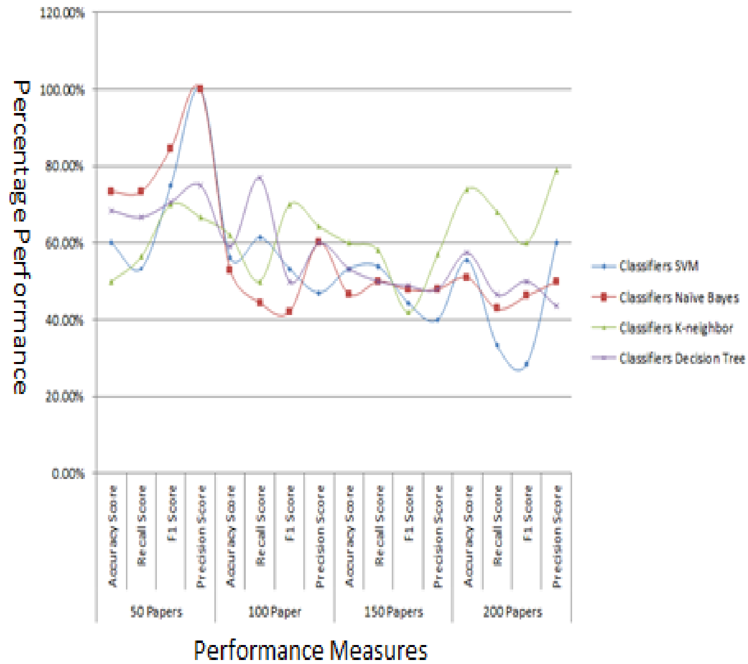

Figure 8. Classifiers score result of selected Datasets

V. CONCLUSION & FUTURE WORK

Usually structured research articles contain different features and classifications. However, there are published research articles with missing such proper features. In this paper, we had targeted those research papers with missing features commonly known as semi-structural research articles. We have gathered a data set of around 200 semi-structured research papers which is extracted from different digital libraries. Different python libraries were used to extract keywords from research articles. We then applied different supervised machine learning techniques on our dataset to see which technique gives the best results. We ran the test on different machine using different data sizes ranging from 50 to 200. In term of accuracy, K-neighbor gave the best result. SVM gave best result in term of precision on 50 research articles but the precision value drastically went down as we kept on increasing the research articles.

In future, we are looking forward to enhance the technique by increasing the dataset and run the cluster on huge data for validating techniques on the larger scale. Furthermore, we will include other classifiers and improve the approach by further working on optimizing the results. We have limitation for example few unstructured articles are not readable properly by the machine so we had to remove them manually. Even though clustering gives right and accurate results by validating the classifiers. The results are not showing clear prediction when dataset is exceeded. Results were changed on different computer systems with distinct specifications.

References

[1] Pan, Chenguang, and Wenxin Li. "Research paper recommendation with topic analysis." In 2010 International Conference on Computer Design and Applications, vol. 4, pp. V4-264. IEEE, 2010.

[2] Woodson, Clinton, Jane Huffman Hayes, and Sarah Griffioen. "Towards reproducible research: automatic classification of empirical requirements engineering papers." In Proceedings of the ACMSE 2018 Conference, p. 8. ACM, 2018.

[3] Li, Tao, Chengliang Zhang, and Shenghuo Zhu. "Empirical studies on multi-label classification." In 2006 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06), pp. 86-92. IEEE, 2006.

[4] Shikano, Hiroaki, Jun Shirako, Yasutaka Wada, Keiji Kimura, and Hironori Kasahara. "Power-aware compiler controllable chip multiprocessor." In 16th International Conference on Parallel Architecture and Compilation Techniques (PACT 2007), pp. 427-427. IEEE, 2007.

[5] Juvaste, Simo. "Studying Asynchronous Shared Memory Computations." In 16th International Conference on Parallel Architecture and Compilation Techniques (PACT 2007), pp. 413-413. IEEE, 2007.

[6] Bui, Duy Duc An, Guilherme Del Fiol, and Siddhartha Jonnalagadda. "PDF text classification to leverage information extraction from publication reports." Journal of biomedical informatics 61 (2016): 141-148.

[7] Ahmad, Shahbaz, and Muhammad Tanveer Afzal. "Combining Co-citation and Metadata for Recommending More Related Papers." In Frontiers of Information Technology (FIT), 2017 International Conference on, pp. 218-222. IEEE, 2017.

[8] Damez, Lionel, and Jean Pierre Derutin. "Fast prototyping of complex Signal and Image Processing applications on SoC using homogenous network of communicating processors." In 16th International Conference on Parallel Architecture and Compilation Techniques (PACT 2007), pp. 404-404. IEEE, 2007.

[9] Ali, Tariq, N. A. Sajid, S. Asghar, and M. Ahmed. "Classification of scientific publications using swarm intelligence." Proceedings of the Pakistan Academy of Sciences 50, no. 2 (2013): 115-126.

[10] Hua, Liu. "Research on Multi-classification and Multi-label in Text Categorization." In Intelligent Human-Machine Systems and Cybernetics, 2009. IHMSC'09. International Conference on, vol. 2, pp. 86-89. IEEE, 2009.

[11] Guo, Zhixin, and Hai Jin. "Reference metadata extraction from scientific papers." In Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2011 12th International Conference on, pp. 45-49. IEEE, 2011

[12] Zhou, Peng, and Nora El-Gohary. "Ontology-based multilabel text classification of construction regulatory documents." Journal of Computing in Civil Engineering 30, no. 4 (2015): 04015058.

[13] Sajid, Naseer Ahmed, Tariq Ali, Muhammad Tanvir Afzal, Munir Ahmad, and Muhammad Abdul Qadir. "Exploiting reference section to classify paper's topics." In Proceedings of the International Conference on Management of Emergent Digital EcoSystems, pp. 220-225. ACM, 2011.

[14] Choi, Wonjun, Jayhoon Kim, Soonyoung Kim, and Jeonghwan Kim. "A study of reference metadata classification with deep learning." In Information and Communication Technology Convergence (ICTC), 2017 International Conference on, pp. 144-146. IEEE, 2017.

[15] Romanescu, Bogdan F., Michael E. Bauer, Daniel J. Sorin, and Sule Ozev. "Reducing the impact of process variability with prefetching and criticality-based resource allocation." In 16th International Conference on Parallel Architecture and Compilation Techniques (PACT 2007), pp. 424-424. IEEE, 2007.

[16] Shikano, Hiroaki, Jun Shirako, Yasutaka Wada, Keiji Kimura, and Hironori Kasahara. "Power-aware compiler controllable chip multiprocessor." In 16th International Conference on Parallel Architecture and Compilation Techniques (PACT 2007), pp. 427-427. IEEE, 2007.

[17] Zhu, Shenghuo, Kai Yu, Yun Chi, and Yihong Gong. "Combining content and link for classification using matrix factorization." In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 487-494. ACM, 2007.

[18] Rastin, Niloofar, Mansoor Zolghadri Jahromi, and Mohammad Taheri. "Multi-label classification systems by the use of supervised clustering." In Artificial Intelligence and Signal Processing Conference (AISP), 2017, pp. 246-249. IEEE, 2017.

[19] Zhou, Peng, and Nora El-Gohary. "Ontology-based multi label text classification of construction regulatory documents." *Journal of Computing in Civil Engineering* 30, no. 4 (2015): 04015058.

[20] Clinton Woodson, Jane Huffman Hayes, Sarah Griffioen, Towards Reproducible Research: Automatic Classification of Empirical Requirements Engineering Papers, ACM SE '18, , Richmond, KY, USA, March 29–31, 2018