Automatic Classification of research grants proposals using a multi– class machine learning model

[1] Rebecca Lupyani
University of Zambia
Department of Computer Science
Lusaka, Zambia
rebecca.lupyani@cs.unza.zm

[2] Jackson Phiri
University of Zambia
Department of Computer Science
Lusaka, Zambia
jackson.phir@cs.unza.zm

***Abstract*— Research and Development has become fundamental to the economic development of every nation. Many countries have established institutions to promote, support and fund research and innovation in societies. These institutions seek to monitor and keep track of how much money they are willing to or have been spending on research and development activities in specific fields or topics. Funding investment decisions are based on whether proposed research ideas fall under disciplines of interest to national development. It is therefore imperative that research proposal documents submitted for funding consideration are classified according to respective disciplines. This paper explores the adaptation of the text classifier Support Vector Machine (SVM) for multi-classification and use it to automatically classify scholarly research documents and predict eligibility of funding. The experiment results demonstrate that the SVM model performed well with an accuracy performance of 89%. The study recommends implementing Application Programming Interface (API) endpoints for the model, to facilitate its integration with third-party tools and services to automatically classify the research proposal documents and award research grants.**

Keywords— text classification, machine learning, research grant, Multi-classifier, Support Vector Machine

I. INTRODUCTION

Research has become an integral part of any country's economic development. It provides the building block upon which societal growth and advancement is hinged. It is a tool for building knowledge, and for facilitating learning. Thus, many countries have established institutions responsible for promoting societies where research and innovation can be created, used and shared by planning and supporting multi-disciplinary research projects, recommend new directions for research thereby resulting in a society with a better research culture and thus contribute to national development. These institutions responsible for research and development are interested in monitoring and keeping track of funds invested in research and development activities in specific fields or topics [1]. Institutions such as the National Institute of Scientific and Industrial Research (NISIR) in Zambia focus on facilitating, handling, and funding multiple research projects in the education domain in order to offer support for technical and vocational skills development and application of science, technology and innovation through research and development [2]. The institution seeks to award research grants in Higher Education Institutions (HEI) according to specific areas of discipline namely technology, science and communications. Additionally, the institution seeks to monitor and keep track of its investments in the specific research areas and to keep accurate records to recall or prove research projects that have been undertaken under its mandate. The funding institutions seek to determine how much funding has been invested in a particular field and also to decide whether or not submitted research proposals by grant applicants are eligible for funding [2]. One common way in which institutions handle or classify research proposal documents to be considered for funding aid is by using tagging or keyword based systems in which human indexers tag the documents and assign them to predefined categories. This allows users to specify the category to which the research documents belongs upon submission of the document [3]. Even though these systems are effective, they suffer from one or more weaknesses such as inconsistent or incomplete category assignments [4]. These weaknesses have led to the emergence of text categorisation using machine learning techniques to automate the assignment of natural language texts to predefined categories based on their content [4]. Text categorisation is considered a decision-making criteria which is very useful in analysing content and it works well if a particular piece of text piece belongs to a specific prescribed category or class [5]. Instead of individuals manually classifying documents, machine learning algorithms can be used to automatically classify unseen documents and generate their respective subject classes on the basis of human-labelled training documents [6].

This paper explores how machine-learning techniques specifically the supervised learning model, Support Vector Machine (SVM) can be adapted for multi- classification to automatically classify research documents according to specific disciplines and to determine the eligibility of the research topic for awarding of research grants.

II. RELATED WORK

This section describes the characteristics of several algorithms that have emerged in handling text classification tasks as well as their application to real world problems.

A. Text Classification Algorithms

Research interest in text categorisation has grown in popularity in the recent years. It has been explored in the applications of information processing tasks, information retrieval, content filtering and machine learning tasks particularly because it facilitates the decision making criteria [4]. Several algorithms have been applied to the problem of text categorisation. Some of the most popular classifiers that have emerged over the years are Naïve Bayes (NB), K-Nearest Neighbor (KNN), and Support vector machine (SVM) which are considered statistics-based or traditional methods [7] [8]. NB is the first model to be applied to text classification problems [9] and later came KNN [10], RF and SVM [11]. These classifiers have been applied to different text classification problems. According to a study by Li et al [7], the NB is a model based on the Bayes' theorem and is the simplest and most broadly used model. The Bayes' theorem is a probability theory which is used to make predictions or decisions by making a strong assumption that the presence of one feature is independent of the presence of other features [12]. The Naive Bayes classification model is particularly useful for problems with discrete features. There are several types of Naïve Bayes models; Bernoulli NB, Gaussian NB, Bayesian NB classifier and Multinomial NB. [6]. Each of the models makes different assumptions about the features and are specific to the type of data. Hence the performance and application of each model is associated with the nature of the dataset and the characteristics of the features in the problem at hand [6]. A study carried by Xu [6] revealed that Multinomial NB performs slightly better than Bernoulli NB on few labeled datasets. The Bernoulli NB is an extension of the Multinomial NB but it is specifically designed for binary features. In another study Singh et al [13] concluded that the Bayesian NB classifier and the Gaussian event model is more superior to the Multinomial NB when subjected to 20 Newsgroups. The Naïve Bayes models (binary or categorical) are said to be most suitable in text classification tasks such as fake news detection [7], spam filtering and sentiment analysis [14]. The other popular classifier is the K-Nearest Neighbour which is a simple machine learning algorithm suitable for regression and classification tasks [15]. It makes predictions based on the majority class of its K- nearest data points. In other words it classifies an unlabeled data sample by finding the category with most samples on the k nearest samples [7]. The 'K' refers to the number of nearest neighbors to consider when making predictions for a new data point. A data point refers to the data subject for which we want to predict the class label [15]. The KNN model can be used for classification problems such as document classification, sentiment analysis, spam detection and language detection to mention but a few. It is however important to note that even though it can handle such problems, it may not be the most efficient option for all text classification problems as it works well on smaller datasets, which need to be properly preprocessed and represented in a suitable vector space [16]. Additionally, there is need to properly hyper-tune the K value for the model to exhibit a good performance. Another popular classifier in the field of text classification is the Support Vector Machine (SVM). It is popular for data that is highly dimensional and tasks that have complex decision boundaries. SVM as a text classifier represents each text as a vector. It uses the concept of a hyperplane where the SVM constructs an optimal hyperplane in a one-dimensional feature space that maximizes the margin between different classes [17]. The margin is defined as the distance between the hyperplane and the nearest data points from each class. Maximising this margin helps the SVM to achieve the best possible generalization to unseen data. This quality of the SVM has made it useful to many text classification and regression tasks [18]. From the literature above, it is evident that the SVM is one of the most popular text classifiers as it exhibits a great accuracy and precision when handling text classification or categorisation tasks [19] and thus it will applied in this study. The SVM can be used for both binary and multi-class text classification tasks even though it is originally designed for binary classification tasks. In binary text classification, text documents are classified into one of two classes whereas in multi-class text classification, text documents are assigned to one of multiple classes [20]. Both approaches are popular although some studies have revealed that SVM exhibits a competitive performance in multi-class classification tasks as opposed to binary classification tasks [20]. In this study, an SVM multi-classifier will be used. Having looked at some of the popular text classifiers that have been applied to text classification tasks, this study will further review how these text classifiers have been applied to document classification problems.

B. Application of text classifiers to real world problems

Lam et al [21] carried out a research to determine the effectiveness of automatic text categorisation on text retrieval. They developed an approach that was derived from a combination of a learning paradigm known as instance-based learning and an advanced document retrieval technique known as retrieval feedback. They experimented the effectiveness of this model on two real world document collections extracted from the Medline database namely the HERSH and the OHSUMED. Their findings revealed that automatic categorization of documents improves the text retrieval quality compared to using manual categorization [21]. Phiri [22] also applied the concept of automatic classification of digital objects to improve Metadata Quality of Electronic Theses and Dissertations in Institutional Repositories. The study evaluated the performance of three classification models namely, the Multinomial Naive Bayes, Random Forest (RF) and Stochastic Gradient Descent (SGD) [22]. The datasets used for conducting experiments were prepared using data harvested from the University of Zambia Institutional Repository (UNZA-IR) and from the annual MEDLINE/PubMed citations [23]. The results exhibited a highest performance accuracy from the SGD text classifier. Thus, it was concluded that automatic classification using the supervised text classifiers applied in their study have the possibility of reducing the errors that result during the preparation of metadata.

In another study, Khor, K. A., Ko, Giovanni and Theseira [24], Walter carried a study to evaluate research grant programs using machine learning. They compared the performance of three machine learning classification models, multinomial Naïve Bayes (MNB), Multinomial Logistic Regression (MLR) and Support Vector Machines (SVM) to classify the research proposals according to the research funding structure used by the European Research Council (ERC). The results revealed that the SVM model exhibited a

better performance compared to the MNB and MLR in classifying the research proposals. Based on the results, the study was able to determine which funding programme was more successful than the other in a particular research discipline. The concept in this study is similar to the one that will be applied in this study, the difference is that while text classification using supervised machine learning was used to help evaluate the performance of funding programs based on the particular research discipline invested in, our study intends to use text classification using supervised machine learning techniques for the purpose of awarding research grants. Thus, this paper explores the use of the SVM Multi-classifier to facilitate the awarding of research grants by classifying research proposals according to the research discipline.

III. MATERIALS AND METHODS

This section discusses the methodology and materials used to develop the model as well.

A. Methodology

The study adopted the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology [25]. It is a common model that provides a structural approach for handling data mining and machine learning projects. The six stages of the CRISP- DM model were utilised as follows:

i. Business Understanding:
In this stage, the study's objectives and requirements were defined by understanding the process that applicants use when applying for research grants, how funding institutions process the research grant applications and lastly the challenges that the funding institutions face when processing applications. This step was taken to ensure that the study's objectives are aligned to the business perspective of the funding institutions.

ii. Data Understanding:
During this stage, data sources were explored to understand the available data and assess its quality and relevance to the study. The data was then extracted and examined to identify patterns and trends that would be relevant for the modeling phase.

iii. Data Preparation:
Under this stage the data extracted was preprocessed which is a method used to clean the data and prepare it for modeling. Cleaning of data involves removal of stopwords (unnecessary words), null and missing values, outliers and transforming it into formats required for the modeling. This stage also involves the selection of the necessary features that will be used as inputs and selecting relevant features for modeling. The preparation of data is a critical step to the performance of the model as the quality of the data directly impacts the performance of the models.

iv. Modeling:
In this phase, the SVM model was applied to the preprocessed data and its performance tested. This is process is known as training the model. The SVM model does not support multi-classification and so it had to be adapted by using the One to One Multi-classification approach.

v. Evaluation
Under this stage, the trained model was evaluated using different evaluation measures to obtain results such as accuracy, precision, recall and F1-score.

vi. Deployment:
In order to facilitate the deployment and integration of the classification model into other platforms and third party tools, Application Programming Interface (API) endpoints were implemented.

B. Materials

The materials that were used for this study are discussed below:

***Library*:** The library that was used to handle the machine learning tasks is PHP-ML.

***Text Classifier*:** The Support Vector Machine (SVM) Multi-classifier was selected as the model for text classification. Figure 1 shows the code used to create the model.

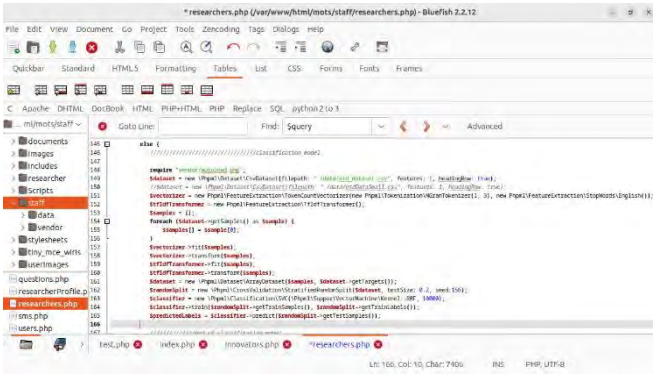

Figure 1: Classification model (SVM) code snippet

Dataset: The dataset used was constructed from the data that was extracted from the University of Zambia Institutional Repository (UNZA-IR). 6997 records were extracted which include details about research documents namely the author name, title of the research, abstract, publisher date, publication type, to mention but a few. The title and the subject area were used for the purpose of this study.

Figure 2: Data Extract from the UNZA -IR

IV. EXPERIMENTS

The paper seeks to use the SVM Multi-classifier for research document classification. Thus, the SVM model was trained using the UNZA-IR which was split into 80% training set and 20% testing set. The training set was used to train the model and the testing set was used to test the performance of the

model to make predictions of classification labels. The first experiment was carried by using the Title as the feature, the Abstract as the feature and lastly by using both the Title and Abstract together. The performance of the model was determined by calculating the accuracy, the precision the recall and the F1 Score. The experiment to predict a sample of a research title was carried and results obtained are shown in table 1.

Table 1: Results

Subject Class	Precision	Recall	F1 Score
Business	0.91	0.91	0.91
Education	0.80	0.76	0.78
Engineering	0.81	0.95	0.88
Medicine	0.91	0.96	0.93
Sciences	0.94	0.93	0.94
Social Science	0.85	0.73	0.79
Predicted Category		*Science*	

The performance metrics are explained below:
Precision = True Positive/ (True Positives +False Positives)
Recall=True Positive/ (True Positives +False Negatives)
Accuracy= (True Positives+ True Negatives)/ (True Positives +True Negatives+ False Positives +False Negatives)
[26]
Where:
True Positive = Actual class is positive and is predicted as positive
False Negative= Actual class is positive but is predicted as negative
True Negative= Actual class is negative and is predicted as negative
False Positive= Actual class is negative but is predicted as positive [26]
Additionally, the F1 score was calculated as it provides a more realistic measure of the model's performance and it is calculated as the weighted mean of the precision and recall.

F1 Score =2 ((precision +recall)/ (Precision * recall)) [26]
A sample subject

V. RESULTS AND DISCUSSION

This section discusses the results as obtained from the experiment carried out. The precision, recall and F1 Score results of the model's prediction on a sample title are expressed as a graph in figure 3 and the average performance of the model are expressed in table 2.

SVM + Title

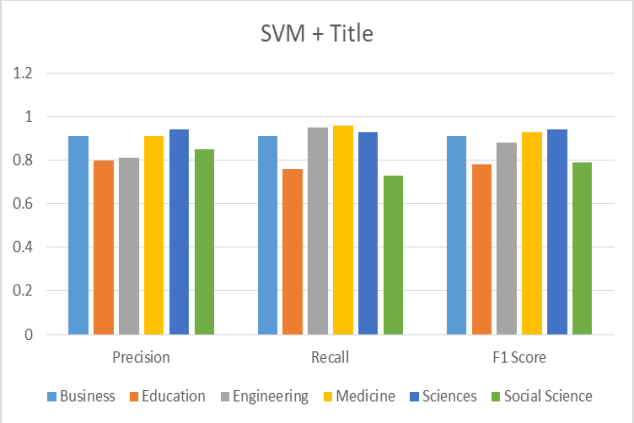

Figure 3: Results of SVM on sample title

The SVM Multi classifier works by calculating the probability of a subject title belonging to each of the subject classes. The subject class that gives the highest score is selected as the predicted subject class. It can be seen from figure 3 that the science subject class exhibits the highest scores and thus is selected as the predicted category. Table 2 shows a report on the general performance of the model when subjected to the UNZA-IR dataset. The model displays an accuracy performance of 0.89 and F1 score of 0.87.

Table 2: Average Performance SVM

SVM Multi- Classifier	
Accuracy	0.89
precision	0.86
recall	0.87
f1score	0.87

The results in table 2 indicate that the SVM classifier performs generally well in text classification tasks and thus can be applied to the problem of granting of research grant awards.

VI. CONCLUSION

The model exhibits a good performance in classifying the research titles according to disciplines. This makes it suitable to be used in the granting of research funds as the primary objective of funding institutions is to grant funds for research projects that address problems under disciplines of interest to national development. It also helps to avoid mistakes such as incorrect tag labelling that may occur when applicants manually tag their research titles upon submission of their application. The use of a machine learning model eliminates such mistakes by automatically categorizing the research titles according to discipline or field.

VII. LIMITATIONS

In addition to the research title and abstract, funding institutions may consider other factors such as age, gender, prior qualifications and affiliated educational institution, in their granting of research funds. These factors were not incorporated in the training data set due to challenges in obtaining an adequate dataset that comprises all such factors. Thus, the model maybe biased to making recommendations that are based on the submitted research topic only.

VIII. RECOMMENDATIONS

The study concludes that the SVM multi- classifier using the title as the feature exhibited a good performance. The study therefore recommends implementing an Application Programming Interface (API) for the model, to facilitate its integration with third-party tools and services that automatically classify research proposal documents and award research grants.

IX. FUTURE WORK

Based on the findings and limitations of the dataset used, for future work, the study recommends incorporating other input features such as the abstract of the research, the age, affiliation and prior qualifications of the applicant into the training dataset. This will help reduce the bias and make the model more suitable for research grant funding.

REFERENCES

[1] I. R. Matthew L Wallace, "Research portfolio analysis in science policy: moving from financial returns to social benefits," *Minerva,* vol. 53, no. 2, 2015.

[2] MOTS, "Ministry of Technology and Science," [Online]. Available: https://www.mots.gov.zm/?page_id=548. [Accessed 26 November 2022].

[3] S. R. ,. C. J. P. P. P. Lewis David D, "Training algorithms for linear text classifiers," in *ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996.

[4] Y. Y. T. G. R. F. L. David D. Lewis, "RCV1: A New Benchmark Collection for Text Categorization Research," *Journal of Machine Learning Research,* vol. 5, 2004.

[5] W. A. G. David D Lewis, "A sequential algorithm for training text," in *international ACM SIGIR conference on Research and development in information retrieval*, 1995.

[6] S. Xu, "Bayesian Naïve Bayes classifiers to text classification," *Journal of Information Science ,* pp. 48-49, 2018.

[7] Q. H. P. J. L. C. X. R. Y. L. S. P. S. Y. L. H. Li, "A survey on text classification: From traditional to deep learning," *ACM Transactions on Intelligent Systems and Technology (TIST),* vol. 13, no. 2, pp. 1-41, 2022.

[8] V. G. Nidhi, "Recent Trends in Text Classification Techniques," *International Journal of Computer Applications,* vol. 35, no. 6, 2011.

[9] M. E. Maron, "Automatic indexing: An experimental inquiry.," *ACM,* p. 404–417, 1961.

[10] P. E. H. Thomas M. Cover, "Nearest neighbor pattern classification," *IEEE Trans. Inf.,* 1967.

[11] h. Joachims, "Text categorization with support vector machines: Learning with many relevant features.," in *ECML*, 1998.

[12] T. Bayes, "An Essay toward Solving a Problem in the Doctrine of Chances.," *Philosophical Transactions of the Royal Society of London,* p. 370–418., 1763.

[13] B. K. L. G. a. A. T. 2. G. Singh, "Comparison between multinomial and Bernoulli Naïve Bayes for text Classification," in *International Conference on Automation, Computational and Technology Management (ICACTM).*, 2019.

[14] S. N. P. Russell, Artificial Intelligence: A Modern Approach, Prentice Hall, 2016.

[15] T. H. P. Cover, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory,* vol. 13, no. 1, pp. 21-27, 1967.

[16] T. T. R. F. J. Hastie, "(2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction Springer.," *Springer,* 2009.

[17] C. V. V. Cortes, "Support-vector networks," *Machine Learning,* vol. 20, no. 3, pp. 273-297, 1995.

[18] T. T. R. F. J. Hastie, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," *Springer.,* 2009.

[19] S. K. V. T. M. IKONOMAKIS, "Text Classification Using Machine Learning Techniques," *WSEAS TRANSACTIONS on COMPUTERS,* vol. 4, no. 8, pp. 966-974, August 2005.

[20] C.-W. L. C.-J. Hsu, "A comparison of methods for multi-class support vector machines," *IEEE Transactions on Neural Networks,* vol. 13, no. 2, pp. 415-425, 2002.

[21] W. &. R. M. &. S. P. Lam, "Automatic text categorization and its application to text retrieval," *Knowledge and Data Engineering, IEEE Transactions* , pp. 865 - 879, 1999.

[22] L. Phiri, "Automatic classification of digital objects for improved metadata quality of electronic theses and dissertations in institutional repositories," *International Journal of Metadata, Semantics and Ontologies,* vol. 14, no. 3, pp. 234-248., 2020.

[23] U. N. L. o. Medicine, "Overview of Annual Baseline Distribution of MEDLINE/PubMed Data.," MEDLINE/PubMed Record, 2018.

[24] K. A. K. G. T. W. Khor, "Applying Machine Learning to Compare Research Grant Programs," in *STI 2018 Leiden Conference on Science and Technology Indicator*, Netherlands, 2018.

[25] R. H. J. Wirth, "CRISP-DM : Towards," in *Practical Application of Knowledge Discovery and Data mining*, 1995..

[26] Abhigyan, "Calculating Accuracy of an MLModel," *Analytics Vidhya,* 2020.

[27] A. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," *Springer link,* vol. 52, p. 273–292, 2019.

[28] C. S. R. a. A. Desai, "A Review on Knowledge Discovery using Text Classification Techniques in Text Mining," *International Journal of Computer Application,* vol. 111, no. 6, p. (0975 – 8887), 2015.

[29] K. J. M. M. H. S. M. L. B. D. B. Kamran Kowsari, "Text Classification Algorithms: A Survey," *MDPI Journals,* vol. 10, no. 4, 23 April 2019.

[30] Y. B. Y. H. G. LeCun, "Deep learning. Nature," *Google Scholar,* p. 436–444, 2015.

[31] T. Verma, R. Renu and D. Gaur, "Tokenization and filtering process in RapidMiner," *International journal of applied information systems,* vol. 7, pp. 16-18, 2014.

[32] A. Kumar, "Feature Selection vs Feature Extraction: Machine Learning," *Data Analytics,* 24 March 2023.

[33] S. F, "Machine Learning in Automated Text Categorization," *ACM,* vol. 34, no. 1, pp. 1-47, 2022.

[34] J. C. T. K. a. L. S. Russell Power, "Document Classification for Focused Topics," *Association for the Advancement of Artificial Intelligence ,* vol. 1, 2010.

[35] B. Ankit, "Document classification using machine learning," *Master's Theses and Graduate Research at SJSU ScholarWorks,* 2017.

[36] S.-U. H. ,. V. ,. T. N. R. N. S. T. Iqra Safder, "Deep Learning-based Extraction of Algorithmic Metadata in Full-Text Scholarly Documents," *Information Processing & Management,* vol. 57, no. 6, 2020.

[37] X. Z. J. L. Y. Zhang, "haracter-level convolutional networks for text classification," *Advances in neural information processing systems,* p. 649–657, 2015.

[38] A. S. H. B. L. L. Y. Conneau, "Very deep convolutional networks for text classification.," *ECACL,* 2016.

[39] F. G. L. S. A. Mai, "Using Deep Learning for Title-Based Semantic Subject Indexing to Reach Competitive Performance to Full-Text," in *ACM/IEEE on Joint Conference on Digital Libraries*, ACM, 2018.

[40] N. D. T. B. Z. Christina Viola Srivastava, "Challenges and opportunities for research portfolio analysis, management, and evaluation," *Research Evaluation,* vol. 16, no. 3, pp. 152-156, 2007.

[41] B. B. E. A. F. William A. Ingram, "Summarizing ETDs with deep learning," *Cadernos BAD,* 2019.

[42] J. M. a. M. E. S. a. M. E. M. O. Jayoma, "OCR Based Document Archiving and Indexing Using PyTesseract: A Record Management System for DSWD Caraga.," in *IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology*, Phillipines, 2020.