INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 5, May 2026 | ISSN: 2456-4184 | IJNRD.ORG

BUDDY MASTER: SMART STUDY PRIORITY ANALYZER USING MACHINE LEARNING

Sathish. T[1]
Department of Advanced Computing & Analytics
Vels Institute of Science, Technology
&Advanced Studies. Chennai
India
sathish271101@gmail.com

Dr. G. Thailambal[2]
Department of Advanced Computing & Analytics
Vels Institute of Science, Technology
&Advanced Studies. Chennai
India
thaila.scs@vistas.ac.in

Abstract- Text from a course outline is analyzed in a way called as Study Priority Analyzer. The topics is determined by analyzing word patterns that are recognized by specialist. One of the approaches is machine learning methods rather than sorting. Natural language processing is a part of analyzing. Decisions on priorities carries the frequency and context analysis. A statistical model called logistic regression is used. Students get an idea of what priorities first and the guess prediction carried out with high accuracy. Prioritization is an Interpretation of structured data. The software accepts different file formats; however, its strongest point is helping students focus on important topics. While helping students organize study material, it identifies priorities so time is used efficiently.

Keywords *- Natural Language Processing, Machine Learning, TF-IDF, Logistic Regression, Topic Classification, Study Planning.*

I. INTRODUCTION

Spending more time and effort for all subjects may cause progress to be slowed down, thus wasting valuable hours, We can complex text and pattern Study Priority Analyzer helps learners see what is important. With this system, learners can make the most out of their time valuable, not based on guesses but on intelligent sorting.

A. Objectives

The objective of this project is to develop the system that can help learners identify important topics within a course outline based on analysis. Rather than spreading effort across all subjects equally, learners can cover the important topics based on the analysis. By sorting course outlines according to relevance, learners may find it easier to manage time effectively.

B. Motivation

The learners felt difficulty about where to start and how to manage the times accordingly. This is the concept of problems they facing because some can spend more time studying less relevant material due to lack of guidance. This project was created to address this based on high accuracy and efficiently. The project helps to organized the learning task, rather than guessing. Knowing that learning time management can impact one's results positively. The results we focusing is in the right place that motivate the students to be confident to face the exams based on the output priority that we analysis accurately.

II. LITERATURE SURVEY

Many research papers are created by applying natural language processing and machine learning techniques for text classification and

decision making. Earlier techniques carried and based on syllabus analysis which is more about time consuming and efficient is lesser comparing to this priority analyzer. Recently, research papers has been published on applying techniques such as TF-IDF and classification techniques such as Logistic Regression and Naïve Bayes for text classification

Many systems have been developed for text classification and recommendation systems. Moreover, most of these systems don't use automated techniques to identify the importance of topics for academic purposes. The techniques used are mostly based on generalized text classification and not on priority levels of syllabus topics.

The proposed study Priority Analyzer is different from other models because it uses Natural Language Processing techniques for text processing and Machine Learning techniques for assigning priority levels to syllabus topics so that students can optimize their study planning.

III. PROJECT DESCRIPTION

The learning starts with an assistant that is digitalized for the content according to their importance and priority. This digital system processes for the content or files instantly. Recognizing the content and reading the processes in the This digital tool processes written content or files instantly. Recognizing the content occurs through sophisticated reading processes in the background. The importance levels emerge after the engine analyses the weight and relevance of the topic. High, medium, or low - every topic has its place based on obvious signs. The choices are based on patterns in the way ideas relate to one another in the texts. Clarity is important, and finding key areas is faster if the information is obvious. One glance reveals where to put in the effort without confusion. It becomes simpler to focus after obvious priority emerges. It is simpler to start without confusion because the direction is already included. With priority focused on what requires more work, progress is natural.

A. Project Scope and Objective

The project involves creating a method to scan syllabi and determine which topics are more important. Since certain topics are more crucial to understand, the students will have a better guide for their routine preparations. Preparations can be more focused when the priorities change, which is another reason why relevance detection can lead to more intelligent preparations. The precision of the preparations can be improved by using new information for more advanced versions of the program in the future. The preparations can be more efficient as less energy is spent on less relevant materials.

B. Key Project Deliverables

Starting with the digital tool, the program looks through course outlines to identify key areas of the subject. Using machine learning, the program is able to identify the key areas without the need for human sorting. The program is also able to take information through an online space, making the process easy. Instead of complex data, the information is presented in simple forms. The most notable feature is the simplicity with which it points users towards the most important information. The site starts with the information on the course outline and then sorts the information based on its importance. The information is sorted in such a manner that it becomes easy for the users to plan their time based on the most important information

.IV. TECHNICAL SPECIFICATIONS

All the operations is executed in python and the application uses in flask to control all web related functions. Scikit-learn is helpful to charge the operations related to machine learning. TF-IDF is used for understanding the text content. Data related operations like NumPy and pandas are calculated accordingly. The success is more efficient in all the process of operations that contribute effectively in the applications. It depends on other factors for stability.

A. Core Libraries and Frameworks

Python is chosen for simplicity and strong for information management. The site operations are taken in flask libraries and their framework which smoothly navigates the users interactions. Scikit-learn is executed for the core process in machine learning. The data management operations are analysis based on the libraries called pandas. The numerical operations are taken by NumPy. The text from different file formats like PDF, Word, and PowerPoint is extracted by tools like PyPDF2, python-docx, and pptx.

V. PROPOSED METHODOLOGY

The system has a step-by-step method for studying the syllabus and determining what is of the greatest priority. It uses text analysis and a special computer program that can learn from examples to produce good results. This helps the system to determine the priority level of the content in the syllabus. The system makes use of text analysis and the Machine Learning model. It analyses the content in the syllabus. Then, it uses the Machine Learning model to determine the priority level. The results are intended to be generated every time. The system makes use of text analysis and the Machine Learning model.

A. Approach Overview

The system gets the information from the user and processes. The way of understanding language is another task her. It processes and converts language into numbers for computer understanding. The special process is to analysis the numbers along with the topics and what they trying to convert. The programs are in grouping based on the topics and importance of the syllabus. It does by using terms called TF-IDF to convert text into numbers. It also carried special kind of Algorithm in machine learning process and classify the syllabus information into different priority levels.

B. Dataset Preparation and Preprocessing

The dataset is embedded for priority levels where we can clean the dataset and preprocessing it. These processes help to ensure all the characters and text consistently. The text has formatted correctly. It has to form and analyzed. The topics in the syllabus and their priority levels are the once that analyzes based on the conditions. This step is important before actual analysis. The priority levels help in understanding the importance.

C. Hierarchical Model Development

The system carries first viewing the syllabus we uploaded then one by one the process are converted into many levels based on importance and priorities. The machine learning algorithm looks all the process in the topics and categories whether the topics are important or not. It helps to classify topics so the systems knows what to do with them.

D. Training Strategy

The model is trained on data that is labeled so that it can learn about topics and how important they are. This way, the model is better and more precise. We can trust it when it convey about topics and priority levels.

E. Analysis Framework

We use a confusion matrix to evaluate incorrect classifications that explains how to evaluate system performance through metrics like accuracy and consistency of predictions. All this helps improve the model and ensure that we get our results.

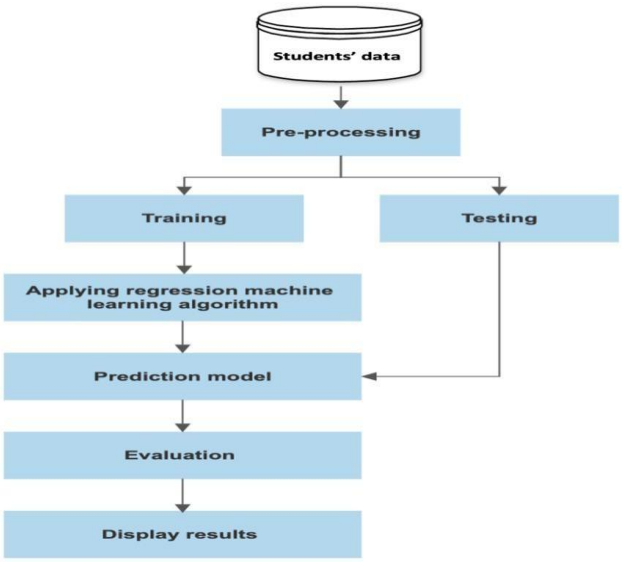

Fig 1: Methodologies

VI. ARCHITECTURE DIAGRAM

The Study Priority Analyzer is a system that has a lot of steps to follow. It takes the syllabus data that turns into something useful. The Study Priority Analyzer has parts that work together to help and turn the data into results that make sense. This helps the Study Priority Analyzer give the priority to the results.

A. Input Layer

Here, the syllabus from users in text form or in any file types such as PDF, DOCX, and PPT are acceptable in this system.

B. Text Processing Layer

we clean up the data that has been received. Remove duplicate data and irrelevant data that converts text into useful format to ensure the documents ready to analysis.

C. Classification Layer

We use a machine learning algorithm known as Logistic Regression that helps to classify the topics into three levels like high, medium, low.

D. Output Layer

We now present the output to the users in a specific format. Each topic is presented along with its priority level.

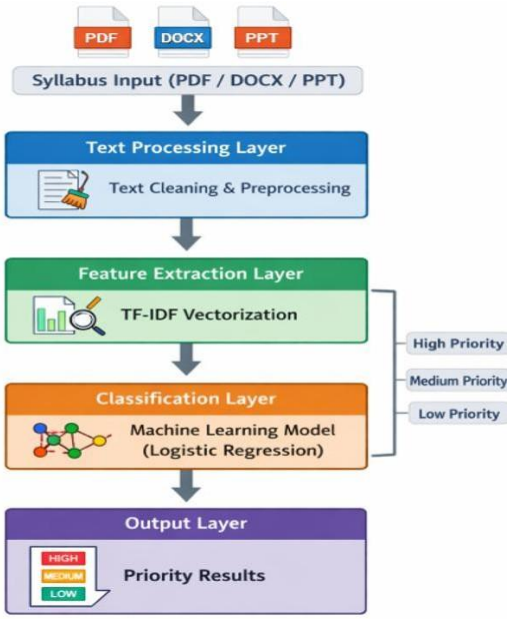

Fig. 2: Proposed Learn Sure System Architecture

VII. INPUT DATA DESCRIPTION

The Study Priority Analyzer describes the data and helps to determine what is important and which priorities more. Those data are collected and implemented the input in many formas like PDFs, word documents, Power point and text format. This tool helps and identify the topics which the data provided for your syllabus.

A. Dataset Overview

The dataset has topics related to school and college subjects. The topics relate to one idea or one chapter in a subject. The system examines each topic and gives the priority chart based on the analysis. Each topic is one idea and one chapter where it examines each topic. The dataset is made up of topics, concepts overall dataset are executed and implemented efficiently.

B. Dataset Statistics

The dataset contains:

•Topic text, which we add

•Priority label, which we obtain

C. Class Distribution

The topics are divided into three classes. We have the priority class. This class is for the topics that we need to know for the exams. We have the MEDIUM priority class. This class is for the topics that're moderately important. We have the LOW priority class. This class is for the topics that're not as important. For example, Neural Networks is a priority topic. On the other hand, Layers is a LOW priority topic. We can see this from your system output. Neural Networks is marked as HIGH, and Layers is marked as LOW.

D. Data Preprocessing Steps

Now before we examine the code that the information requires to be cleaned up. This is what we do:

- We remove words that we do not need in the text. This includes things like titles and symbols.
- We separate the text into groups that make sense. This includes different topics.
- We make sure that the information is correct and easy to read.

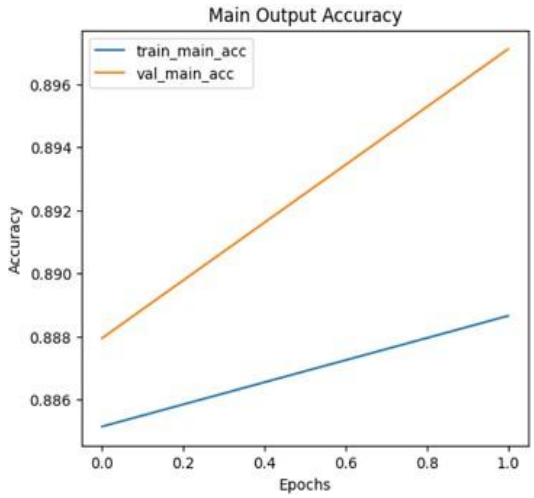

Fig 3: Training & Validation Accuracy Curves.

VIII. PSEUDOCODE & IMPLEMENTATION

We will discuss how this system is implemented to give us results based on priority after feeding the syllabus into the system.

A. Dual Output Data Generator

This system provides two different results for single topics. One result knows the priority level which gives the results like high, low, medium. The second result is all about the table format which helps us for the users to comprehend this classification that displayed for decision making process easier and it utilize effectively.

B. Model Architecture Implementation

The architecture of this model is the text and a classifier based on Machine Learning algorithm. We start by input the text and analysis using TF-IDF. After the logistic Regression model helps to classify the text.

C. Training Pipeline

In order to train the system, we have to prepare the data that extract the features, then fit the model to the data. The process carried out in the beginning of fata into vectors by TF-IDF, then trained logistic regression model identify the patterns from the topics to priority level of the topics. Once's the entire model trained we will save the model to further predications.

D. System Development

The front part of the system was developed using HTML and CSS to allow users to easily input the syllabus data, view the results. The backend of the system uses the model to process the data that provide us the results in right away, ensuring the system runs smoothly and efficiently.

IX. OUTPUT ANALYSIS

The output analysis is done to check how the Study Priority Analyzer performs with the syllabus data, how well it can provide the priority level of the topics.

A. Metric of Training Performance

The performance of the training shows how well the model is able to learn from the data that well trained for analysis. This helps us understand how accurate the predictions are. If the accuracy is high, The model has learned well in between the topics and priority level. If the results are consistent the model works well.

B. Validation Performance

The validation performance helps us understand how well the model works and understand if the model has memorized. if it works well with the syllabus data. By using the data with the model, we are able to find it for future predictions.

C. Visualization Results

The results are presented in the form of graphs and charts to show us how well the model has performed.

X. RESULTS AND DISCUSSIONS

Fig 4: Home page

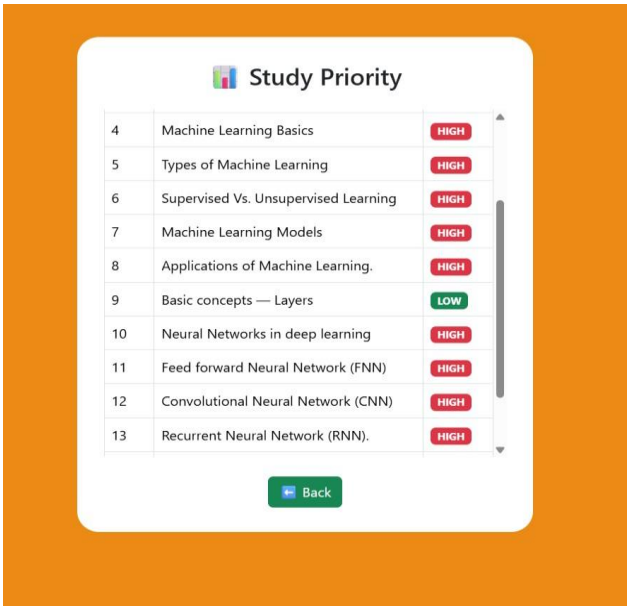

Fig 6: Result Pages

XI. CONCLUSION

The Study Priority Analyzer is used for the key topics in a syllabus. The Study Priority Analyzer uses Natural Language Processing and Machine Learning to help the students concentrate on the important topics of the syllabus, which helps in planning the studies more efficiently. It saves time by indicating the topics that are not as important. therefore, perform better at school. The Study Priority Analyzer can also be improved by using real-time data to make it more accurate and efficient for the users.

REFERENCES

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.

[2] J. Leskovec, A. Rajaraman, and J. Ullman, *Mining of Massive Datasets*, Cambridge University Press, 2014.

[3] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008.

[4] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.

[5] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[6] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, Springer, 2013.

[7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[8] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012.

[9] H. Schütze and C. D. Manning, "Foundations of Statistical Natural Language Processing," MIT Press, 1999.

[10] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, O'Reilly Media, 2019.

[11] J. Brownlee, *Machine Learning Mastery with Python*, Machine Learning Mastery, 2016.

[12] W. McKinney, "Data Structures for Statistical Computing in Python," *Proceedings of the 9th Python in Science Conference*, 2010.

[13] Flask Documentation, "Flask Web Framework," [Online]. Available: https://flask.palletsprojects.com

[14] Scikit-learn Documentation, "Machine Learning Library," [Online]. Available: https://scikit-learn.org