Ranked Stack Overflow: Mathematics & Statistical Analytics

Aboli Wankhade, Deekshita Prakash Savanur, Gouri Benni, Uzair Riyaz Pachhapure

Department of Applied Data Science, San Jose State University

DATA 298A: MSDA Project 1

Dr. Ming Hwa Wang

December 05, 2023

Abstract

In today's data-driven society, accessing and assessing solutions from the Internet is crucial for professionals. The project aims to streamline the process of finding precise and effective solutions, by designing ranking metrics and implementing an improved Stack Overflow system. It is driven by the growing need for trustworthy, graded answers in these domains. The project’s scope includes creation of a user-friendly application that answers all queries related to the field with ranked solutions by assessing upvotes & downvotes (accepted answers), user views, solution completeness by user scores, and other factors.

Measurable goals include giving the best possible solution to the user query by multi-site data aggregation while maintaining an average solution quality rating of at least 4.0 out of 5.0 and considering posted queries that are recently received. A large dataset of questions in mathematics and statistics is compiled along with the associated answers from Stack Exchange and other reliable websites. The textual data will be preprocessed and cleaned using natural language processing (NLP) techniques, and pertinent mathematical expressions and formulas will be extracted. The dataset is used to train a model based on a ranking algorithm for solution ranking in order to efficiently capture semantic linkages and contextual data. Statistical analytic approaches are used to assure the accuracy of ranking solutions. The project's quality is assessed based on the ranking system's correctness, the effectiveness of its retrieval and presentation of solutions, and the usability of the user-friendly application.

1.1 Project Background and Executive Summary

Project Background

The Digital Age is underpinned by vast and intricate data structures, with Mathematics and Statistical Analytics serving as the keystones. As the digital landscape broadens and deepens, the questions and challenges encountered by experts, educators, and learners in these fields become ever more intricate. The current online resources, while vast, are scattered and often lack cohesion.

While platforms like Stack Overflow have pioneered a collaborative model of knowledge-sharing in tech domains, there's a noticeable void when it comes to a consolidated platform dedicated to the complex world of Mathematics and Statistical Analytics. Existing resources either cast too wide a net, leading to a dilution of specialist knowledge, or they are so niche that they inhibit wide-scale interaction and collaboration. Additionally, the sprawling nature of online resources creates a maze for those seeking answers. An answer on one portal may be contradicted elsewhere, leading to confusion and wasted time.

Enter "Ranked Stack Overflow: Mathematics & Statistical Analytics." This initiative seeks to unify this scattered landscape by pooling knowledge from trusted sources and complementing it with insights from a vibrant community of experts and enthusiasts. Through advanced algorithms, users will be presented with a curated list of solutions, ranked by accuracy, relevance, and peer validation. The vision is more than just a Q&A platform—it's to cultivate a knowledge hub that nurtures collaboration, ensures information authenticity, and sets a new benchmark for online discourse in Mathematics and Statistical Analytics.

Needs and Importance

Today's digital landscape, especially within the realms of Mathematics and Statistical Analytics, is reminiscent of an expansive forest—rich in resources but challenging to navigate. A pressing demand exists for a unified platform, a beacon of sorts, to guide both novices and experts towards accurate and comprehensive knowledge. With numerous digital resources each offering a slice of the vast knowledge pie, the quest for coherent and comprehensive insights becomes tedious.

The sheer volume of information available online is double-edged. While it offers a wealth of knowledge, the challenge lies in discerning the credible from the chaff. This underscores the necessity for a stringent quality assurance system—a platform that doesn't merely aggregate but also curates, ensuring that the information presented is both authentic and relevant. This is the crux of the "Ranked Stack Overflow: Mathematics & Statistical Analytics" initiative.

This endeavor is not merely about creating a knowledge reservoir. It's about shaping a community-centric hub, driven by advanced algorithms, that puts peer-reviewed information at users' fingertips. In an era where digital clutter is the norm, having a trusted, centralized space for learning and collaboration becomes not just advantageous but essential for the continued evolution and integrity of Mathematics and Statistical Analytics.

Target problem

Mathematics and Statistical Analytics are disciplines that thrive on precision. Every day, countless individuals delve into the web's vast expanse to seek answers for their complex questions. Yet, with such an overload of information, pinpointing the right answers becomes akin to finding a needle in a haystack. Many online platforms, in their bid to cater to all, end up offering watered-down, generalized responses that lack depth and specificity.

Herein lies a significant challenge: assimilating data from sources. A single query could fetch many responses from diverse corners of the web, spanning forums, academic portals, blogs, and the like. The crux is determining the relevance, accuracy, and reliability of these answers. While the internet is abundant, specialized tools to assess, and rank this sea of information, especially for our chosen domains, are scarce.

Inaccurate or generalized information in such precision-driven fields can pave the way for foundational errors, misguided research methodologies, or even flawed practical applications. It's not just about time inefficiency; it's about the potential compromise on the quality of knowledge and insights.

The core issue that "Ranked Stack Overflow: Mathematics & Statistical Analytics" aims to tackle is multifaceted. Firstly, it seeks to bridge the void of a platform adept at synthesizing and ranking vast information using cutting-edge data analytics. Secondly, it addresses the overarching concern of ensuring that stakeholders in Mathematics and Statistical Analytics receive nuanced, validated, and context-rich information. Ignoring this pressing concern could be detrimental, causing setbacks and knowledge gaps in these crucial fields.

Motivation and Goals

Within the intricate realms of Mathematics and Statistical Analytics, practitioners and curious minds often face a dual-edged sword: the complexities innate to these fields and the overwhelming task of discerning quality insights amidst the plethora of online resources. Recognizing this unique conundrum, our project was born out of a desire not only to bridge the knowledge gap but also to innovate how information is curated and consumed in these sectors. Our vision transcends mere problem-solving; we aspire to reshape the digital landscape of learning and inquiry for these subjects.

In echoing this vision, our objectives are clear-cut. Our primary mission is to offer users a singular platform that amalgamates data from diverse digital origins. Yet, gathering information is just one part of the equation. Given the vast disparities in online content quality, we're committed to designing a sophisticated ranking system. Through cutting-edge algorithmic techniques, we aim to present answers that stand out in accuracy, credibility, and contextual relevance. In essence, our goal is to redefine the user journey: ensuring each question is met with reliable, top-tier, and tailored responses. Through this initiative, we hope to set new benchmarks for digital exploration within Mathematics and Statistical Analytics.

Project approaches and Methods

Data Collection and Aggregation. The first step in the "Ranked Stack Overflow: Mathematics & Statistical Analytics" project is to meticulously gather relevant data. We have collected and combined the datasets namely “mathematica.stackexchange.com.7z”, “mathoverflow.net.7z ”, “math.stackexchange.com.7z”, “matheducators.stackexchange.com.7z” from the Internet Archive.org.

Data preprocessing and cleaning. Once a sizable dataset is amassed, the next challenge is ensuring its quality. By harnessing python libraries, every piece of textual data is sieved, stripping off any superfluous elements. Considering the domain, the dataset consists of question and answers, all the questions which do not have answers were removed.

Model Training. To predict the tags, TF-IDF with Logistic Regression, TF-IDF with NaiveBayes Classification, TF-IDF with SVM, TF-IDF with Random Forest and Bagging with Decision Tree are used. Utilizing Hugging Face’s sentence transformer models namely “all-MiniLM-L6-v2” and “all-mpnet-base-v2”, which is an advanced model, will adapt it to the unique dataset to comprehend the intricate relations between questions and answers and rank and provide the top 3 answers. With its deep learning prowess, the goal is to gauge the pertinence of responses in the respective contexts. This early-stage assessment acts as groundwork for the next steps in ranking, guaranteeing the system's ability to discern and judge the material beyond just basic parameters.

Ranking Algorithm Development. In this phase, identify and prioritize features that dictate answer quality — factors like community user view count, the reliability of an answer, or the credibility of the user score for the answers. Drawing from these insights, sophisticated computational models, the pre-trained models are used for the ranking of answers. This ensures that the displayed rankings resonate with genuine solution quality.

Performance Measurement and Evaluation. The final hurdle is determining the project's success. Setting clear performance indicators, like user satisfaction metrics or the quality of returned solutions, offers a measurable result.. To evaluate the predicted tags for the user queries, Accuracy, F1 score, Precision, Recall, Hamming Loss is used. Furthermore, practical tests, like A/B comparisons, are executed to judge the real-world impact of system modifications for the answers based on user queries.

Expected project contributions and applications

Centralized Knowledge Base. This project seeks to offer a comprehensive collection of vetted and precise answers exclusively focused on Mathematics & Statistical Analytics. It's envisioned as a platform that emphasizes the essence of information, leaving behind redundant details.

Sophisticated Answer Prioritization. At its core, the project will utilize sophisticated algorithms to discern the value of responses, ensuring that users are immediately presented with the most fitting and credible solutions.

User-friendly Digital Environment. Designed with the end-user in mind, the platform promises an easy-to-understand system, incorporating features tailored to the unique needs of the mathematical and statistical community.

Pedagogical Resource. Scholars and instructors can lean on this resource for elucidating complex topics, verifying academic content, and fostering a more enriched educational landscape.

Operational Troubleshooting. Professionals navigating the intricate corridors of analytics or mathematical modeling will find the platform an indispensable ally. It's designed to guide them through challenges, explore novel methods, and adhere to industry standards.

Support for Scholars. Academic researchers can resort to this platform as a sounding board for their hypotheses, a solution-finder for challenging questions, or a forum to discuss nuanced methodologies.

1.2 Project Requirements

Functional Requirements

The proposed method of assessing and ranking effective solutions by ranking metrics needs to meet certain functional requirements. After assessing the most accurate solutions to the user queries, the system must be able to rank these solutions depending upon certain ranking metrics. The ranked solutions will be based upon certain metrics such as upvotes & downvotes (user scores) , user accepted answers and solution completeness. The performance of these solutions will be evaluated based on the efficiency and accuracy measure. The data for these solutions will be aggregated from Math Stack Exchange and other websites. Maintaining the quality of solutions is necessary, for this the system will maintain an average standard of rating of 4.0 on a scale of 5.0. This will be based on the user feedback and ratings. It is crucial to have a substantial dataset containing questions and answers related to mathematics and statistics for training and testing the ranking model. As the dataset contains both structured and unstructured data the system should be able to provide an efficient data storage solution.

For preprocessing and cleaning the textual data from the dataset, Natural Language Processing (NLP) techniques will be used, these techniques include removing noisy data, normalizing the text and extracting mathematical expressions and formulas. Ranking algorithms will be used to effectively rank the solution based on the pre-defined metrics of the system.

AI Powered Requirements

The system uses various machine learning or deep learning models to predict the best solution for a user query. In order to do so, the system should be fed with a large dataset containing mathematical and statistics questions and answers which will be further divided into train and test data. The models will be trained on the training data and it will make predictions on the testing data based on the ranked metrics.

TF-IDF a text-based ranking algorithm can be used that will assess the importance of each term within a document relative to its frequency in documents.

Models namely, NaiveBayes Classification, SVM (Support Vector Machines), Random Forests and Decision Trees are explored for predicting Tags.

Hugging Face’s “all-mpnet-base-v2” is applied to tasks such as clustering and semantic search, mapping sentences and paragraphs to a dense vector space of 768 dimensions.

Hugging Face’s “all-MiniLM-L6-v2” is applied to tasks like clustering and semantic search by mapping sentences and paragraphs to a 384 dimensional dense vector space.

The system should be able to accurately parse and interpret mathematical formulas and expressions within the textual data. Accuracy in mathematical expression parsing can be measured by comparing parsed expressions with ground truth data. Model performance can be tested using performance evaluation metrics such as accuracy, precision, recall, hemming loss and F1-score.

Data Requirements

In order to efficiently rank the solutions of the user queries the system should train and test well on the dataset. For this the dataset should be large enough and rich in mathematical and statistical questions and their related answers. For this the data is collected from the Mathoverflow folders under Internet Archive.org. will be used for collecting the data from dedicated various platforms. In addition to this a dataset containing various features such as post id, accepted answer id, score of the answer - which will give the upvotes and downvotes for the questions, title and body of the questions, answer count and comment count - which will give the the number of answers and comments for a particular question will also be used. Collecting and storing the user profile data, including their previous interactions, should also be done to enhance the ranking algorithm's performance. The system should also extract and store mathematical expressions and formulas from the textual data for efficient analysis. The relevance of the answer also plays a crucial role in ranking the solutions. This relevance can be determined using the maximum keywords match between the question and answer.

1.3 Project Deliverables

Table 1

Deliverables Including Reports, Prototypes, Development Applications, and/or Production Applications

Deliverable	**Description**	**Due Dates**

Project Proposal	This document outlines the project's objectives, scope, and methodology, providing a clear roadmap for stakeholders and ensuring alignment with project goals.	09/06/2023
Work Breakdown Structure (WBS)	A hierarchical breakdown of project tasks and activities, facilitating efficient project management, resource allocation, and progress tracking.	09/29/2023
Gantt Chart	A visual timeline representation of project tasks and their dependencies, aiding in project scheduling, task prioritization, and deadline management.	10/02/2023
PERT Chart	A graphical tool for project planning and risk assessment, illustrating task dependencies and identifying critical path activities.	10/03/2023
Project Management Plan	A comprehensive document detailing project roles, responsibilities, communication strategies, risk management, and quality assurance measures.	10/04/2023
Data Collection Plan	A structured approach to gather relevant data, specifying sources, methods, and frequency	10/09/2023

	to ensure data quality and consistency.	
Data Exploration	An initial analysis of the collected data to identify patterns, outliers, and trends, informing subsequent preprocessing steps.	10/12/2023
Data Cleaning and Preprocessing Code	Code scripts and procedures for data cleaning, transformation, and preparation, ensuring data suitability for modeling.	10/19/2023
Predictive Models	Development of machine learning models based on the collected data, aimed at ranking solutions effectively.	10/24/2023
Performance Metrics	Defining evaluation criteria to assess the accuracy and effectiveness of predictive models in ranking solutions.	11/06/2023
Model Evaluation and Selection Report	A comprehensive report detailing the evaluation results of different models and the rationale for selecting the final ranking algorithm.	11/29/2023
Project Presentation	A formal presentation summarizing project goals, methods, findings, and deliverables for stakeholders and peers.	

Final Project Report	An in-depth report containing a summary of the project's approach, findings,and suggestions for future research.	

The first deliverable is a project proposal in which there is a detailed explanation of the project's objectives, scope, and methodology. It also contains clear clarification of the project goals. To define the scope of the project, propose tasks and deliverables, and allocate resources, different charts like Work Breakdown Structure (WBS), Gantt Chart, PERT Chart will be used to define the proper structure of the project. The Project Management Plan, which is the next step, is a document providing complete detailing of project roles and responsibilities. Data management and exploration is the next phase in which there is a structured approach of Data Collection, Data Exploration and Data Cleaning. This section explains how the data is collected, what are the approaches followed and managed. This section also explains how data will be stored and how it will be processed. Data cleaning objective is to make the data modeling-ready by ensuring that it is error- and inconsistency-free.

The Performance Metrics consists of reports of the machine learning algorithm aimed at ranking solutions. The performance is evaluated in the performance metric reports that define evaluation criteria to assess the accuracy and effectiveness of all the used models. The Model Evaluation and Selection Report will be the final report of models that are selected, and their comprehensive evaluation. The project presentation will be the final presentation that summarizes project goals, methods, findings, and deliverables. The final report will be the detailed explanation of all the sections and an in-depth report of project approach, findings,and suggestions for future research. These meticulously crafted deliverables collectively advance the project's objective of creating an improved solution-ranking platform, aligning seamlessly with the growing need for precise and effective solutions in our data-driven society. Each deliverable serves as a building block, contributing to project efficiency, quality, and, ultimately, overall success.

1.4 Technology and Solution Survey

The technology survey includes a number of models and deep learning techniques that are applied while ranking answers from various websites. The choice of model depends on the details of the problem and the qualities of the data, and each model has advantages and disadvantages.

The goal of the project by Agarwal et al. (2021) is to overcome the shortcomings of current domain-specific QA frameworks and improve the precision of user inquiries by introducing a ranking-based question-answering system. To find the most effective model, the method compares a wide range of advanced models, including QANet with U-Net features, XL-Net, and BERT-Large. The selected model is then incorporated into a web and mobile application to produce a chatbot that can answer questions and extract contextual responses. Additionally, a Deep Learning classifier is suggested to raise the classification accuracy of user questions.

In a study by Setiawan et al. (2021), they developed a method to rate potential answers for fact-based queries in an Indonesian restricted domain question answering system. Their architecture, which included Question Analysis for query creation, Candidate Document Selection, Candidate Document Analysis, Answer Extraction with weighted scores, and Response Generation, merged information retrieval and natural language processing. Utilizing Answer Ranking's weighted scores, their method attained a remarkable accuracy rate of 54%.

Advanced NLP techniques, such as morphological to semantic processing, were credited with this improvement. Additionally, removing unnecessary words from user queries could improve efficiency by 15% to 39%.

Perera et al. (2020) sought to improve factoid-based question and answer (QA) systems on the web by addressing the difficulties in rating potential answers. Their strategy includes increasing Question Analysis by merging phrase and boolean queries, as well as improving Answer Extraction by incorporating weighted features and question tokens. Using NLP methods and data augmentation, the performance of three deep learning models were assessed: Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), and Conventional Long Short Term Memory (CLSTM).

Using an actual question-answering system, Li et al. (2016) addressed the answer ranking issue by highlighting the significance of ranking answer-bearing sentences correctly. In order to capture both structural relevance and semantic similarity, their method involved creating a composite representation for questions and responses using models from bidirectional long short-term memory (biLSTM) and convolutional neural network (CNN). They included a hypernym method to supplement the training data, improving the robustness of the model. The deep learning-based strategy outperformed earlier approaches that relied on syntactic characteristics and some deep learning models in the evaluation on a TREC benchmark dataset.

Wang et al. (2023) address the problem of lowering the labeling efforts needed for training deep learning models, notably CNN-based and LSTM-based answer selection models, by introducing a deep active learning framework for answer selection tasks. The solution uses the Deep Expected Loss Optimisation (DELO) technique,which enables batch-mode active learning. A thorough analysis carried out on real-world datasets like YahooCQA and

SemiEvalCQA, illustrates the surprising benefits of DELO, showing noticeably lower labeled sample requirements than competing techniques.

As part of this research, Özyurt and Grethe (2019) unveiled Bio-AnswerFinder, a biomedical question-and-answer system made to handle the difficulties brought on by the growing body of scientific material. They used supervised deep learning approaches for keyword selection and relevance ranking within a greedy iterative retrieval framework to tackle the challenge of document retrieval. A baseline employing syntactic parsing, an ensemble strategy, and an approximation of the k-nearest neighbor method based on word embeddings were all used in comparison to this creative approach and uses deep learning models like Bidirectional Encoder Representations from Transformers (BERT) and Glove word/phrase embeddings, significantly improved Mean Reciprocal Rank (MRR@10) performance.

Yu et al. (2017) introduced a hybrid approach to improve factoid question answering by combining information retrieval (IR) grammars, Latent Dirichlet Allocation (LDA) grammars, and Doc2Vec models with deep learning techniques. Their system leveraged both term-level and semantic-level knowledge to enhance answer ranking. They utilized an IR indexer for term-level document retrieval and an LDA indexer for implicit semantic relevance identification. Candidate answers were scored using cosine similarity with the question, leading to improved ranking accuracy. Experimental results demonstrated the effectiveness of our approach, particularly the contribution of the LDA model in enhancing answer ranking.

El-Batrawy et al. (2020) tackle the difficult challenge of assessing the relevance of images within the field of computer vision. They offer a ranking model based on the ideas of transfer learning and RetinaNet by utilizing the potential of deep learning. With RetinaNet, a one-stage detector that effectively extracts features from whole images while giving "hard" samples priority to increase prediction accuracy, we aim to improve the ranking retrieval process. The work represents a significant advance in the field of deep image ranking due to their contributions, which include using convolutional neural networks (CNNs) for feature extraction, using transfer learning ideas and showcasing RetinaNet's efficacy in enhancing image ranking accuracy.

In order to address the problem of understanding user-generated comments and reviews, which is common in the era of ubiquitous internet usage and e-commerce platforms, Akinlaja and Mosia (2021) go into the field of Natural Language Processing (NLP). The study introduces techniques for determining the true intent behind customer reviews and then identifying disparities between these evaluations and accompanying ratings by utilizing sentiment analysis and a Deep Learning Convolutional Neural Network.

Utilizing developments in mobile web technology and the rising need for online healthcare services, X. Li et al. (2019) present a two-stage framework for semantic matching of user medical questions. The strategy optimizes the automatic retrieval of medical information by combining deep learning models, such as Siamese-inspired recurrent neural networks, with established information retrieval methods, such as TF-IDF and soft search. This framework's performance in expediting online medical consultations was evaluated on a significant mobile medical platform with over 0.3 million daily active users.

1.5 Literature Survey of Existing Research

The research by Omondiagbe et al. (2022) delves into the assessment of model performance and quality for the prediction of accepted answers on Stack Overflow, a widely used platform for developer questions. The study employs a dataset composed of Stack Overflow posts spanning the years 2014-2016. Four models are employed, including Random Forest and

Recurrent Neural Network (RNN) models, both with and without hand-crafted and neural-generated features. The results reveal that the RNN model equipped with hand-crafted and neural-generated features outperforms all other models, achieving an impressive balanced accuracy of 82.73%. The research employs various techniques such as feature engineering, sampling methods like SMOTE and ADASYN, and hyperparameter tuning. Evaluation metrics such as balanced accuracy, F1-score, and Matthews Correlation Coefficient (MCC) are utilized to gauge model performance. Additionally, the study incorporates a developer questionnaire to validate model predictions from a human perspective. Notably, the RNN model with hand-crafted and neural-generated features emerges as the most accurate in predicting accepted answers, offering valuable insights for software developers aiming to integrate Q&A prediction with Stack Overflow.

This scholarly paper, by Yazdaninia et al. (2021) titled "Characterization and Prediction of Questions without Accepted Answers on Stack Overflow," addresses the growing concern surrounding unanswered questions on Stack Overflow, a prominent platform for programming-related queries. The study conducts a comprehensive examination of factors that influence the likelihood of a question receiving an accepted answer. It introduces innovative features related to tags, content quality, and user engagement, aiming to predict this critical outcome. Utilizing predictive modeling, particularly the XGBoost algorithm, the paper demonstrates the significant impact of these features on predicting whether a question will obtain an accepted answer. The top-performing model achieved an impressive AUC score of 0.70, signifying strong predictive capabilities. Additionally, the research highlights the relative importance of various features, offering insights into the key determinants of question resolution. This paper offers valuable guidance for Stack Overflow users striving to enhance the quality of their questions and researchers interested in understanding the dynamics of online programming communities.

Chen et al. (2022) introduces an innovative recommendation system designed to efficiently match and cluster mathematical questions available on the internet. The proposed approach employs a two-stage process: initially, questions are matched based on their difficulty levels, followed by ranking them using a BERT-based semantic similarity calculation. The foundation of this system lies in a knowledge graph that encapsulates mathematical questions, encompassing knowledge points, solution steps, and difficulty values. To construct this graph, the paper employs techniques such as Named Entity Recognition and relationship extraction. Notably, the BERT model plays a pivotal role in computing semantic similarity. The experimental results demonstrate the efficacy of the BERT-based method, achieving an impressive semantic similarity accuracy of 93.1%, surpassing alternative methods like ESim and Linkage. Consequently, the BERT-based approach not only excels in accuracy but also enhances the efficiency of mathematical problem recommendation algorithms.

In this research paper, by Qin et al. (2022) titled “Related Questions Retrieval Model in Stack Overflow based on Semantic Matching”, the goal of the study is to make it simple for programmers to utilize Stack Overflow to obtain answers. When any programmer detects any problem in the code they search for some similar answers in the stackoverflow. The problem in the currently built system is that it doesn't always find the best matches. So to understand the question better they have created a system called RMSO. Deep Learning techniques such as “Integral Fusion” is used by the researchers to understand the whole question at one go and “Inter-Attention” is used to focus on an important part of the question. To understand the software engineering terms, the researcher has used the Word2Vec skip-gram algorithm. NLP tasks such as question similarity or relatedness detection are done using recurrent neural networks (RNNs) or transformer-based models. The evaluation metrics such as MRR (Mean Reciprocal Rank), NDCG@5 (Normalized Discounted Cumulative Gain at 5) and NDCG@10 (Normalized Discounted Cumulative Gain at 10) are used to evaluate their related questions retrieval model (RMSO).

In Subramani et al. (2023) research paper, the researcher focuses on the prediction of the tags or labels that should be attached in the questions based on what questions are about. For instance, if any user is asking a question about any bugs to be fixed in a specific programming language, their system should automatically suggest the tag name “bug” for the specific programming language. To implement this they have used deep learning methods called Long Short Term Memory, Gated Recurrent Unit (GRU) and Multi-Layer Perceptron that helps the system to learn the text in the questions and suggest right tags for them. This has become useful to help users to organize questions and make it easier to find answers to that question. The evaluation of the model is done using test accuracy, hamming loss, subset accuracy, jaccard score, precision, recall & fl score. After comparing all the models based on the evaluation metrics, GRU algorithm found to be better performing which has highest subset accuracy and lowest hamming loss as compared to other models.GRU algorithm found to be better performing which has highest subset accuracy and lowest hamming loss as compared to other models..

Rahmah et al. (2019) discusses how technology can improve learning in higher education, known as Technology-Enhanced Learning (TEL). As technology evolves, it's important to understand how to use it effectively in education. The research aims to find key words or terms that describe TEL and its success factors. They use two methods, one called "Luhn's significant words" and another called "TF-IDF words weighting," to find these important words from existing research. They want to see if the results from these methods match and if they can get a comprehensive understanding of TEL. Essentially, they are trying to figure out what's important when it comes to using technology for learning.

The goal of the study by Kadhim (2019) was to find a solution to the problem of feature extraction in text categorization, especially in the setting of Twitter data. The importance of text mining and feature extraction in managing the enormous volume of unstructured data present on social media networks like Twitter was underlined. For the evaluation, the well-known feature extraction methods BM25 and TF-IDF were used. In order to extract keywords and rank documents, the strategy involves gathering Twitter data via API techniques and using BM25 and TF-IDF to extract those keywords. Categorization performance indicators, such as the F1-measure was used. The outcomes clearly showed that TF-IDF performed better than BM25, improving the effectiveness of feature extraction for Twitter data. This study's machine learning models and deep learning methods served as the foundation for the analysis and performance comparison of the BM25 and TF-IDF models.

In order to account for diverse user interests across various topic areas, X. Wang and Yuan (2009) extended the BM25 ranking algorithm to create an improved recommendation mechanism. In order to reflect users' varied preferences, the project's methodology required translating their interests into discrete word sets and modifying BM25 to weigh phrases in accordance with individual interest levels. Using actual course selection data from Tsinghua University, a course recommendation system was developed to assess this unique technique. The outcomes of the tests showed the efficacy and viability of our suggested approach, highlighting its potential use in a variety of recommendation situations across information systems in addition to course recommendation. The innovative adaptation of BM25 that this project makes, utilizing deep learning and machine learning techniques, improves the relevance of recommendation candidates to users with different interest profiles, ultimately optimizing the recommendation process for greater user engagement and satisfaction.

Xu et al. (2010b) introduced Channel Distribution Information (CDI) IDF schema to address the problem of improving text feature extraction in web news. The project's introduction emphasized the necessity of accurate feature representation for successful news analysis as well as the increased complexity of information organization and access in the age of online news sources. By utilizing distribution data from multiple news outlets, we refined the IDF values of terms and were able to differentiate between Top terms and meaningless terms. The innovative application of distribution information among news channels to improve text feature extraction, the use of machine learning models and deep learning techniques to achieve superior results in news web page analysis, and the improvement of information processing accuracy and efficiency are the main contributions of this project.

Trang and Shcherbakov (2020) explored various BERT models, including multilingual and language-specific versions, for Vietnamese question answering (QA) systems in their research. While monolingual BERT outperformed multilingual models, the study recommended using multilingual BERT fine-tuned on XQuAD as an alternative for building a Vietnamese QA system. The BERT model has revolutionized natural language processing (NLP) tasks, particularly question answering, across multiple languages. QA systems are in high demand, and BERT's effectiveness has been demonstrated in English. The study leverages BERT, including Multilingual BERT (M-BERT), to construct a Vietnamese QA system, addressing the need for precise question responses. The study confirmed BERT's effectiveness in NLP tasks and found that the Vietnamese-specific BERT model, PhoBERTbase, achieved the best F1-score for question answering. Among multilingual models, mBERT_XQuAD performed the best. The study recommends mBERT_XQuAD for a Vietnamese QA system, with future plans to enhance performance using larger datasets, knowledge bases, and TPUs for efficiency.

Self-supervised transformer-based models, powered by deep learning, have revolutionized transfer learning in natural language processing (NLP) due to their self-attention mechanism. Akhila et al. (2023) in their study compared seven prominent models, including Bertbase-uncased, Distilbert-base-cased, Distilbert-base-uncased, and Roberta, for their effectiveness. Roberta models exhibit the highest accuracy after three epochs, while Distilbert-base-cased models outperform Bert-base-uncased, Distilbert-base-uncased, and Distilbert base-cased models. Electra-base-squad2 outperforms Bert base-cased and Bert-medium-squad2-distilled in cases with two epochs, although Bert and Roberta models require substantial training time to achieve higher accuracy. In an era of information abundance, effective information retrieval is paramount. Question Answering (QA) systems, a subset of natural language processing (NLP), play a crucial role in this context. The study delves into extractive question answering, where models like BERT, employing Transformer architecture, have shown success. Stanford's SQuAD dataset serves as the foundation, and BERT is compared to RoBERTa, a robustly optimized approach for pre-training NLP systems. Transformers, with their self-attention mechanism, are widely utilized in NLP and computer vision. The study's findings reveal that ROBERTa outperforms other BERT models due to lower training and validation losses. Among BERT variants, Distilbert-base-cased performs exceptionally well. The comparative study involves models trained for three and two epochs, providing valuable insights into their performance.

The research by Cheng et al. (2022) introduces a BERT-based hybrid question answering matching model that improves upon existing similarity matching methods. The model effectively leverages BERT's semantic information representation and integrates syntactic features through Bi-LSTM and GCN layers. The proposed algorithm is validated on two datasets, demonstrating its effectiveness in question answering. Question answering systems (QAS) are vital for retrieving precise natural language answers from user queries. This study highlights the division between deep learning and conventional methods in similarity matching for QAS. The paper introduces the BERT_Bi-LSTM_GCN model, combining BERT's pre-training with advanced feature extraction to enhance sentence pair judgment and enrich semantic information. The model's performance is compared to common multiple answer selection algorithms on two datasets. Experimental results on the WikiQA dataset indicate that the BERT_BLSTM_TCNN model, despite its slightly increased parameter size, significantly outperforms the BERT-base model, achieving higher accuracy and superior performance in question answering tasks.

2.1 Data Management Plan

Data collection approaches

Gathering information is a pivotal initial step, involving specific strategies to accrue precise and applicable data that form the cornerstone for any analytical endeavor. These tactics are designed to be orderly, adhere to ethical standards, and be tailored to meet the needs of the project to guarantee the dependability and excellence of the data amassed. In the context of the "Ranked Stack Overflow: Mathematics & Statistical Analytics" initiative, the approach to amassing data is finely honed, with a keen focus on web scraping and harnessing precompiled datasets. With the aid of specialized Python tools, data pertinent to the domain is harvested from a select array of online mathematics and statistics forums and platforms. In lockstep, readily available datasets, replete with valuable insights, are assimilated to broaden the analytical horizon and depth of evaluative scrutiny. Throughout each juncture, paramount emphasis is placed on upholding ethical norms and legal protocols to ensure the sanctity and caliber of the data collated.

Data Management methods

Management practices are crucial for effectively handling, safeguarding, and maximizing the utility of accumulated data. These are tailored to ensure easy accessibility and reliability while maintaining a systematic arrangement of the data. For the project, there is an emphasis on implementing an organized database infrastructure tailored to efficiently handle and oversee the diverse data sets. After data is collected, it’s systematically arranged and labeled to facilitate swift and efficient access for detailed analysis. Precautionary measures are strengthened to ensure the confidentiality and security of sensitive data, aligning with the highest standards of data privacy. Additionally, the incorporation of a version control system aids in documenting each modification made to the data, providing a detailed account of its transformation over various phases.

Storage methods

Git plays a crucial role in version control, meticulously tracking and managing every change to ensure the project’s accuracy and currency. Its capability to create multiple branches enhances the development and testing phases, allowing seamless experimentation without impacting the primary code. Data transformation is another critical phase, where raw data is refined into an analyzable and insightful format. This involves thorough data cleaning to eliminate inconsistencies and errors, and data normalization to standardize and optimize the datasets for in-depth analysis.

A well-organized folder system is employed to facilitate intuitive data access and management. Distinct folders for different data stages, including raw, cleaned, processed, and analyzed data, are established. A comprehensive file naming protocol, incorporating elements like source, timestamps, and descriptive identifiers, ensures quick and easy data retrieval.

From the onset, the project is anchored by clearly defined guidelines for data structure, file naming, and formatting. This consistency ensures that the team adheres to uniform practices, promoting efficient data management, seamless sharing, and effective analysis, ultimately bolstering the project’s objective of delivering ranked and insightful solutions in Mathematics & Statistical Analytics.

Usage mechanism

In the "Ranked Stack Overflow: Mathematics & Statistical Analytics" project, strategies for data usage are comprehensive and systematic, ensuring data is not only accessible and secure but also optimally utilized for meaningful insights. Data housed within organized repositories is effortlessly accessed, with each modification meticulously recorded. This process assures consistency, facilitating a cohesive team effort where every participant is well-informed and aligned. Responsibilities regarding data handling are allocated specifically, ensuring a streamlined operation where accuracy, security, and consistency are paramount.

A well-defined framework dictates the ethical and legal usage of data. Adherence to stipulated guidelines is rigorously maintained to assure ethical integrity and legal compliance. The multifaceted approach to data usage combines technology and human oversight, ensuring data is leveraged effectively while upholding the highest standards of security and ethics.

Roles within the team are designated with precision. While one individual is entrusted with assuring the quality and comprehensiveness of collected data, another oversees the adherence to ethical norms and legal mandates in data utilization. The task of transforming raw data into comprehensible insights falls to team members specialized in data analysis and reporting. They are tasked with delivering articulate and visually supported interpretations of the data.

2.2 Project Development Methodology

(1) Data analytics with intelligent system development life cycle.

The project is rooted in a systematic approach following the CRISP-DM model. Initially, it hones in on identifying and articulating the intricate issues tied to delivering quality, rapid, and engaging responses to complex mathematical and statistical inquiries. The variability in the quality and speed of answers offered is the principal challenge to be tackled.

The second phase is an intense engagement with data collection and refinement. A plethora of data is extracted via web scraping from a range of credible sources, followed by a rigorous cleaning process. This ensures that the foundational dataset is not only expansive but also pristine, laying the groundwork for the subsequent analytical processes.

Analytical prowess is showcased as the project delves into employing advanced machine learning tools like BERT to decipher complex patterns and insights embedded within the data. These insights are pivotal, feeding into the creation of a dynamic and efficient ranking algorithm that ensures users receive answers that are not just relevant but ranked for their quality and applicability.

Model construction integrates these insights to birth a system characterized by its dynamism and user-centric design. It’s a balance between sophisticated algorithmic processing and intuitive user interaction, ensuring every user finds value and engagement.

Assessing the model’s efficacy involves a comprehensive review against benchmarks of accuracy, user engagement, and relevance of ranked responses. Each assessment iteration is a step towards refinement, enhancing the model's capability to deliver consistently valuable outputs.

Visual representation of insights and data is realized through advanced tools, transforming complex datasets into understandable, actionable visual narratives. The final deployment sees the model transitioning into a live, adaptive entity, constantly learning and evolving with every user interaction.

The project embodies a blend of CRISP-DM’s structured methodology with agile adaptability, ensuring each phase, from initial understanding to final deployment, is marked by precision, clarity, and user engagement, resulting in a platform that’s as insightful as it is user-friendly.

(2) Planned project development processes and activities

The development journey of the project begins with an in-depth analysis of existing platforms offering answers in these specialized fields. We employ advanced web scraping methodologies to extract diverse data from a range of trusted online sources, ensuring a comprehensive and rich dataset. This collection undergoes a meticulous cleaning process, aided by customized algorithms that guarantee its accuracy and reliability, setting the stage for the analytical phase.

In the analytics stage, tools like BERT specifically sentence transformers like all-MiniLM-L6-v2 and ll-monet-base-v2 become central, unveiling intricate patterns and insights that contribute to crafting an efficient ranking algorithm. The model undergoes continuous assessments and refinements to meet established performance standards. The user interface is engineered for optimal user engagement, enriched with data visualization tools that translate complex data insights into easily understandable formats. The evolution of "Ranked Stack

Overflow" doesn’t halt at deployment; it marks the beginning of an era of constant refinement. Each user interaction is a source of learning, contributing to the ongoing enhancement of the system, ensuring it remains a dynamic and adaptive resource in the world of mathematics and statistical analytics.

2.3 Project Organization Plan

Work breakdown structure presenting the hierarchical and incremental decomposition of the project into phases, deliverables, and work packages.

Figure 1

Work Breakdown Structure (WBS)

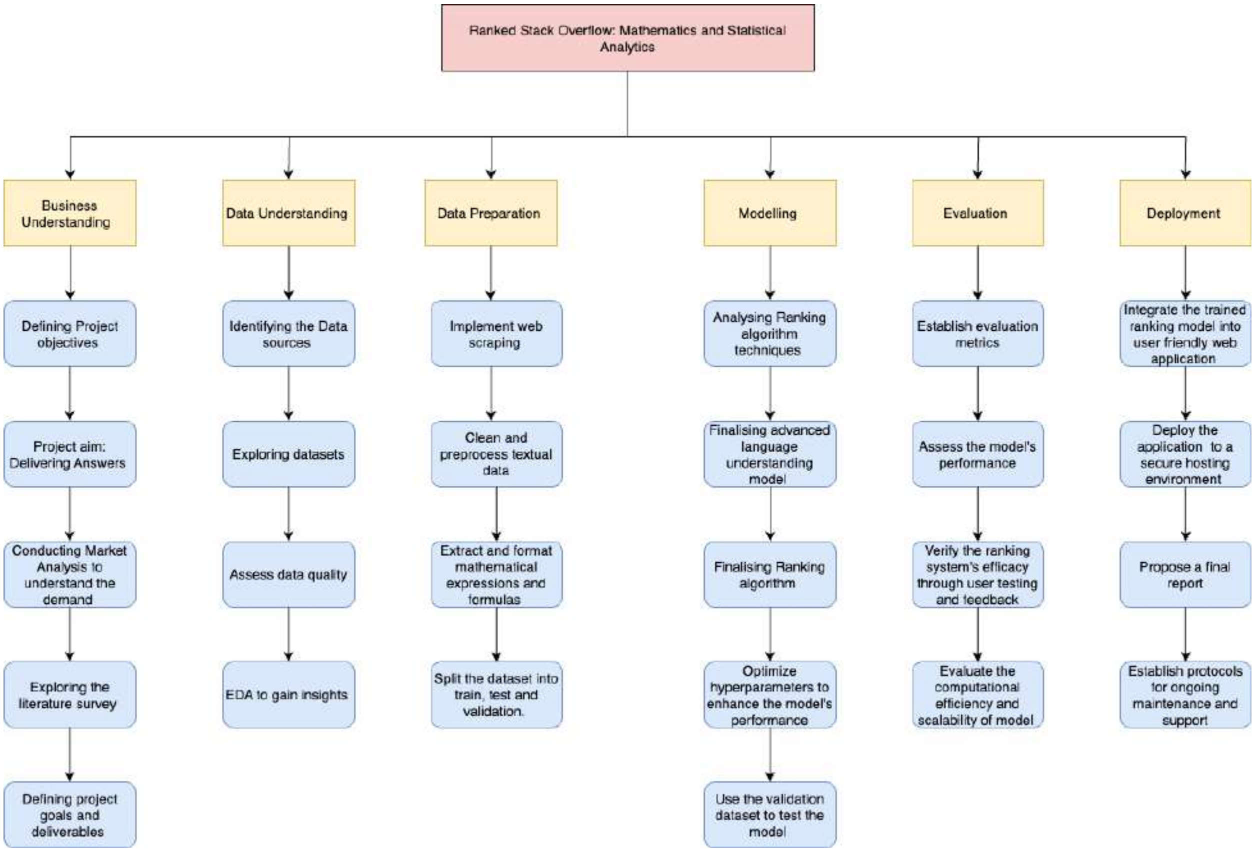

Ranked Stack Overflow: Mathematics and Statistical Analytics
Business Understanding
Defining Project objectives
Project aim: Delivering Answers
Conducting Market Analysis to understand the demand
Exploring the literature survey
Defining project goals and deliverables
Data Understanding
Identifying the Data sources
Exploring datasets
Assess data quality
EDA to gain insights
Data Preparation
Implement web scraping
Clean and preprocess textual data
Extract and format mathematical expressions and formulas
Split the dataset into train, test and validation.
Modelling
Analysing Ranking algorithm techniques
Finalising advanced language understanding model
Finalising Ranking algorithm
Optimize hyperparameters to enhance the model's performance
Use the validation dataset to test the model
Evaluation
Establish evaluation metrics
Assess the model's performance
Verify the ranking system's efficacy through user testing and feedback
Evaluate the computational efficiency and scalability of model
Deployment
Integrate the trained ranking model into user friendly web application
Deploy the application to a secure hosting environment
Propose a final report
Establish protocols for ongoing maintenance and support

Figure 1 shows the complete work breakdown structure of the project. The work breakdown structure includes various components such as business and data understanding, data preparation, modeling and evaluation, and deployment. The project's work breakdown structure highlights a systematic approach to building a ranking system for mathematical solutions. Each phase is meticulously designed to contribute to the overall success of the project, ensuring the delivery of a user-friendly application that provides trustworthy and graded answers in the domains of mathematics and statistics.

Business Understanding

In this phase the main task is to understand the business objectives and requirements of the project. This involves identifying stakeholders, defining project goals, and establishing success criteria. The business understanding phase is critical for setting the foundation of the project. Clear project goals have been outlined, emphasizing the importance of providing precise and effective solutions to mathematical queries. This phase serves as a compass, guiding the project toward delivering valuable solutions in the realm of mathematics and statistics.

Data Understanding

The Data understanding phase involves collecting and exploring the dataset, gaining insights into its characteristics, and preparing for subsequent steps in the project. Data understanding phase is initiated by employing web scraping tools to collect a rich dataset of mathematical questions and answers from various sources, with a primary focus on Stack Exchange. An exploratory data analysis (EDA) is conducted to unveil patterns and distributions within the dataset. Visualizations provided insights into the nature of the data, setting the stage for informed decisions in subsequent phases.

Data Preparation

In the Data Preparation step, the collected data is cleaned, preprocessed, and organized for modeling purposes. The data preparation phase commenced with text preprocessing, including tokenization and the removal of noise and unnecessary characters. The extraction of mathematical formulae and expressions from the textual material received particular focus. Furthermore, the dataset was split into training, validation, and test sets to facilitate the subsequent modeling phase.

Modeling

The modeling phase involves selecting appropriate algorithms, integrating all-MiniLM-L6-v2 and ll-monet-base-v2 for textual understanding, training the model, and optimizing hyperparameters. For the Ranking system, a hybrid approach is considered, integrating both traditional models and neural network-based models. The implementation involves machine learning models like Logistic Regression, Naive Bayes, SVM, Random Forest and Bagging with Decision Tree for classifying the tags of the questions and all-MiniLM-L6-v2 and ll-monet-base-v2 embeddings for deep contextual understanding and learning-to-rank the solutions.

Evaluation

Evaluate the performance of the model using defined metrics and validation datasets. The evaluation phase is crucial in assessing the effectiveness of the ranking system. Metrics such as precision, recall, and relevance scores will be employed to gauge performance. A robust validation process ensures that the model generalizes well to new data, and user feedback mechanisms will be incorporated to capture real-world usability and effectiveness.

Deployment

In this phase, the user-friendly web application is developed, and the model is integrated for real-world use. The deployment phase involves translating models into a user-friendly web application. The integration of the ranking system ensures that users can access and benefit from the enhanced solutions. Emphasis is placed on data aggregation from multiple sources to provide diverse and comprehensive solutions.

2.4 Project Resource Requirements and Plan

Hardware requirements describe the particular hardware elements needed by a system to successfully carry out a certain task or function. In accordance with the software and programmes being utilized as well as the kind of operation or function being carried out, the requirements may change. A MacBook laptop is being utilized with particular hardware setups for our purpose. For the project to be completed successfully, the necessary hardware specifications must be present. The CPU, GPU, and other key hardware elements of the system, as well as each component's individual functions, are described in depth in Table 1.

Table 2

Hardware requirements

Hardware	Configuration	Memory	Cost
8-core CPU with 6 performance cores and 2 efficiency cores	SSD : 256 GB	16 GB	600$
14-core GPU	Configurable to 1TB	16 GB	300$

For the study, software requirements are equally as crucial to take into account. Specific software programmes are needed for our project, including the Python programming language, Jupyter Notebook, and Microsoft Word. The seamless execution of the project can be greatly impacted by using the appropriate software versions and making sure compatibility is established. The requirements are listed below:

Table 3

Software requirements

Software	Version	Purpose
Python (Libraries including Pandas, Numpy, Matplotlib, Seaborn and scikit - learn)	3.9.13	Data Preprocessing, Cleaning, Analysis, visualizations, Model Building
NoSQL	8.0.29	Manage, manipulate, and perform complex operations on the dataset.
Tableau	2023.1	Visualizing and analyzing the data.
GitHub	3.8.0	Version control and to track code changes.

The project requires a certain set of tools and licenses in addition to the necessary hardware and software. To avoid any legal or ethical difficulties, it is essential to make sure that all tools and licenses are current and that all licensing criteria are met.

Table 4

Tools specifications

Tools	License	Purpose
Jupyter Notebook	Open Source	Widely used for data analysis, visualizations, and machine learning and deep learning tasks
JIRA	Cloud Licenses	Task tracking and project management
Draw.io	Free	
GitHub	Free	Tracking code changes
AWS	Pay-as-you-go pricing	Storing data

Table 5

Resource specifications

Resources	Duration	Cost	Justification
Grammarly	6 months	$70	Grammar check, spell check, sentence organization
Jira	6 months	$0	Project management, task tracking and assigning
Scribbr	8 months	$0	Citations

2.5 Project Schedule

Gantt Chart

A common tool for creating a project schedule that shows the tasks, timetable, accountable team members, and the status of deliverables is the Gantt chart. To split the project down into smaller components and produce a project schedule, a WBS Gantt chart tool is employed. The Gantt chart displays each task's start and end dates, duration, and interdependencies. Based on their significance and dependency on other projects, the tasks for each sprint are scheduled and prioritized. For instance, data collection cannot begin until the project requirements phase is finished, and accuracy assessment cannot begin until modelling is finished. Each task is given to a responsible team member, and deliverable due dates are established.

Figure 1

Gantt chart - Business understanding

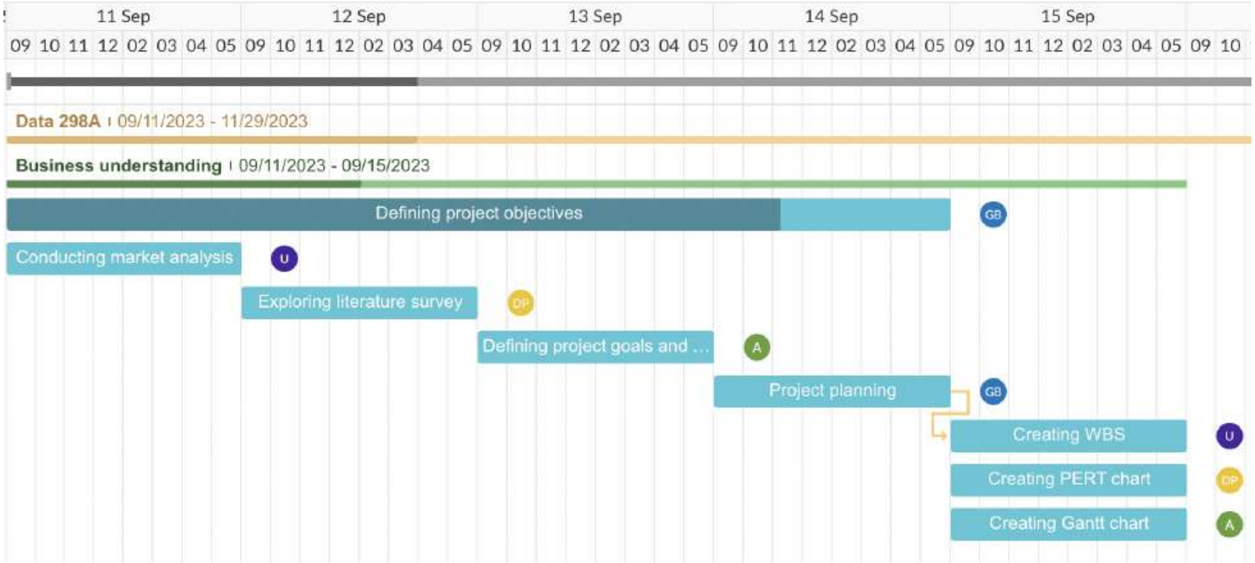


Figure 2

Gantt chart - Data understanding

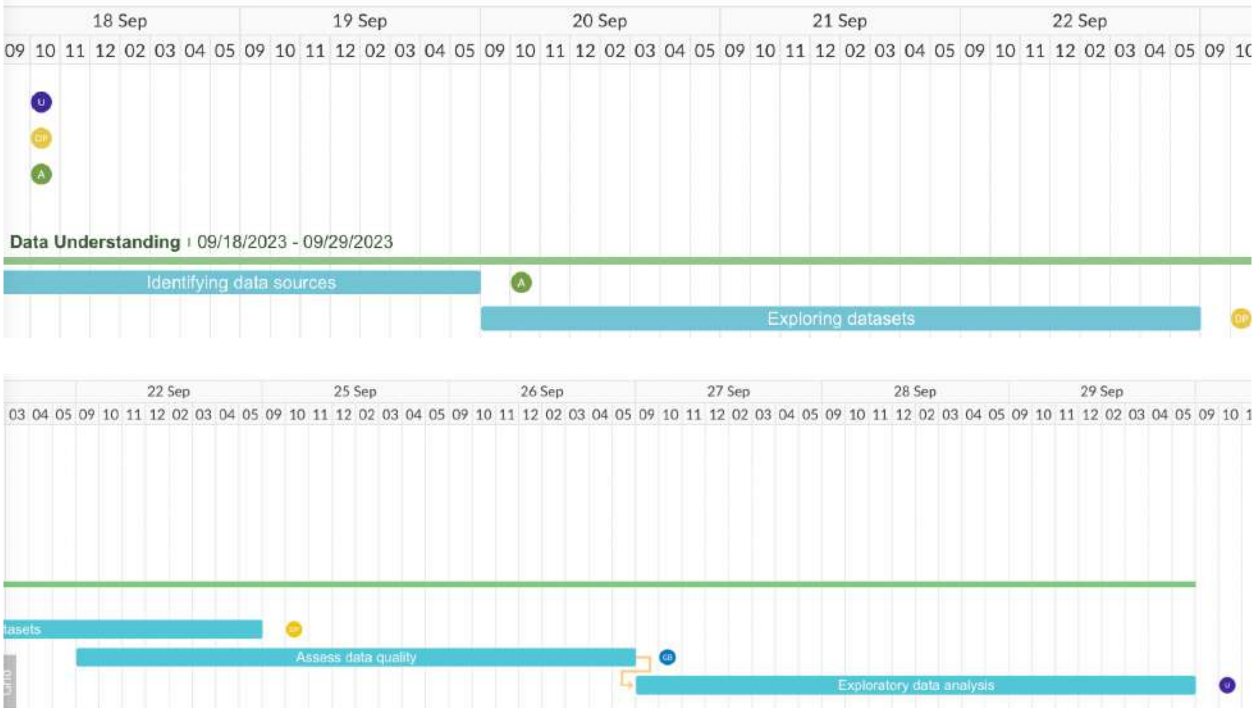


Figure 3

Gantt chart - Data preparation

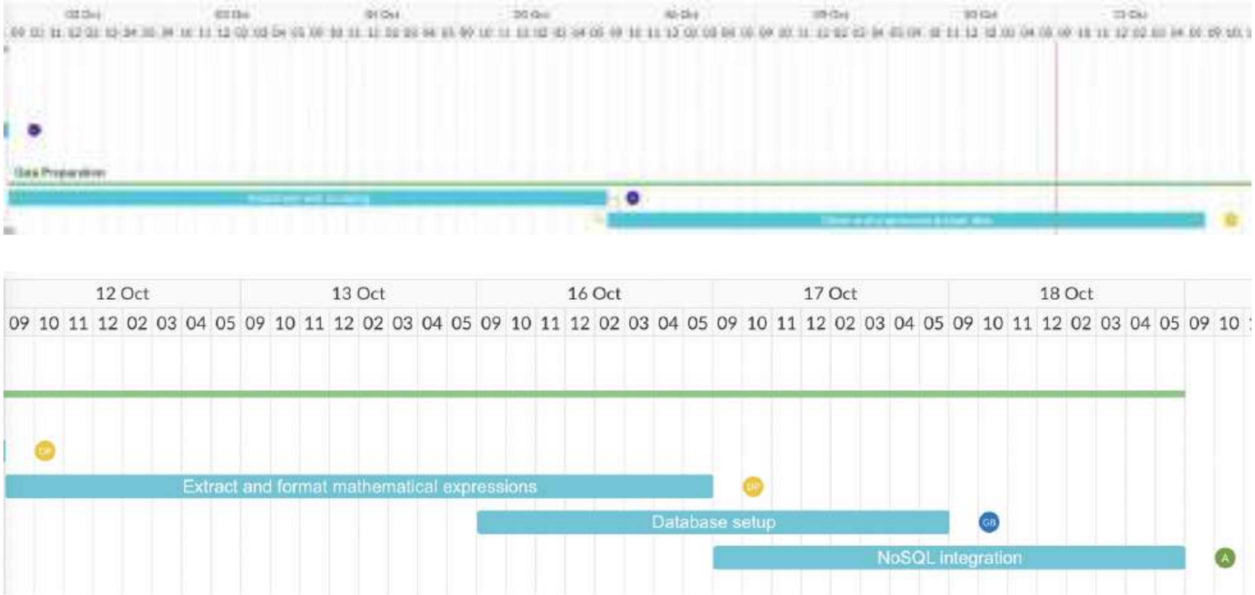
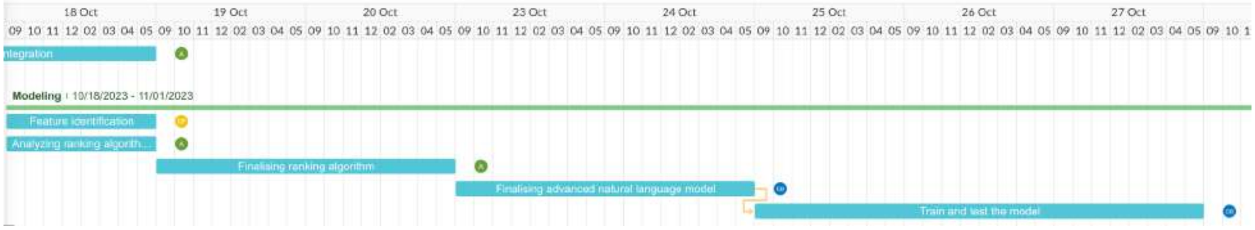


Figure 4

Gantt chart - Modeling



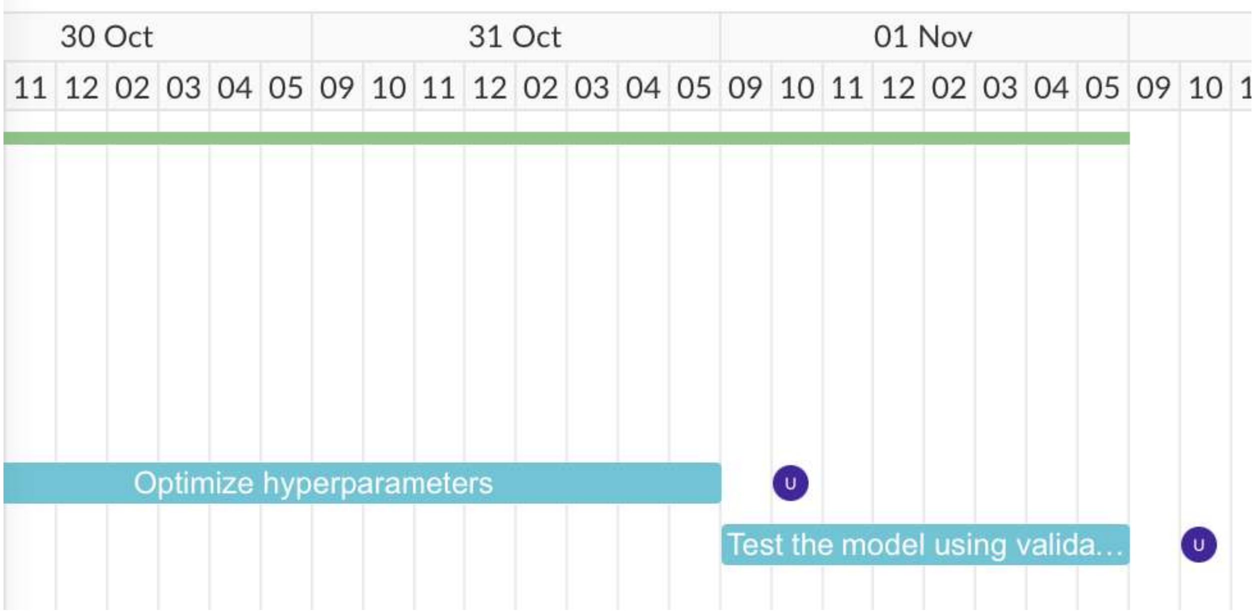
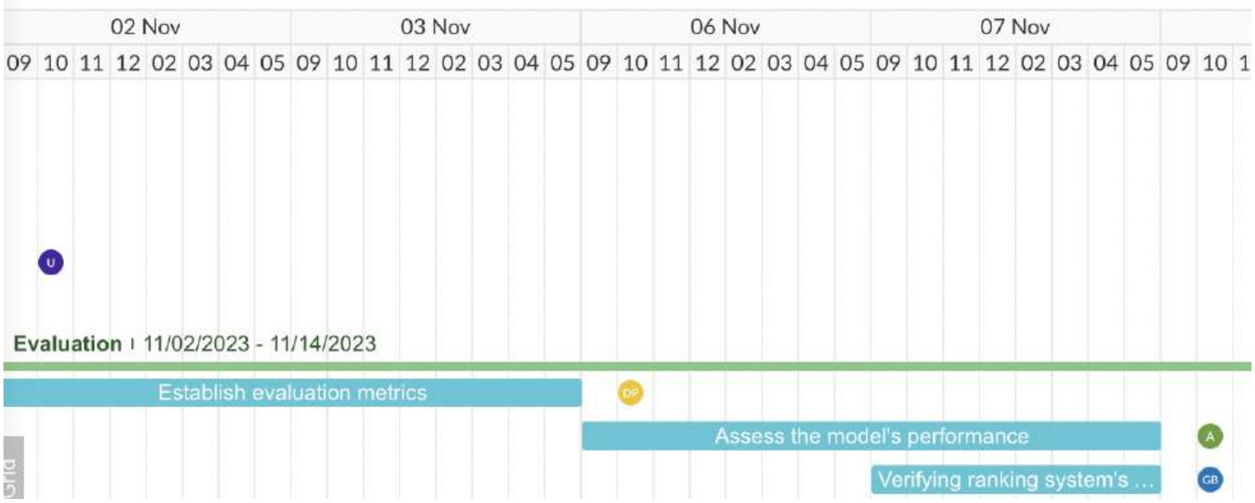


Figure 5

Gantt chart - Evaluation



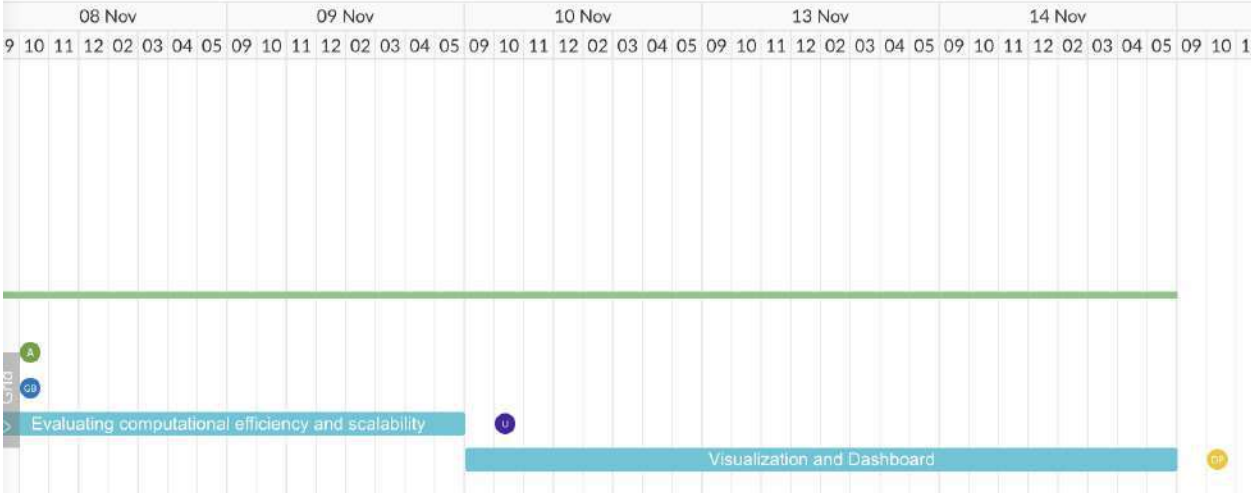
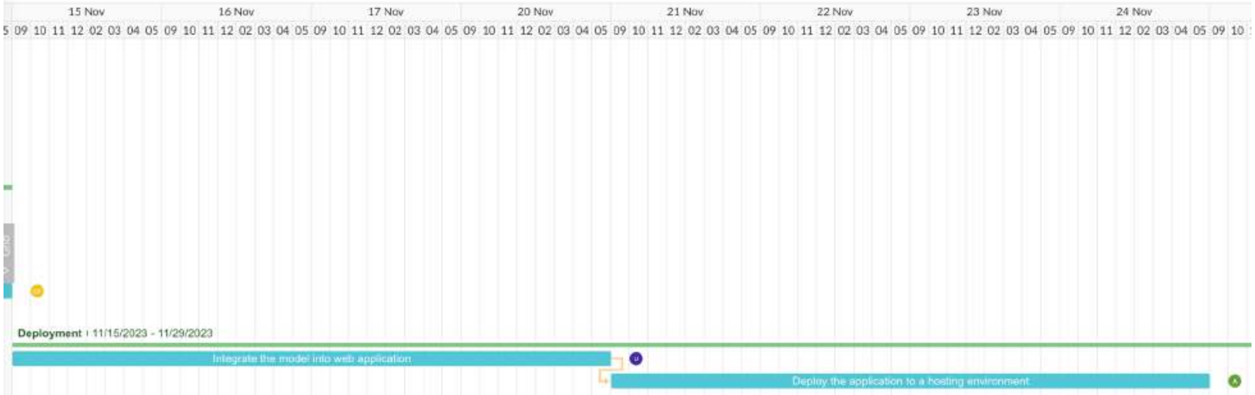


Figure 6

Gantt chart - Deployment



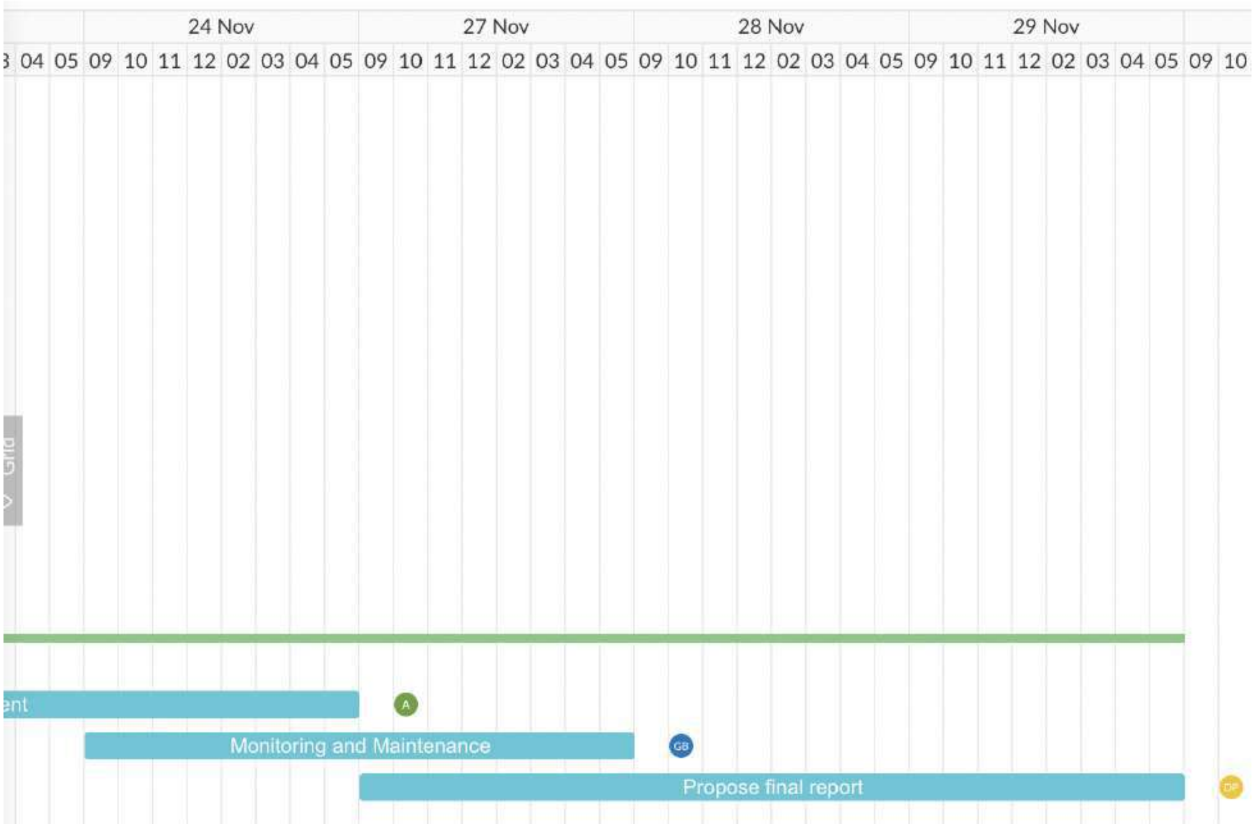
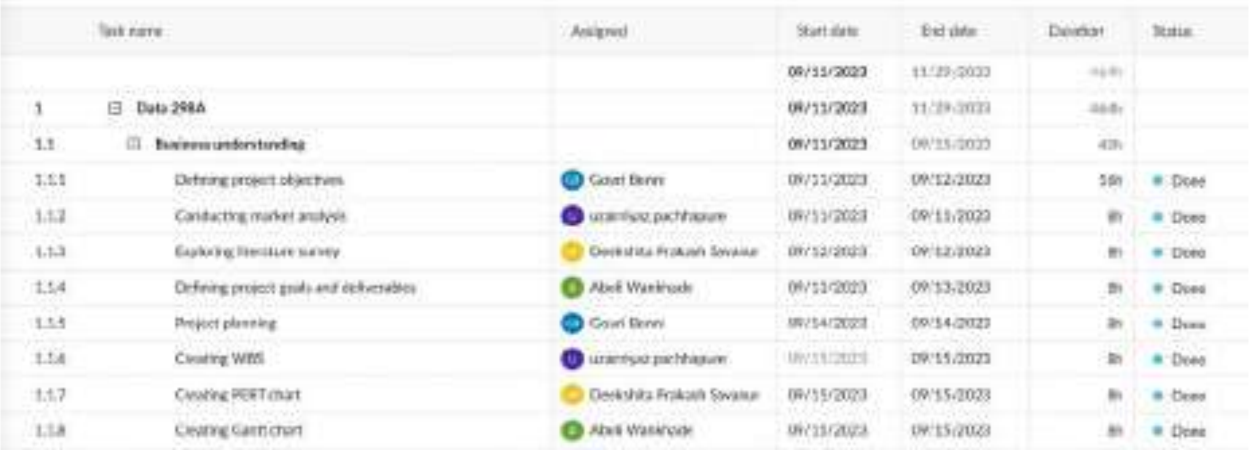


Figure 7

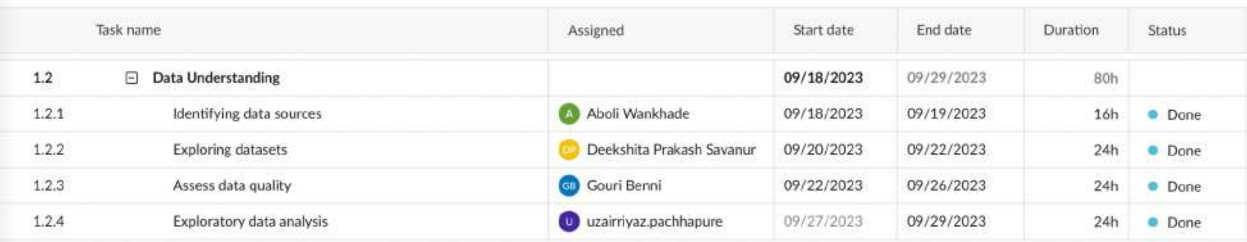
Effort Estimate - Business understanding



	Task name	Assigned	Start date	End date	Duration	Status
			09/11/2023	11/29/2023	464h	
1	⊟ Data 298A		09/11/2023	11/29/2023	464h	
1.1	⊟ Business understanding		09/11/2023	09/15/2023	43h	
1.1.1	Defining project objectives	Gouri Benni	09/11/2023	09/12/2023	16h	● Done
1.1.2	Conducting market analysis	uzamiyaz pachhapure	09/11/2023	09/11/2023	8h	● Done
1.1.3	Exploring literature survey	Deekshita Prakash Sawalur	09/12/2023	09/12/2023	8h	● Done
1.1.4	Defining project goals and deliverables	Abeli Wankhade	09/13/2023	09/13/2023	8h	● Done
1.1.5	Project planning	Gouri Benni	09/14/2023	09/14/2023	8h	● Done
1.1.6	Creating WBS	uzamiyaz pachhapure	09/15/2023	09/15/2023	8h	● Done
1.1.7	Creating PERT chart	Deekshita Prakash Sawalur	09/15/2023	09/15/2023	8h	● Done
1.1.8	Creating Gantt chart	Abeli Wankhade	09/15/2023	09/15/2023	8h	● Done

Figure 8

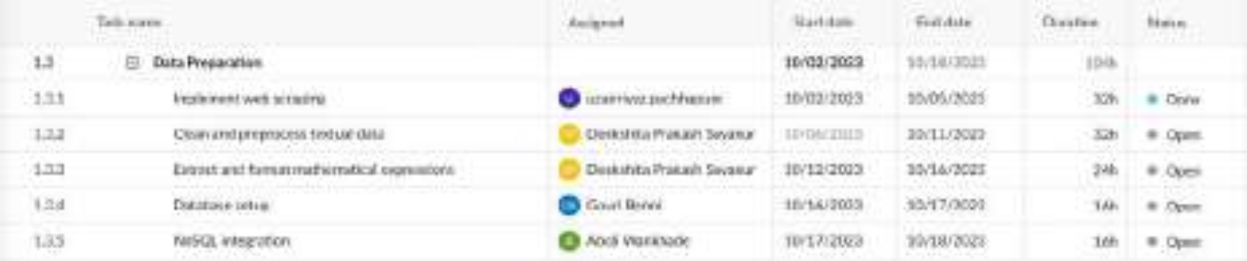
Effort Estimate - Data understanding



	Task name	Assigned	Start date	End date	Duration	Status
1.2	⊟ **Data Understanding**		**09/18/2023**	09/29/2023	80h	
1.2.1	Identifying data sources	A Aboli Wankhade	09/18/2023	09/19/2023	16h	● Done
1.2.2	Exploring datasets	DP Deekshita Prakash Savanur	09/20/2023	09/22/2023	24h	● Done
1.2.3	Assess data quality	GB Gouri Benni	09/22/2023	09/26/2023	24h	● Done
1.2.4	Exploratory data analysis	U uzairriyaz.pachhapure	09/27/2023	09/29/2023	24h	● Done

Figure 9

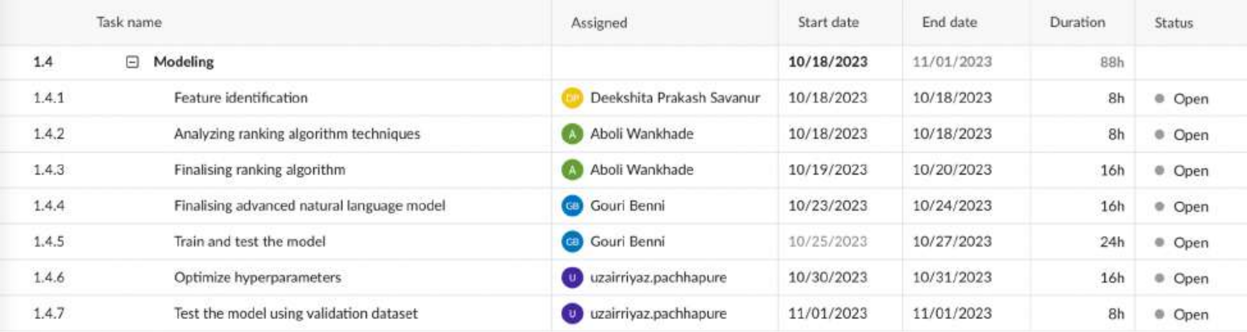
Effort Estimate - Data preparation



	Task name	Assigned	Start date	End date	Duration	Status
1.3	⊟ **Data Preparation**		**10/02/2023**	10/18/2023	104h	
1.3.1	Implement web scraping	U uzairriyaz.pachhapure	10/02/2023	10/05/2023	32h	● Done
1.3.2	Clean and preprocess textual data	DP Deekshita Prakash Savanur	10/06/2023	10/11/2023	32h	● Open
1.3.3	Extract and format mathematical expressions	DP Deekshita Prakash Savanur	10/12/2023	10/16/2023	24h	● Open
1.3.4	Database setup	GB Gouri Benni	10/14/2023	10/17/2023	16h	● Open
1.3.5	NoSQL integration	A Aboli Wankhade	10/17/2023	10/18/2023	16h	● Open

Figure 10

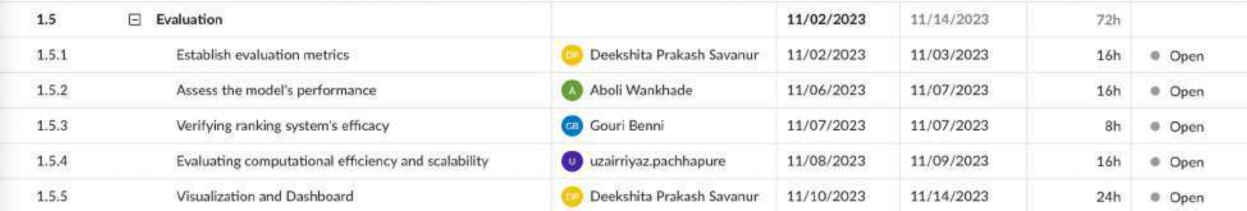
Effort Estimate - Modeling



	Task name	Assigned	Start date	End date	Duration	Status
1.4	⊟ **Modeling**		**10/18/2023**	11/01/2023	88h	
1.4.1	Feature identification	DP Deekshita Prakash Savanur	10/18/2023	10/18/2023	8h	● Open
1.4.2	Analyzing ranking algorithm techniques	A Aboli Wankhade	10/18/2023	10/18/2023	8h	● Open
1.4.3	Finalising ranking algorithm	A Aboli Wankhade	10/19/2023	10/20/2023	16h	● Open
1.4.4	Finalising advanced natural language model	GB Gouri Benni	10/23/2023	10/24/2023	16h	● Open
1.4.5	Train and test the model	GB Gouri Benni	10/25/2023	10/27/2023	24h	● Open
1.4.6	Optimize hyperparameters	U uzairriyaz.pachhapure	10/30/2023	10/31/2023	16h	● Open
1.4.7	Test the model using validation dataset	U uzairriyaz.pachhapure	11/01/2023	11/01/2023	8h	● Open

Figure 11

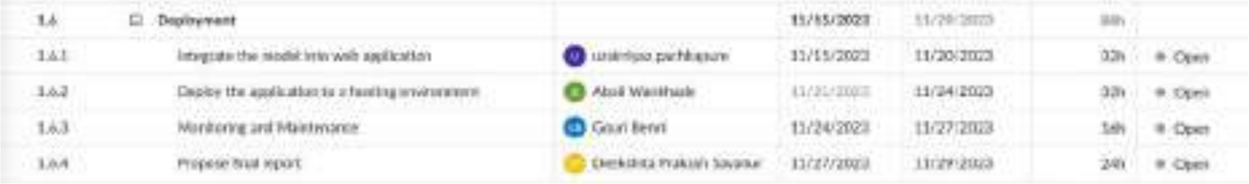
Effort Estimate - Evaluation



1.5	⊟ **Evaluation**		**11/02/2023**	11/14/2023	72h	
1.5.1	Establish evaluation metrics	DP Deekshita Prakash Savanur	11/02/2023	11/03/2023	16h	● Open
1.5.2	Assess the model's performance	A Aboli Wankhade	11/06/2023	11/07/2023	16h	● Open
1.5.3	Verifying ranking system's efficacy	GB Gouri Benni	11/07/2023	11/07/2023	8h	● Open
1.5.4	Evaluating computational efficiency and scalability	U uzairriyaz.pachhapure	11/08/2023	11/09/2023	16h	● Open
1.5.5	Visualization and Dashboard	DP Deekshita Prakash Savanur	11/10/2023	11/14/2023	24h	● Open

Figure 12

Effort Estimate - Deployment



1.6	□ Deployment		11/15/2023	11/29/2023	88h	
1.6.1	Integrate the model into web application	uraimpa parhkapure	11/15/2023	11/20/2023	32h	● Open
1.6.2	Deploy the application to a hosting environment	Aboli Wanthade	11/21/2023	11/24/2023	32h	● Open
1.6.3	Monitoring and Maintenance	Gouri Benni	11/24/2023	11/27/2023	16h	● Open
1.6.4	Propose final report	Deekshita Prakash Savarur	11/27/2023	11/29/2023	24h	● Open

PERT Chart

PERT (Project Evaluation and Review Technique) charts are a type of visual representation used to show a project's timetable. Its goal is to illustrate the order of the project's tasks, their interdependencies, and the time frames at which they are expected to be completed. The PERT chart's degree of detail was selected to give a clear picture of the project timetable. A PERT chart based on milestones is developed for the project, which precisely depicts the dependencies for all tasks and activities necessary to accurately portray the project timeline. The project's job sequence, expected completion times, and interdependencies are the main topics of the chart. The following is the project's critical path: Start - Determining Project Objectives - Data Collection & Aggregation - Web Scraping - API Retrieval - Data Processing & Cleaning - Data Storage - Data Setup - NoSQL Integration - Ranking Algorithm Development - Feature Identification - Model Training - UI/UX Development - Functionality Development - Performance Measurement & Evaluation - Project Deployment - End. The project completion path with the most ideal and required tasks has been recognized as the critical path. Based on prior experience, the number of hopeful days for each work was assigned. The milestone-based PERT chart presents an accurate depiction of job interdependence, a clear overview of the project timeline, and a crucial path for project completion.

Table 6

Dependency list in PERT Chart

Activity	Description	Predecessor/Dependency
1	Project Start	-
2	Determining Project Objectives	-
3	Data Collection & Aggregation	2
4	Web Scraping	3
5	API Retrieval	3,4
6	Data Processing & Cleaning	5
7	Data Storage	6
8	Data Setup	7
9	NoSQL Integration	8
10	Ranking Algorithm Development	9
11	Feature Identification	10
12	Model Training	10,11
13	UI/UX Development	-
14	Functionality Development	11,12,13
15	Performance Measurement & Evaluation	14

16	Project Deployment	15
17	Project End	16

Figure 13

PERT Chart

Figure 14

Closer view of PERT Chart - 1

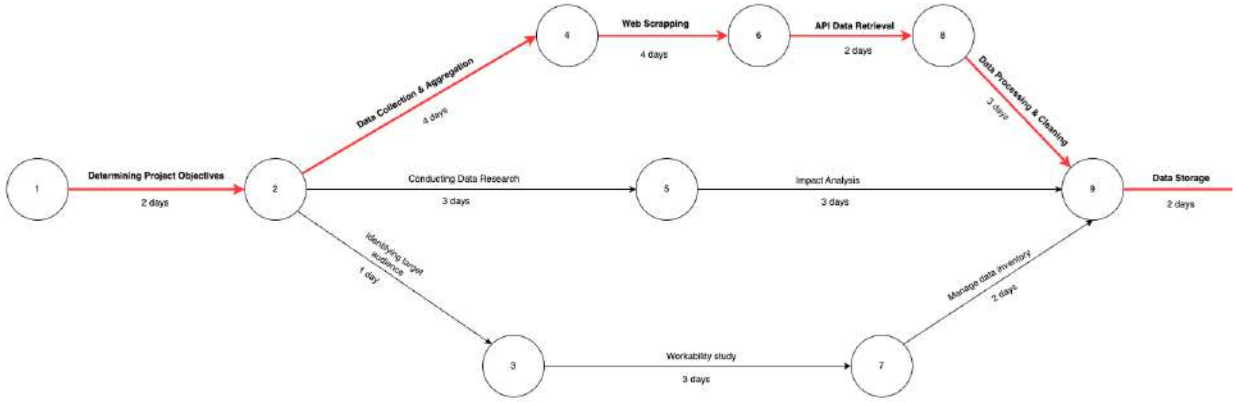


Figure 15

Closer view of PERT Chart - 2

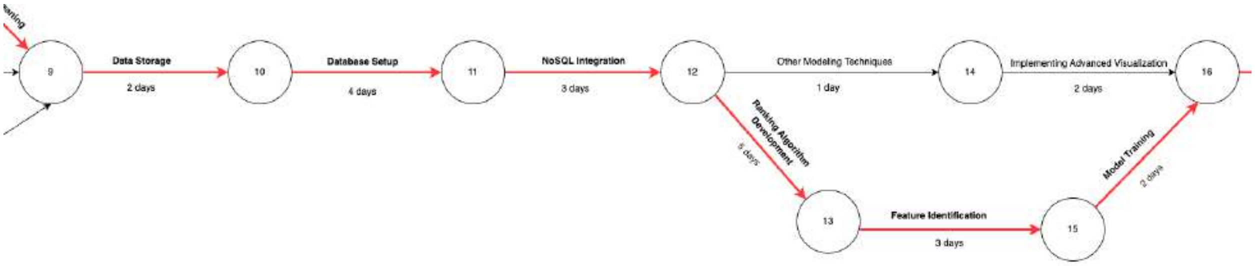
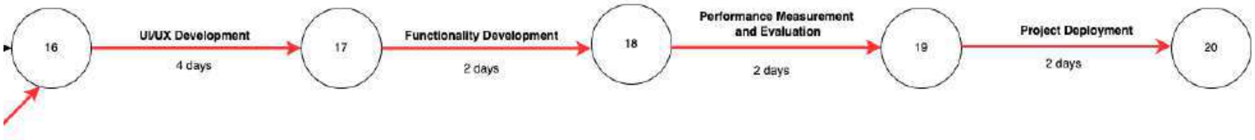


Figure 16

Closer view of PERT Chart - 3



Data Engineering

3.1 Data Process

The enhanced Stack Overflow system necessitates a meticulous approach to data handling. The primary step encompasses gathering data predominantly from Mathematics Stack Exchange, acclaimed for its extensive repository of questions and answers in the domains of mathematics and statistics. This is received from the Internet Archive website. Other credible sites in related fields augment this dataset, ensuring a richer breadth of information.

Post-acquisition, the focus is shifted to inputting this assembled data into the system. This rich dataset, laden with user scores, solutions, upvotes, downvotes, and other pivotal metrics, forms the analytical bedrock. Various models were used in predicting the tags for the user queries. Utilizing Hugging Face's wide-ranging NLP tools, this data undergoes refinement to rank the best answers. Redundancies are pruned, gaps are filled, and uniformity is enforced, paving the way for a seamless, consistent dataset.

With optimized data in place, deeper analytical pursuits commence to identify latent patterns and trends that could gauge solution efficacy. Informed by these insights, machine learning models, especially ranking algorithms, are employed. These models adeptly capture the nuanced semantics and broader contexts of solutions. For validation and robustness, the dataset is apportioned, with 70% earmarked for training and the remainder equally divided between validation and testing.

Concluding this rigorous data journey, diverse visualization tools are harnessed to illustrate the findings. Through graphs and charts, a comprehensive summary is offered, thereby refining the system and elevating the quality of responses to user queries.

3.2. Data Collection

The datasets were collected from the Internet Archive, a nonprofit organization that was established in 1996 with the goal of preserving and making accessible a sizable collection of digital content. A vital tool for historical and research reasons, it archives and makes accessible a variety of media, including webpages, books, music, and movies. It allows users to view archived web pages and follow the development of the internet over time with programmes like the Wayback Machine.

Users Dataset

The Users dataset was collected and combined from the Mathematical Stack Exchange section, that contains details about the user profile. It is made up of 1215674 rows and 9 columns. Each User data has a unique individual identification (ID) that differentiates between various users. The dataset also contains user details like creation date through 2012, Display name, Last Access Date and other details. Table 7 shows a data collection plan for the Users table.

Figure 17

Samples from raw Users dataset

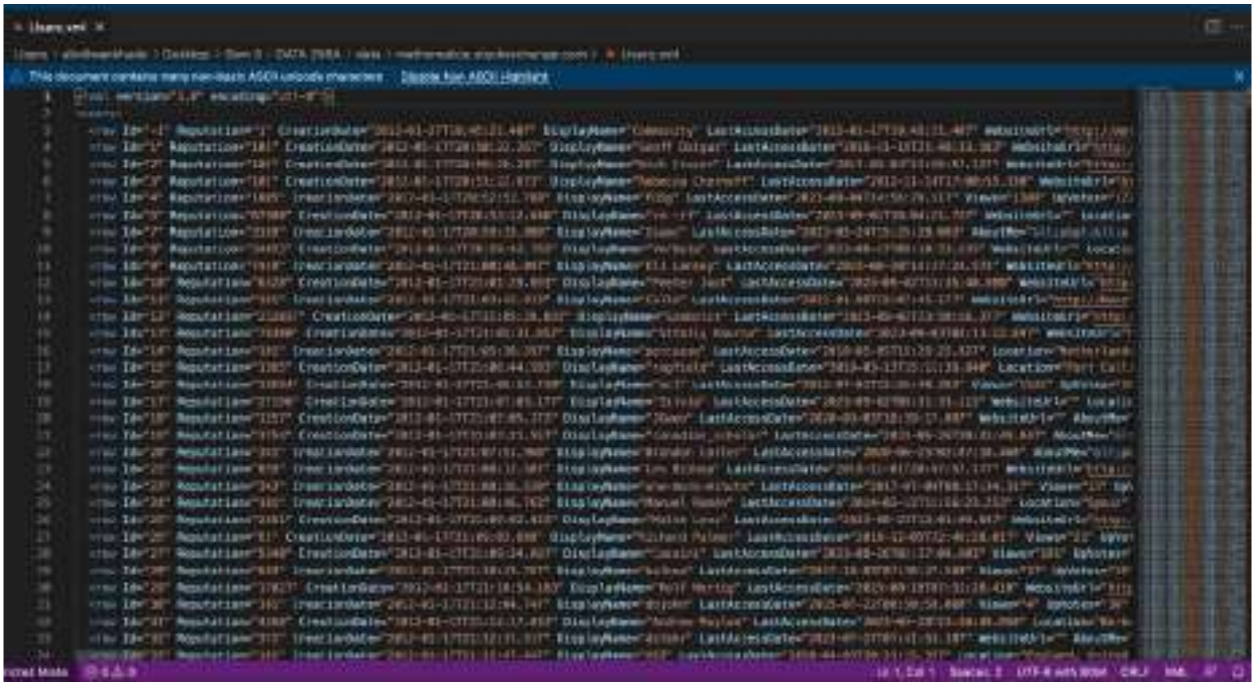
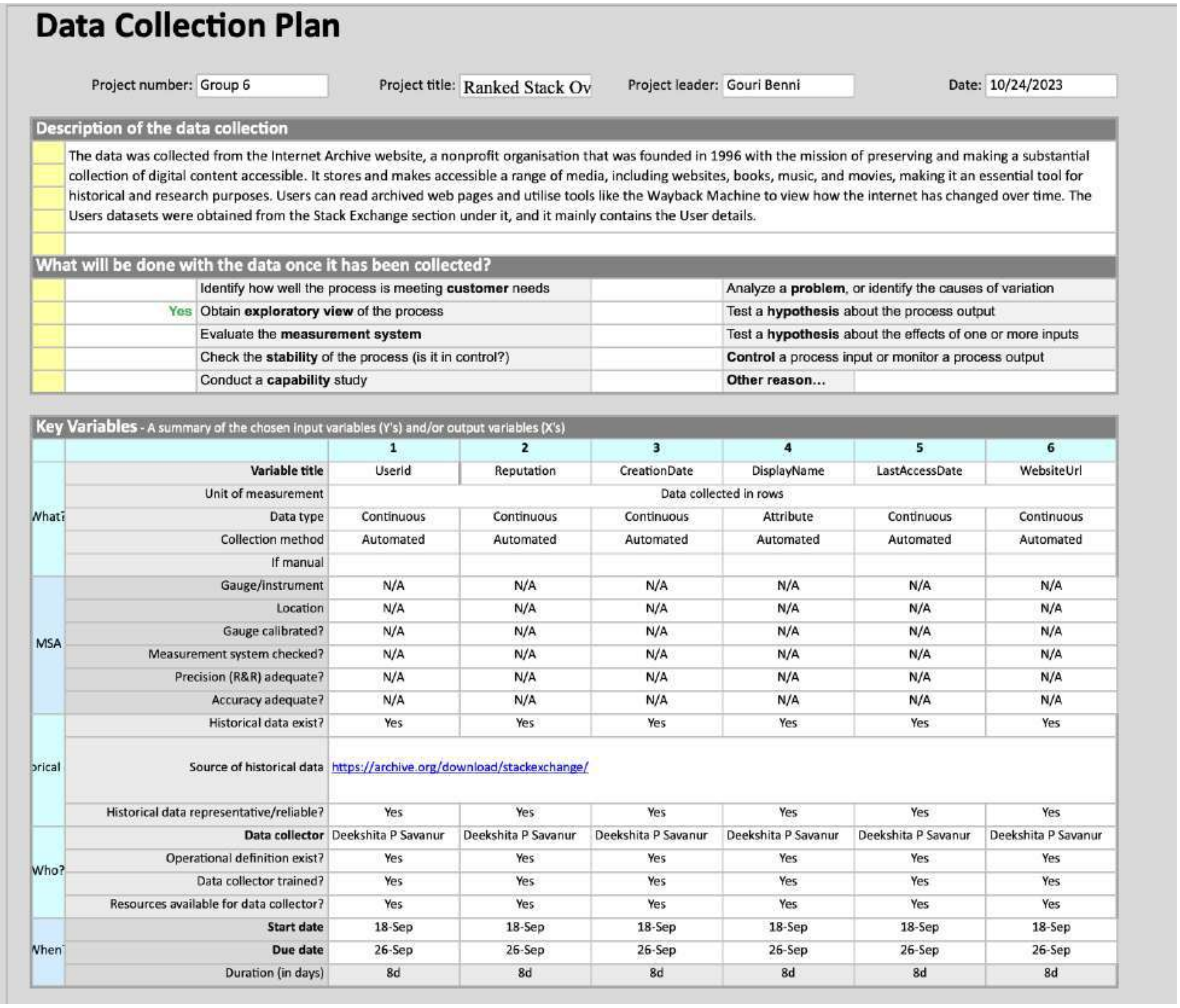


Table 7

Data Collection Plan



Data Collection Plan

Project number: Group 6 | Project title: Ranked Stack Ov | Project leader: Gouri Benni | Date: 10/24/2023

Description of the data collection

The data was collected from the Internet Archive website, a nonprofit organisation that was founded in 1996 with the mission of preserving and making a substantial collection of digital content accessible. It stores and makes accessible a range of media, including websites, books, music, and movies, making it an essential tool for historical and research purposes. Users can read archived web pages and utilise tools like the Wayback Machine to view how the internet has changed over time. The Users datasets were obtained from the Stack Exchange section under it, and it mainly contains the User details.

What will be done with the data once it has been collected?

	Identify how well the process is meeting **customer** needs		Analyze a **problem**, or identify the causes of variation
Yes	Obtain **exploratory view** of the process		Test a **hypothesis** about the process output
	Evaluate the **measurement system**		Test a **hypothesis** about the effects of one or more inputs
	Check the **stability** of the process (is it in control?)		**Control** a process input or monitor a process output
	Conduct a **capability** study		**Other reason...**

Key Variables - A summary of the chosen input variables (Y's) and/or output variables (X's)

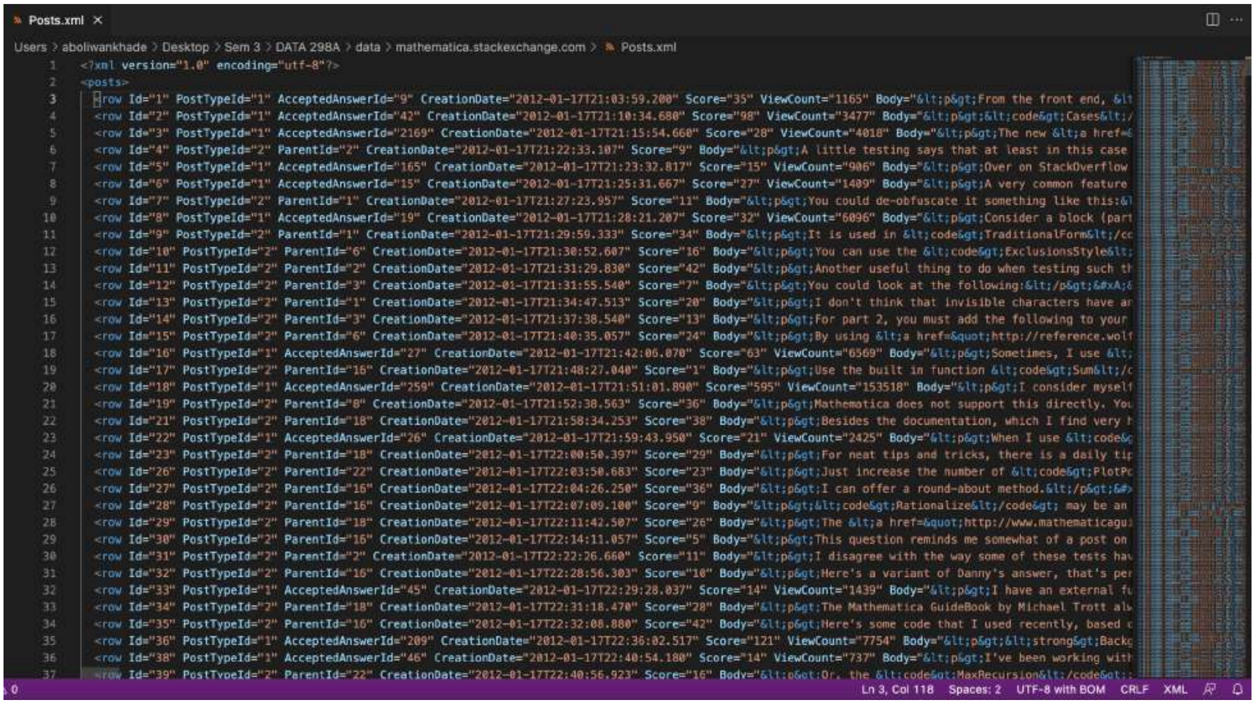
		1	2	3	4	5	6
What?	**Variable title**	UserId	Reputation	CreationDate	DisplayName	LastAccessDate	WebsiteUrl
	Unit of measurement	Data collected in rows					
	Data type	Continuous	Continuous	Continuous	Attribute	Continuous	Continuous
	Collection method	Automated	Automated	Automated	Automated	Automated	Automated
	If manual						
MSA	Gauge/instrument	N/A	N/A	N/A	N/A	N/A	N/A
	Location	N/A	N/A	N/A	N/A	N/A	N/A
	Gauge calibrated?	N/A	N/A	N/A	N/A	N/A	N/A
	Measurement system checked?	N/A	N/A	N/A	N/A	N/A	N/A
	Precision (R&R) adequate?	N/A	N/A	N/A	N/A	N/A	N/A
	Accuracy adequate?	N/A	N/A	N/A	N/A	N/A	N/A
orical	Historical data exist?	Yes	Yes	Yes	Yes	Yes	Yes
	Source of historical data	https://archive.org/download/stackexchange/					
	Historical data representative/reliable?	Yes	Yes	Yes	Yes	Yes	Yes
Who?	**Data collector**	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur
	Operational definition exist?	Yes	Yes	Yes	Yes	Yes	Yes
	Data collector trained?	Yes	Yes	Yes	Yes	Yes	Yes
	Resources available for data collector?	Yes	Yes	Yes	Yes	Yes	Yes
When	**Start date**	18-Sep	18-Sep	18-Sep	18-Sep	18-Sep	18-Sep
	Due date	26-Sep	26-Sep	26-Sep	26-Sep	26-Sep	26-Sep
	Duration (in days)	8d	8d	8d	8d	8d	8d

Posts Dataset

The Posts dataset was collected and combined from the Mathematical Stack Exchange, MathOverflow, Math.educators and Math.StackExchange section, which contains details about the Posts written by users. It is made up of 3708430 rows and 17 columns. Each Post is differentiated by another using a unique individual identification (ID). It contains other details such as Post Creation Date, Score, View Count and other details. Table 8 shows a data collection plan for the Posts table.

Figure 18

Samples from raw posts dataset

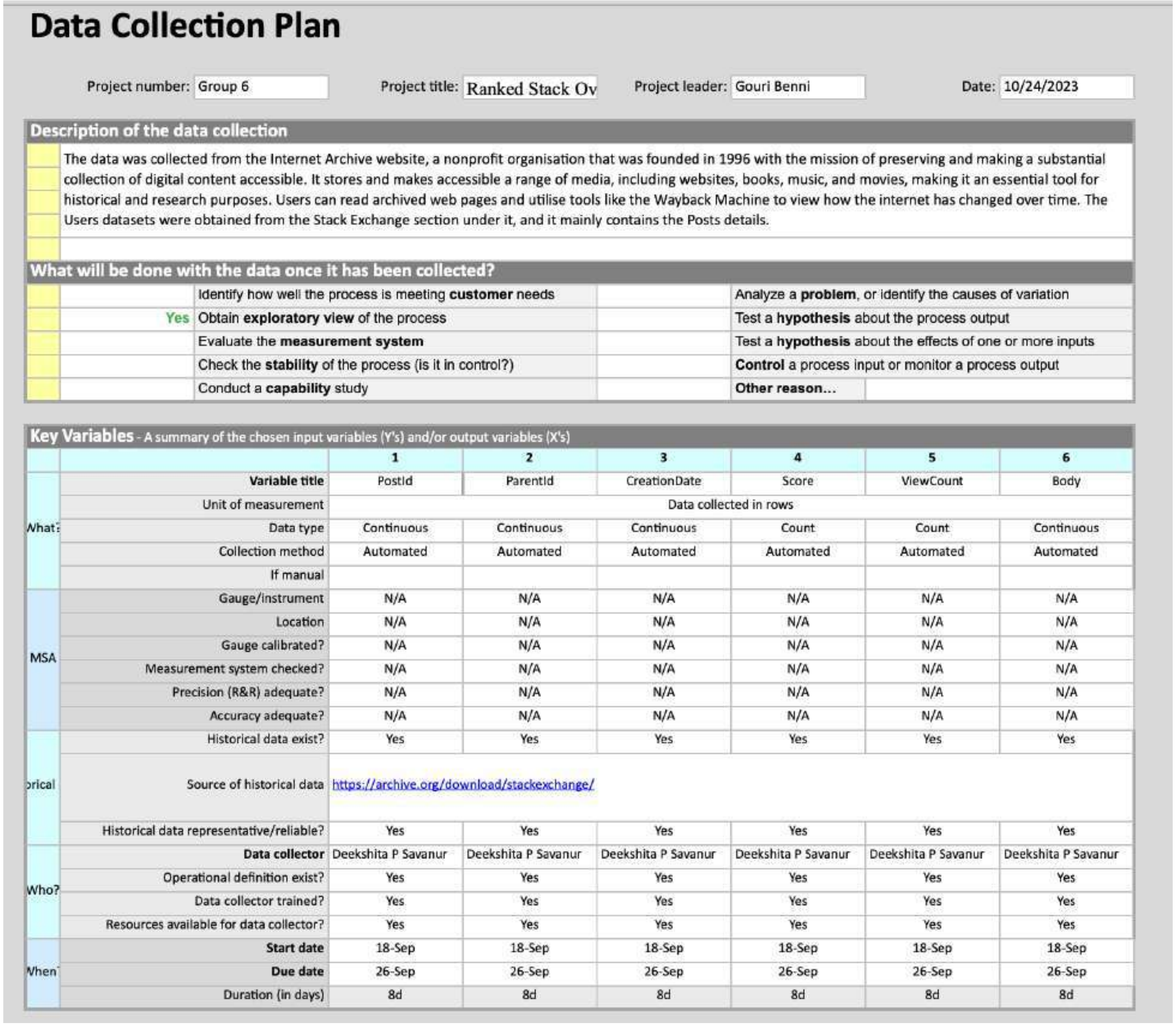


Users > aboliwankhade > Desktop > Sem 3 > DATA 298A > data > mathematica.stackexchange.com > Posts.xml

```
<?xml version="1.0" encoding="utf-8"?>
<posts>
  <row Id="1" PostTypeId="1" AcceptedAnswerId="9" CreationDate="2012-01-17T21:03:59.200" Score="35" ViewCount="1165" Body="&lt;p&gt;From the front end, &lt
  <row Id="2" PostTypeId="1" AcceptedAnswerId="42" CreationDate="2012-01-17T21:10:34.680" Score="98" ViewCount="3477" Body="&lt;p&gt;&lt;code&gt;Cases&lt;/
  <row Id="3" PostTypeId="1" AcceptedAnswerId="2169" CreationDate="2012-01-17T21:15:54.660" Score="28" ViewCount="4018" Body="&lt;p&gt;The new &lt;a href=&
  <row Id="4" PostTypeId="2" ParentId="2" CreationDate="2012-01-17T21:22:33.107" Score="9" Body="&lt;p&gt;A little testing says that at least in this case
  <row Id="5" PostTypeId="1" AcceptedAnswerId="165" CreationDate="2012-01-17T21:23:32.817" Score="15" ViewCount="906" Body="&lt;p&gt;Over on StackOverflow
  <row Id="6" PostTypeId="1" AcceptedAnswerId="15" CreationDate="2012-01-17T21:25:31.667" Score="27" ViewCount="1409" Body="&lt;p&gt;A very common feature
  <row Id="7" PostTypeId="2" ParentId="1" CreationDate="2012-01-17T21:27:23.957" Score="11" Body="&lt;p&gt;You could de-obfuscate it something like this:&l
  <row Id="8" PostTypeId="1" AcceptedAnswerId="19" CreationDate="2012-01-17T21:28:21.207" Score="32" ViewCount="6096" Body="&lt;p&gt;Consider a block (part
  <row Id="9" PostTypeId="2" ParentId="1" CreationDate="2012-01-17T21:29:59.333" Score="34" Body="&lt;p&gt;It is used in &lt;code&gt;TraditionalForm&lt;/co
  <row Id="10" PostTypeId="2" ParentId="6" CreationDate="2012-01-17T21:30:52.607" Score="16" Body="&lt;p&gt;You can use the &lt;code&gt;ExclusionsStyle&lt;
  <row Id="11" PostTypeId="2" ParentId="2" CreationDate="2012-01-17T21:31:29.830" Score="42" Body="&lt;p&gt;Another useful thing to do when testing such th
  <row Id="12" PostTypeId="2" ParentId="3" CreationDate="2012-01-17T21:31:55.540" Score="7" Body="&lt;p&gt;You could look at the following:&lt;/p&gt;&#xA;&
  <row Id="13" PostTypeId="2" ParentId="1" CreationDate="2012-01-17T21:34:47.513" Score="20" Body="&lt;p&gt;I don't think that invisible characters have ar
  <row Id="14" PostTypeId="2" ParentId="3" CreationDate="2012-01-17T21:37:38.540" Score="13" Body="&lt;p&gt;For part 2, you must add the following to your
  <row Id="15" PostTypeId="2" ParentId="6" CreationDate="2012-01-17T21:40:35.057" Score="24" Body="&lt;p&gt;By using &lt;a href=&quot;http://reference.wolf
  <row Id="16" PostTypeId="1" AcceptedAnswerId="27" CreationDate="2012-01-17T21:42:06.070" Score="63" ViewCount="6569" Body="&lt;p&gt;Sometimes, I use &lt;
  <row Id="17" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T21:48:27.040" Score="1" Body="&lt;p&gt;Use the built in function &lt;code&gt;Sum&lt;/c
  <row Id="18" PostTypeId="1" AcceptedAnswerId="259" CreationDate="2012-01-17T21:51:01.890" Score="595" ViewCount="153518" Body="&lt;p&gt;I consider myself
  <row Id="19" PostTypeId="2" ParentId="8" CreationDate="2012-01-17T21:52:38.563" Score="36" Body="&lt;p&gt;Mathematica does not support this directly. You
  <row Id="21" PostTypeId="2" ParentId="18" CreationDate="2012-01-17T21:58:34.253" Score="38" Body="&lt;p&gt;Besides the documentation, which I find very h
  <row Id="22" PostTypeId="1" AcceptedAnswerId="26" CreationDate="2012-01-17T21:59:43.950" Score="21" ViewCount="2425" Body="&lt;p&gt;When I use &lt;code&g
  <row Id="23" PostTypeId="2" ParentId="18" CreationDate="2012-01-17T22:00:50.397" Score="29" Body="&lt;p&gt;For neat tips and tricks, there is a daily tip
  <row Id="26" PostTypeId="2" ParentId="22" CreationDate="2012-01-17T22:03:50.683" Score="23" Body="&lt;p&gt;Just increase the number of &lt;code&gt;PlotPo
  <row Id="27" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T22:04:26.250" Score="36" Body="&lt;p&gt;I can offer a round-about method.&lt;/p&gt;&#x
  <row Id="28" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T22:07:09.100" Score="9" Body="&lt;p&gt;&lt;code&gt;Rationalize&lt;/code&gt; may be an
  <row Id="29" PostTypeId="2" ParentId="18" CreationDate="2012-01-17T22:11:42.507" Score="26" Body="&lt;p&gt;The &lt;a href=&quot;http://www.mathematicagui
  <row Id="30" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T22:14:11.057" Score="5" Body="&lt;p&gt;This question reminds me somewhat of a post on
  <row Id="31" PostTypeId="2" ParentId="2" CreationDate="2012-01-17T22:22:26.660" Score="11" Body="&lt;p&gt;I disagree with the way some of these tests hav
  <row Id="32" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T22:28:56.303" Score="10" Body="&lt;p&gt;Here's a variant of Danny's answer, that's per
  <row Id="33" PostTypeId="1" AcceptedAnswerId="45" CreationDate="2012-01-17T22:29:28.037" Score="14" ViewCount="1439" Body="&lt;p&gt;I have an external fu
  <row Id="34" PostTypeId="2" ParentId="18" CreationDate="2012-01-17T22:31:18.470" Score="28" Body="&lt;p&gt;The Mathematica GuideBook by Michael Trott alw
  <row Id="35" PostTypeId="2" ParentId="16" CreationDate="2012-01-17T22:32:08.880" Score="42" Body="&lt;p&gt;Here's some code that I used recently, based o
  <row Id="36" PostTypeId="1" AcceptedAnswerId="209" CreationDate="2012-01-17T22:36:02.517" Score="121" ViewCount="7754" Body="&lt;p&gt;&lt;strong&gt;Backg
  <row Id="38" PostTypeId="1" AcceptedAnswerId="46" CreationDate="2012-01-17T22:40:54.180" Score="14" ViewCount="737" Body="&lt;p&gt;I've been working with
  <row Id="39" PostTypeId="2" ParentId="22" CreationDate="2012-01-17T22:40:56.923" Score="16" Body="&lt;p&gt;Or, the &lt;code&gt;MaxRecursion&lt;/code&gt;;
```

Ln 3, Col 118 Spaces: 2 UTF-8 with BOM CRLF XML

Table 8



Data Collection Plan

Project number: Group 6 Project title: Ranked Stack Ov Project leader: Gouri Benni Date: 10/24/2023

Description of the data collection

The data was collected from the Internet Archive website, a nonprofit organisation that was founded in 1996 with the mission of preserving and making a substantial collection of digital content accessible. It stores and makes accessible a range of media, including websites, books, music, and movies, making it an essential tool for historical and research purposes. Users can read archived web pages and utilise tools like the Wayback Machine to view how the internet has changed over time. The Users datasets were obtained from the Stack Exchange section under it, and it mainly contains the Posts details.

What will be done with the data once it has been collected?

	Identify how well the process is meeting **customer** needs		Analyze a **problem**, or identify the causes of variation	
Yes	Obtain **exploratory view** of the process		Test a **hypothesis** about the process output	
	Evaluate the **measurement system**		Test a **hypothesis** about the effects of one or more inputs	
	Check the **stability** of the process (is it in control?)		**Control** a process input or monitor a process output	
	Conduct a **capability** study		**Other reason...**	

Key Variables - A summary of the chosen input variables (Y's) and/or output variables (X's)

		1	2	3	4	5	6
What?	**Variable title**	PostId	ParentId	CreationDate	Score	ViewCount	Body
	Unit of measurement	Data collected in rows					
	Data type	Continuous	Continuous	Continuous	Count	Count	Continuous
	Collection method	Automated	Automated	Automated	Automated	Automated	Automated
	If manual						
MSA	Gauge/instrument	N/A	N/A	N/A	N/A	N/A	N/A
	Location	N/A	N/A	N/A	N/A	N/A	N/A
	Gauge calibrated?	N/A	N/A	N/A	N/A	N/A	N/A
	Measurement system checked?	N/A	N/A	N/A	N/A	N/A	N/A
	Precision (R&R) adequate?	N/A	N/A	N/A	N/A	N/A	N/A
	Accuracy adequate?	N/A	N/A	N/A	N/A	N/A	N/A
orical	Historical data exist?	Yes	Yes	Yes	Yes	Yes	Yes
	Source of historical data	https://archive.org/download/stackexchange/					
	Historical data representative/reliable?	Yes	Yes	Yes	Yes	Yes	Yes
Who?	**Data collector**	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur	Deekshita P Savanur
	Operational definition exist?	Yes	Yes	Yes	Yes	Yes	Yes
	Data collector trained?	Yes	Yes	Yes	Yes	Yes	Yes
	Resources available for data collector?	Yes	Yes	Yes	Yes	Yes	Yes
When	**Start date**	18-Sep	18-Sep	18-Sep	18-Sep	18-Sep	18-Sep
	Due date	26-Sep	26-Sep	26-Sep	26-Sep	26-Sep	26-Sep
	Duration (in days)	8d	8d	8d	8d	8d	8d

3.3. Data Pre-processing

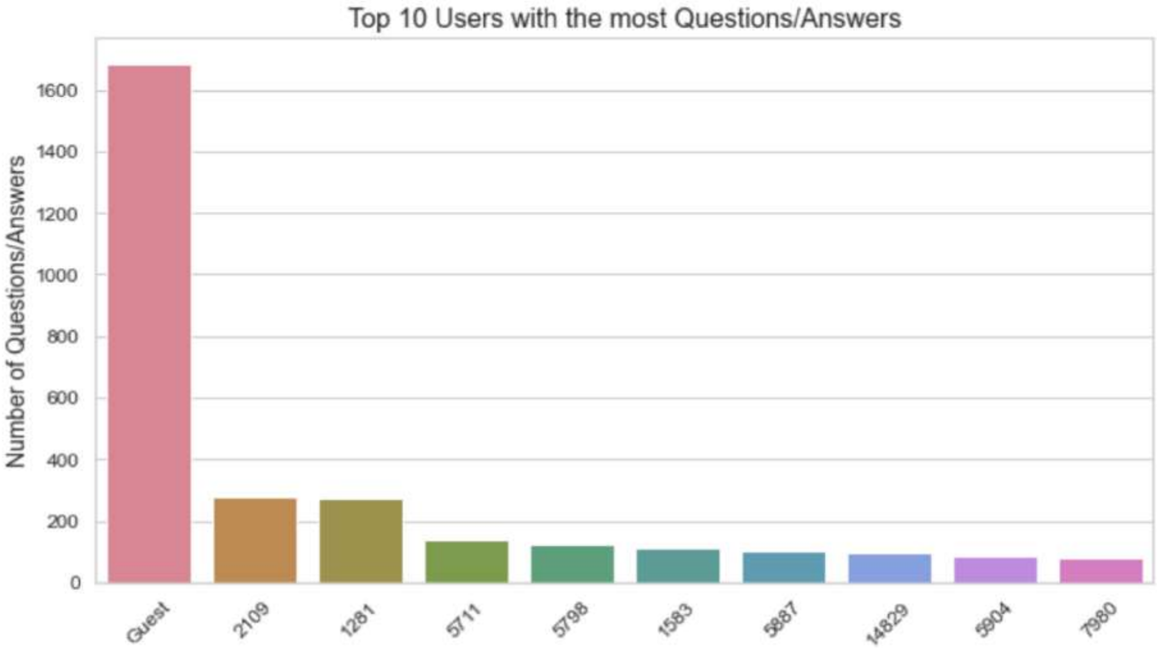
EDA - Data Exploration

EDA involves delving into the intricacies of the dataset, systematically examining patterns, identifying outliers, and uncovering potential insights that can inform the project's objectives. Through EDA, a deeper understanding of the data's distribution, relationships between variables, and any anomalies requiring attention is achieved.

The figure 19 shows the valuable insights into the top 10 users who have actively engaged with the platform by both posing questions and providing answers. Their notable contributions to the community's knowledge-sharing dynamics are prominently featured. For each of these leading users, the figure 19 details the count of Questions they have asked and the count of Answers they have offered. This data not only underscores their involvement but also highlights their pivotal roles as significant contributors to the platform. Examining this allows for the identification of trends and patterns related to user activity in terms of both question posting and answer provision.

Figure 19

Top 10 Users with the most Questions/Answers



The figure 20 presents a comprehensive visualization of the distribution of view counts within the dataset, with a cap set at 5000 for the sake of clarity in presentation. The primary aim of this visualization is to demonstrate the distribution patterns of view counts concerning the number of Questions and Answers. It facilitates the identification of trends and patterns, revealing how view counts are distributed across a number of questions and answers.

Figure 20

Distribution of ViewCounts

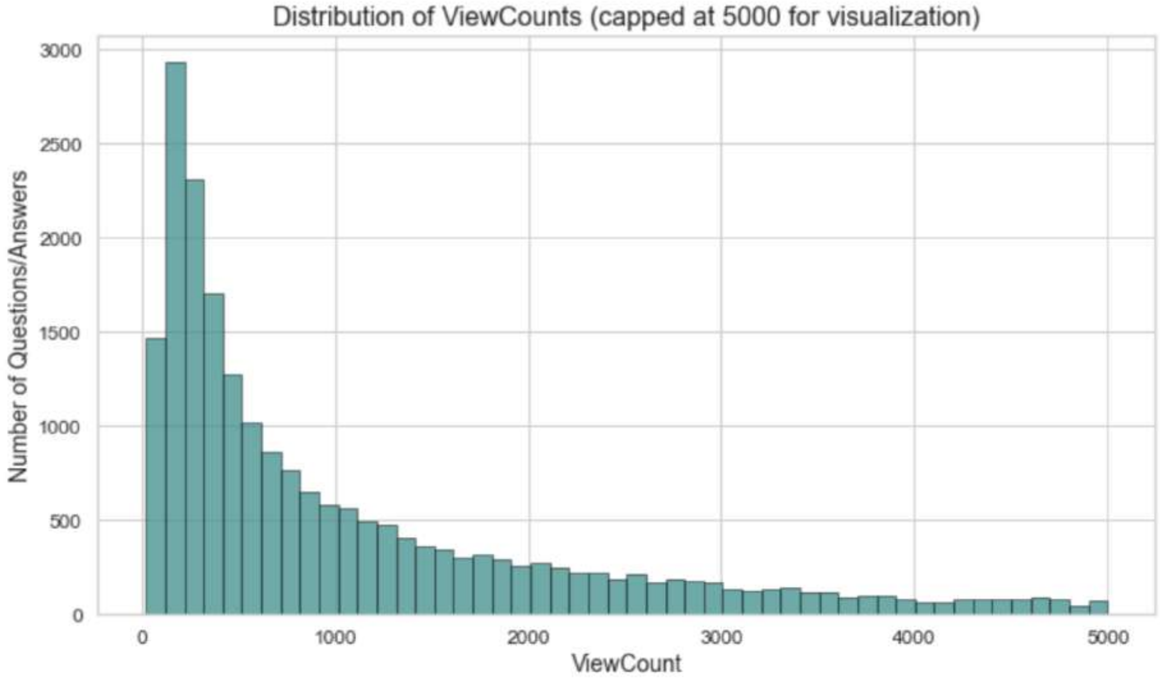
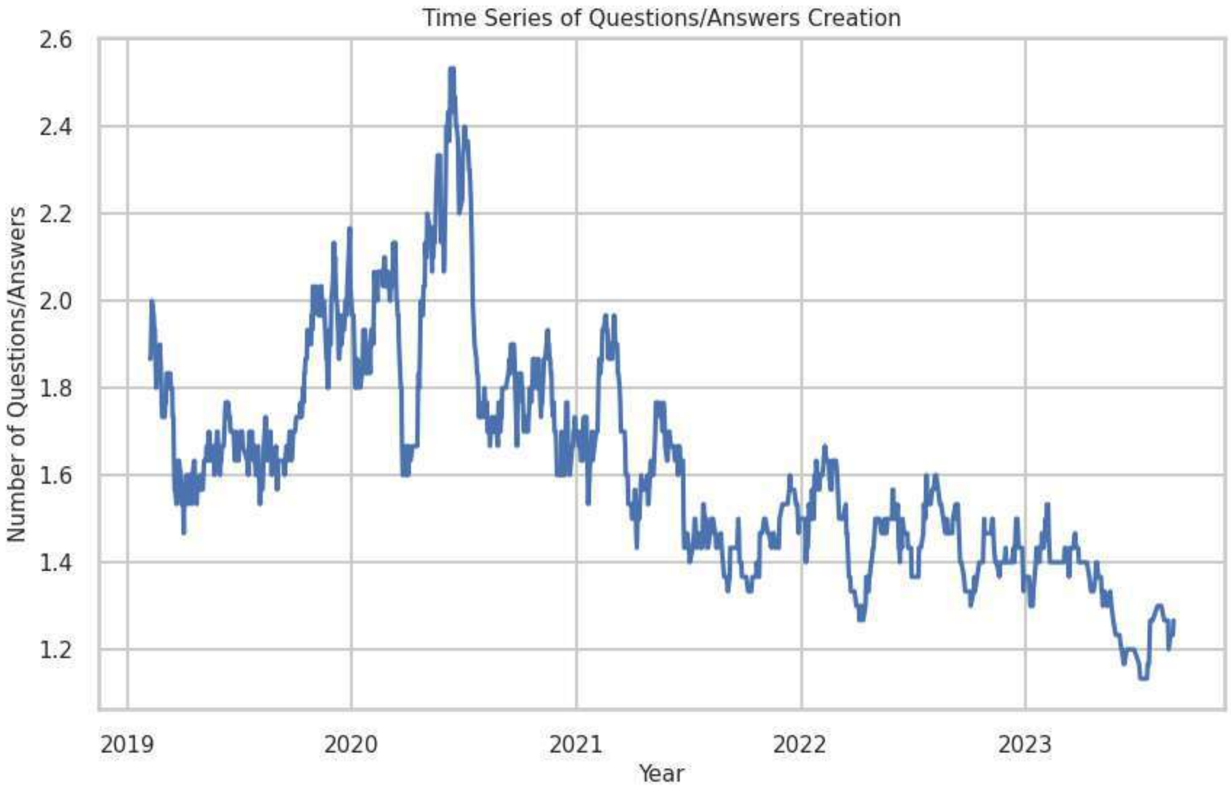


Figure 21 provides a comprehensive view of the time series analysis concerning Questions and Answers creation within the dataset. It effectively visualizes the fluctuations in the counts of Questions and Answers across different dates. By examining this figure, significant trends and patterns in the creation of Questions and Answers over time become apparent. It enables the identification of key dates or time frames when there is a surge in question or answer creation.

Figure 21

Time series of Questions/Answers Creation

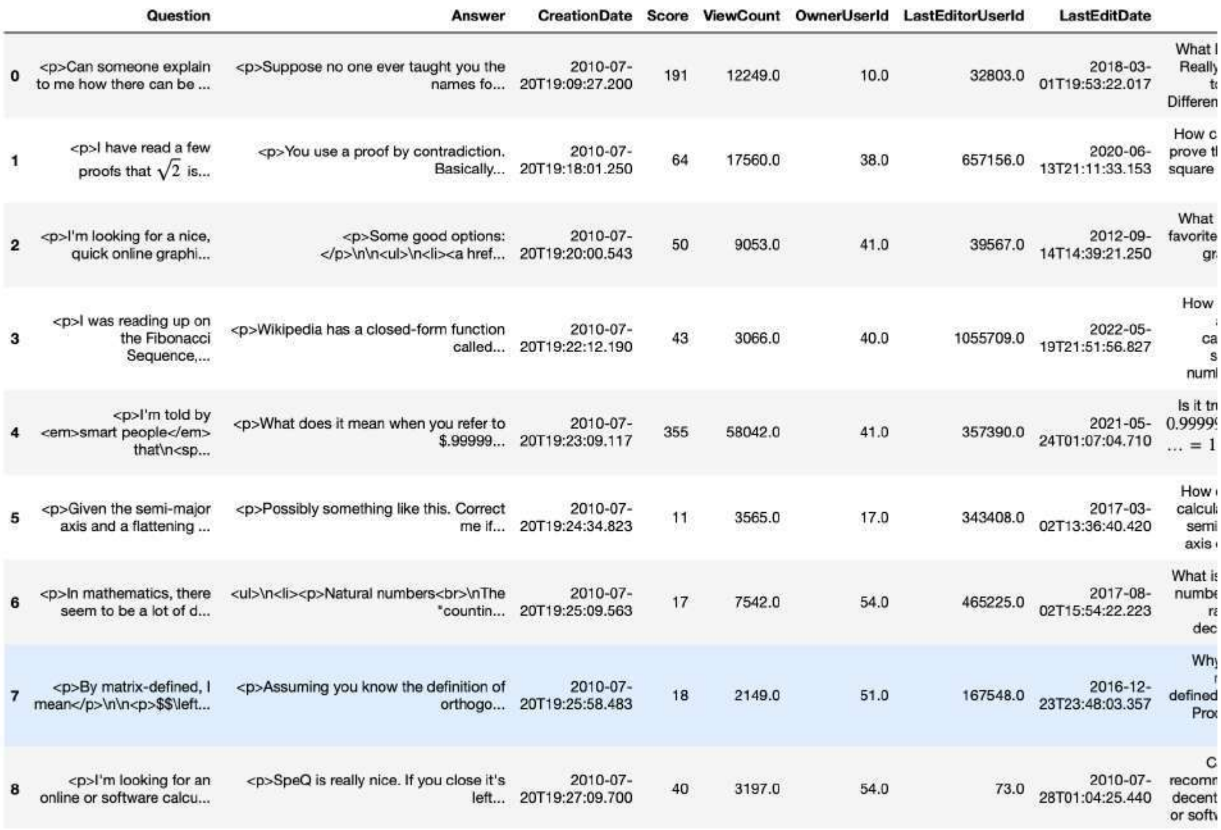


Data cleaning

Removing HTML tags from text data is an essential data cleaning step, especially when working with text from web sources or HTML-rich documents. A python function called “remove_html_tags” was defined, which is used to parse the HTML and extract the text content without the tags.This process ensures that the text data is free of HTML tags, making it more suitable for further natural language processing and analysis tasks. In below figure 22 we can see the presence of HTML tags in columns “Question” and “Answer”. Followed by that is the figure of columns after removal of HTML tags.

Figure 22

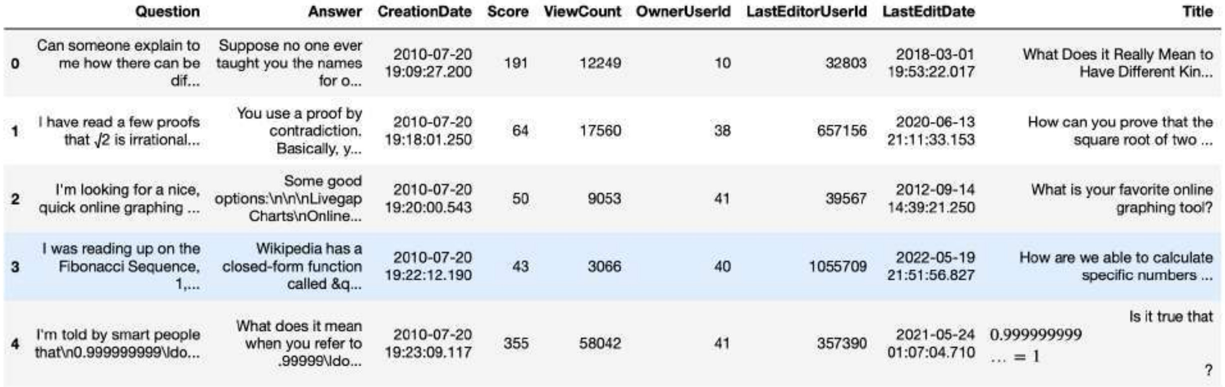
Original dataset with HTML tags in columns Question and Answer



	Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	
0	<p>Can someone explain to me how there can be ...	<p>Suppose no one ever taught you the names fo...	2010-07-20T19:09:27.200	191	12249.0	10.0	32803.0	2018-03-01T19:53:22.017	What I Really t Differen
1	<p>I have read a few proofs that $\sqrt{2}$ is...	<p>You use a proof by contradiction. Basically...	2010-07-20T19:18:01.250	64	17560.0	38.0	657156.0	2020-06-13T21:11:33.153	How c prove tl square
2	<p>I'm looking for a nice, quick online graphi...	<p>Some good options: </p>\n\n<ul>\n<li><a href...	2010-07-20T19:20:00.543	50	9053.0	41.0	39567.0	2012-09-14T14:39:21.250	What favorite gr
3	<p>I was reading up on the Fibonacci Sequence,...	<p>Wikipedia has a closed-form function called...	2010-07-20T19:22:12.190	43	3066.0	40.0	1055709.0	2022-05-19T21:51:56.827	How a ca s numl
4	<p>I'm told by <em>smart people</em> that\n<sp...	<p>What does it mean when you refer to $.99999...	2010-07-20T19:23:09.117	355	58042.0	41.0	357390.0	2021-05-24T01:07:04.710	Is it tr 0.9999 … = 1
5	<p>Given the semi-major axis and a flattening ...	<p>Possibly something like this. Correct me if...	2010-07-20T19:24:34.823	11	3565.0	17.0	343408.0	2017-03-02T13:36:40.420	How calcul semi axis
6	<p>In mathematics, there seem to be a lot of d...	<ul>\n<li><p>Natural numbers \nThe "countin...	2010-07-20T19:25:09.563	17	7542.0	54.0	465225.0	2017-08-02T15:54:22.223	What is numbe ra dec
7	<p>By matrix-defined, I mean</p>\n\n<p>$$\left...	<p>Assuming you know the definition of orthogo...	2010-07-20T19:25:58.483	18	2149.0	51.0	167548.0	2016-12-23T23:48:03.357	Why r defined Proc
8	<p>I'm looking for an online or software calcu...	<p>SpeQ is really nice. If you close it's left...	2010-07-20T19:27:09.700	40	3197.0	54.0	73.0	2010-07-28T01:04:25.440	C recomn decent or softw

Figure 23

Columns after removal of HTML tags

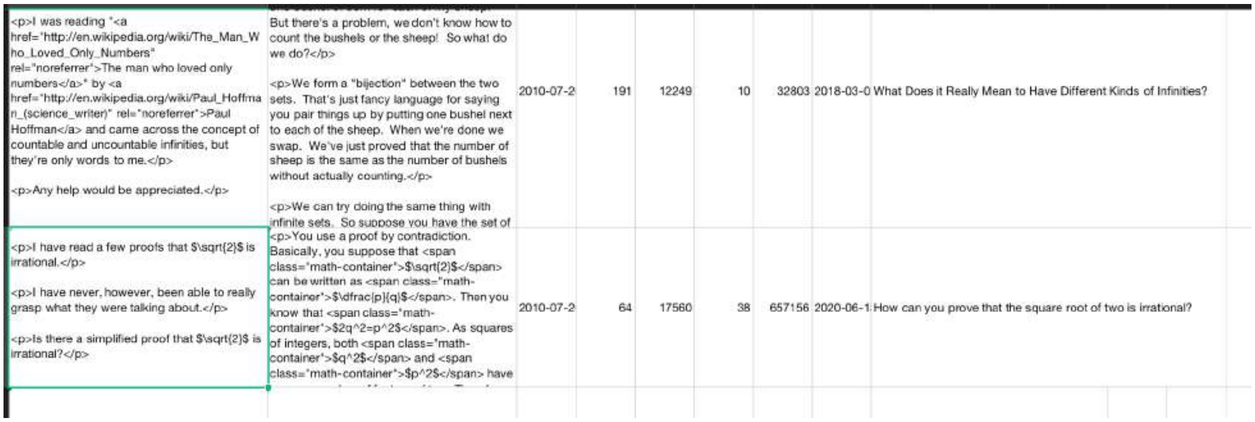


	Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
0	Can someone explain to me how there can be dif...	Suppose no one ever taught you the names for o...	2010-07-20 19:09:27.200	191	12249	10	32803	2018-03-01 19:53:22.017	What Does it Really Mean to Have Different Kin...
1	I have read a few proofs that √2 is irrational...	You use a proof by contradiction. Basically, y...	2010-07-20 19:18:01.250	64	17560	38	657156	2020-06-13 21:11:33.153	How can you prove that the square root of two ...
2	I'm looking for a nice, quick online graphing ...	Some good options:\n\n\nLivegap Charts\nOnline...	2010-07-20 19:20:00.543	50	9053	41	39567	2012-09-14 14:39:21.250	What is your favorite online graphing tool?
3	I was reading up on the Fibonacci Sequence, 1,...	Wikipedia has a closed-form function called &q...	2010-07-20 19:22:12.190	43	3066	40	1055709	2022-05-19 21:51:56.827	How are we able to calculate specific numbers ...
4	I'm told by smart people that\n0.999999999\ldo...	What does it mean when you refer to .99999\ldo...	2010-07-20 19:23:09.117	355	58042	41	357390	2021-05-24 01:07:04.710	Is it true that $0.999999999\ldots = 1$?

The provided dataset encompasses mathematical expressions expressed in LaTeX format. Figure 24 describes LaTeX expressions that feature diverse mathematical notations and symbols used for conveying mathematical concepts. One noteworthy component within the dataset pertains to the representation of square roots in English. In LaTeX, square roots are conventionally denoted using the "\sqrt{}" command, followed by the mathematical expression enclosed within the curly braces that is to be square rooted. The task at hand revolves around the conversion of these LaTeX representations of square roots into their corresponding Mathematical symbols, making the mathematical expressions more comprehensible to a broader audience unacquainted with LaTeX notation.

Figure 24

Latex expressions in the original dataset

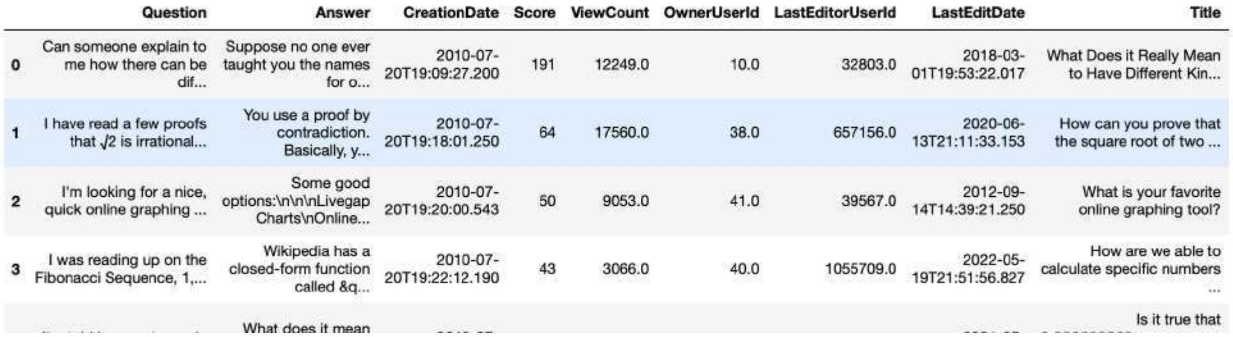


<p>I was reading "<a href="http://en.wikipedia.org/wiki/The_Man_Who_Loved_Only_Numbers" rel="noreferrer">The man who loved only numbers</a>" by <a href="http://en.wikipedia.org/wiki/Paul_Hoffman_(science_writer)" rel="noreferrer">Paul Hoffman</a> and came across the concept of countable and uncountable infinities, but they're only words to me.</p> <p>Any help would be appreciated.</p>	But there's a problem, we don't know how to count the bushels or the sheep! So what do we do?</p> <p>We form a "bijection" between the two sets. That's just fancy language for saying you pair things up by putting one bushel next to each of the sheep. When we're done we swap. We've just proved that the number of sheep is the same as the number of bushels without actually counting.</p> <p>We can try doing the same thing with infinite sets. So suppose you have the set of	2010-07-2	191	12249	10	32803	2018-03-0	What Does it Really Mean to Have Different Kinds of Infinities?
<p>I have read a few proofs that $\sqrt{2}$ is irrational.</p> <p>I have never, however, been able to really grasp what they were talking about.</p> <p>Is there a simplified proof that $\sqrt{2}$ is irrational?</p>	<p>You use a proof by contradiction. Basically, you suppose that <span class="math-container">$\sqrt{2}$</span> can be written as <span class="math-container">$\dfrac{p}{q}$</span>. Then you know that <span class="math-container">$2q^2=p^2$</span>. As squares of integers, both <span class="math-container">q^2</span> and <span class="math-container">p^2</span> have	2010-07-2	64	17560	38	657156	2020-06-1	How can you prove that the square root of two is irrational?

Within the dataset at hand, we've effectively translated the English term "Square root" into its corresponding mathematical reading symbols. This conversion process replaces the English language representation with the precise mathematical notation, aligning the dataset more closely with established mathematical standards. This cleaning ensures that the dataset now presents mathematical concepts and expressions in a standardized manner, enhancing its suitability for various mathematical and computational applications. The accompanying figure 25 visually demonstrates this conversion, depicting how "Square root" seamlessly integrates into the dataset as the mathematical symbol (√). This adaptation guarantees that users can readily engage with and grasp the mathematical content contained in the dataset, thereby facilitating a range of mathematical and scientific analyses.

Figure 25

Conversion of Latex expressions to readable mathematical symbols



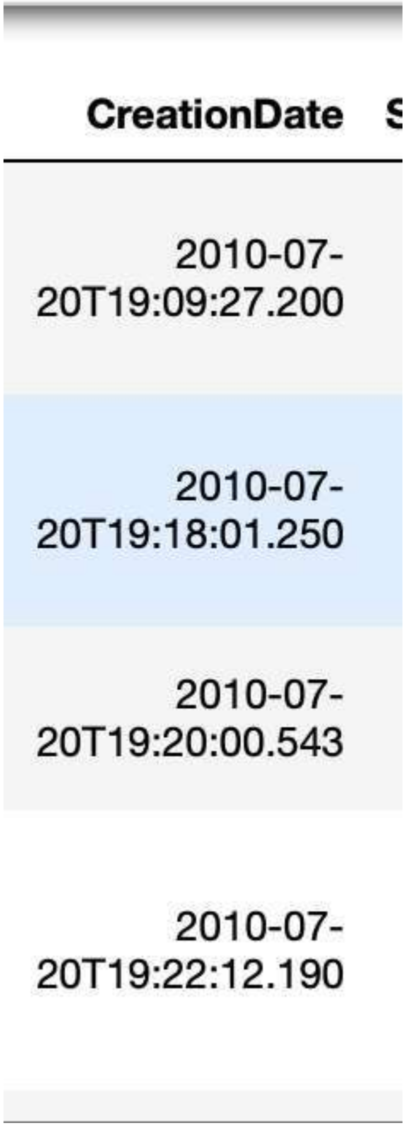
	Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
0	Can someone explain to me how there can be dif...	Suppose no one ever taught you the names for o...	2010-07-20T19:09:27.200	191	12249.0	10.0	32803.0	2018-03-01T19:53:22.017	What Does it Really Mean to Have Different Kin...
1	I have read a few proofs that √2 is irrational...	You use a proof by contradiction. Basically, y...	2010-07-20T19:18:01.250	64	17560.0	38.0	657156.0	2020-06-13T21:11:33.153	How can you prove that the square root of two ...
2	I'm looking for a nice, quick online graphing ...	Some good options:\n\n\nLivegap Charts\nOnline...	2010-07-20T19:20:00.543	50	9053.0	41.0	39567.0	2012-09-14T14:39:21.250	What is your favorite online graphing tool?
3	I was reading up on the Fibonacci Sequence, 1,...	Wikipedia has a closed-form function called &q...	2010-07-20T19:22:12.190	43	3066.0	40.0	1055709.0	2022-05-19T21:51:56.827	How are we able to calculate specific numbers ...
		What does it mean							Is it true that

Handling of Inconsistent Data

Handling inconsistent date formats in data is crucial for ensuring data quality and consistency. In many datasets, dates may appear in various formats which can lead to errors in analysis. To address this issue, Python's date parsing libraries are used like pandas to standardize date formats. In this project, firstly various date formats in the dataset were identified, followed by mapping that defines how each format should be converted into a uniform format, typically the creation date column "YYYY-MM-DDTHH:MM:SS." In below figure 26 it can be seen that the date column is in inconsistent format, as the datatype of the date column is object. After changing the datatype to “date” it can be seen the format has become consistent and changed to “YYYY-MM-DD HH:MM:SS” Handling inconsistent date formats ensures that your data is reliable and facilitates meaningful insights from your dataset.

Figure 26

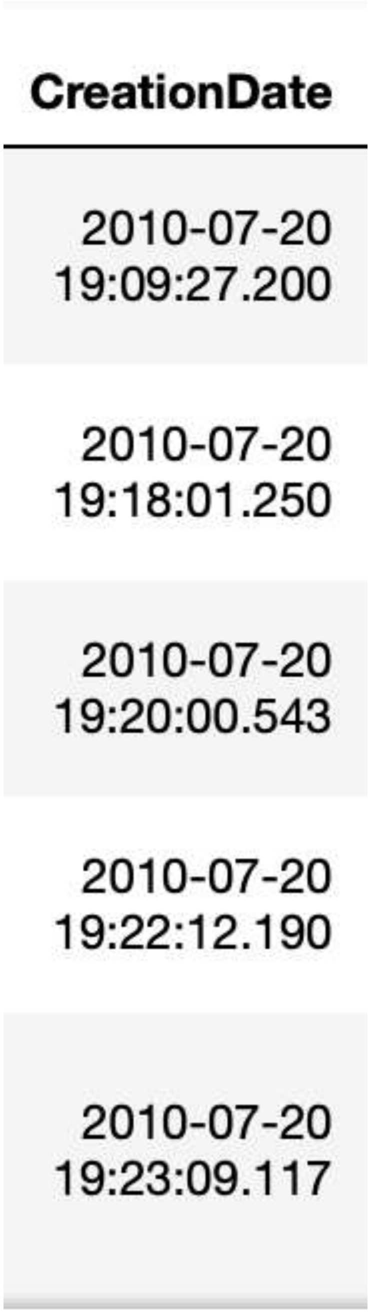
CreationDate column in inconsistent format



CreationDate
2010-07-20T19:09:27.200
2010-07-20T19:18:01.250
2010-07-20T19:20:00.543
2010-07-20T19:22:12.190

Figure 27

CreationDate column in consistent format



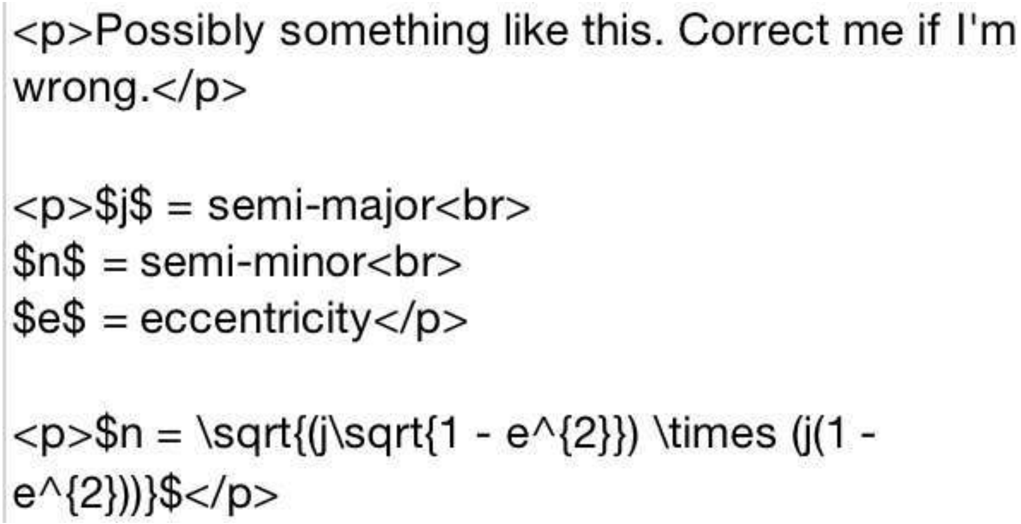
CreationDate
2010-07-20 19:09:27.200
2010-07-20 19:18:01.250
2010-07-20 19:20:00.543
2010-07-20 19:22:12.190
2010-07-20 19:23:09.117

Handling of Noisy Data

Figure 28 shows the presence of extraneous symbols, notably the '$' symbol, scattered throughout the dataset. These '$' symbols lack inherent mathematical or contextual significance, rendering them noise or undesirable elements within the dataset. Managing noisy data, such as the inclusion of '$' symbols, stands as a pivotal preprocessing step in the realm of data analysis and manipulation. The process entails the systematic identification and subsequent removal of these superfluous symbols from the dataset. This effort results in a dataset that is cleaner and more streamlined. Once this noisy data is effectively addressed, it paves the way for more precise mathematical computations, analyses, and various data-driven tasks. This, in turn, enhances the dataset's overall quality and its suitability for a range of analytical and computational purposes.

Figure 28

Presence of $ symbol in the original dataset

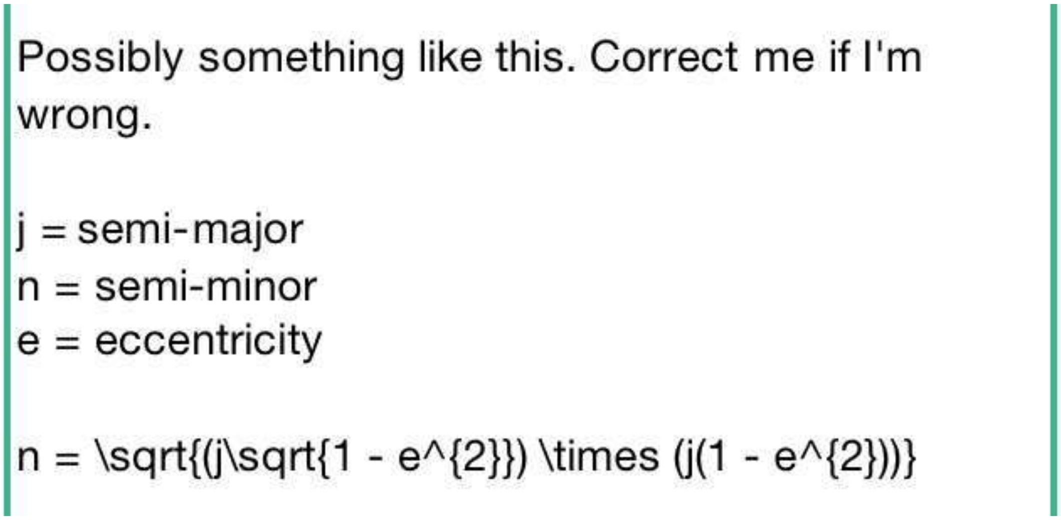


```
<p>Possibly something like this. Correct me if I'm
wrong.</p>

<p>$j$ = semi-major<br>
$n$ = semi-minor<br>
$e$ = eccentricity</p>

<p>$n = \sqrt{(j\sqrt{1 - e^{2}}) \times (j(1 -
e^{2}))}$</p>
```

After conducting a comprehensive data cleaning procedure, the dataset now stands devoid of superfluous symbols, with the '$' symbol being the most prominent among them. This meticulous cleaning effort has resulted in a dataset that is now free of these undesired symbols which is shown in the figure 29. The elimination of the '$' symbols plays a pivotal role in elevating the dataset's overall quality and reliability for subsequent analysis and data manipulation. With these unnecessary symbols successfully purged, the dataset has attained a higher degree of precision and accuracy. This cleaning process not only enhances the dataset's usability but also upholds the overall data integrity, rendering it well-suited for a diverse array of analytical and computational applications.

Figure 29

After removal of $ symbol



Possibly something like this. Correct me if I'm wrong.

j = semi-major
n = semi-minor
e = eccentricity

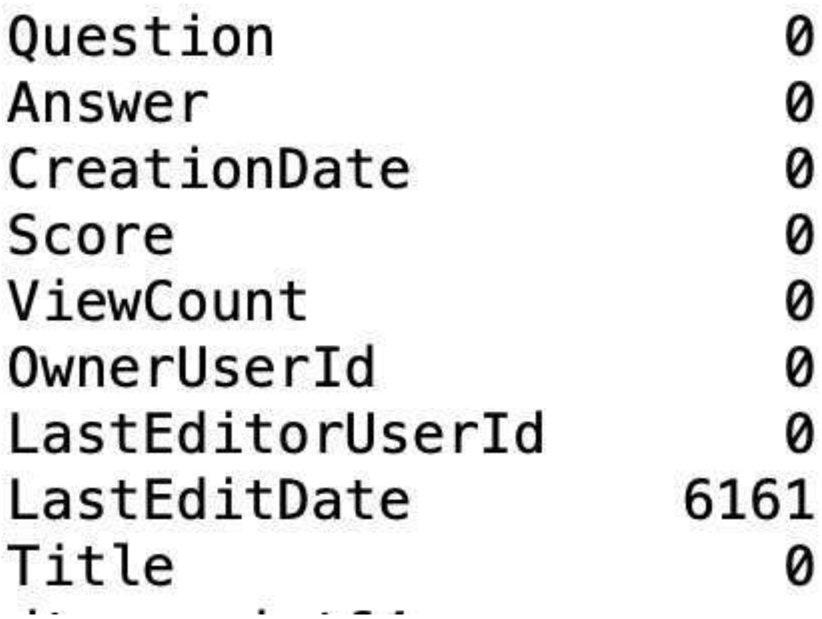
n = \sqrt{(j\sqrt{1 - e^{2}}) \times (j(1 - e^{2}))}

Handling of Incomplete & Missing Data

In the dataset, there exists a "Last Edit Date" column. This column's purpose lies in recording timestamps for any edits made by users to the provided answers. It's worth noting that not all users opt to modify the answers, resulting in instances where the "Last Edit Date" column contains missing values. Given the context of our analysis or project, we have chosen not to undertake specific actions to address or impute these gaps in the "Last Edit Date" column. Instead, we maintain the original dataset as-is, recognizing that this particular column primarily documents user interactions related to edits and may not consistently contain data for all entries. This approach ensures the dataset's integrity while accommodating variances in user behavior.

Figure 30

Handling of missing data



```
Question                 0
Answer                   0
CreationDate             0
Score                    0
ViewCount                0
OwnerUserId              0
LastEditorUserId         0
LastEditDate          6161
Title                    0
```

3.4. Data Transformation

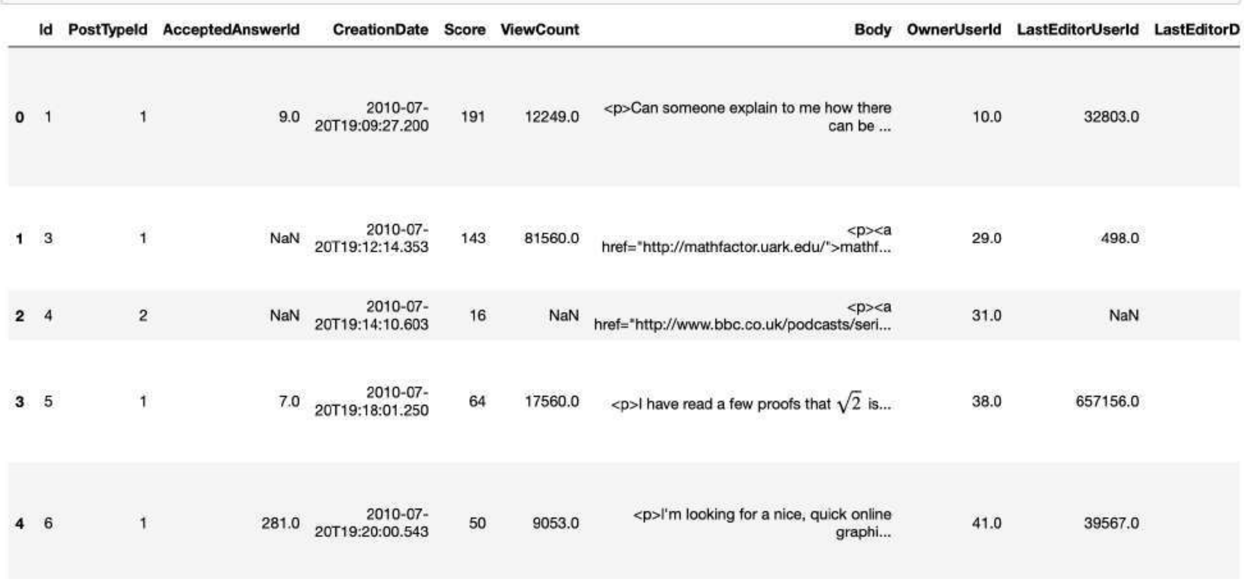
Feature extraction

Feature extraction is a critical step in the field of data analysis and machine learning. It involves selecting and transforming the most relevant information or attributes from a raw dataset to create a compact and meaningful representation. The importance of feature extraction lies in its ability to enhance the performance of machine learning algorithms by reducing dimensionality, removing irrelevant or redundant data, and highlighting the essential characteristics of the dataset. This process not only speeds up computation but also improves the model's accuracy and interpretability. Effective feature extraction enables to uncover hidden patterns and relationships within the data, ultimately leading to more informed decision-making, predictive modeling, and better understanding of complex datasets.

In the dataset of this project the questions and answers were not in separate columns, instead they were in a single column named “Body”. The data regarding questions and answers was unreadable, as it was very difficult to identify the questions and answers separately. In addition to this it was very difficult to identify the corresponding answers to a particular question. Figure 31 shows the column “Body” containing the data of both questions and answers in a single column.

Figure 31

Question and Answer content in single column “Body”

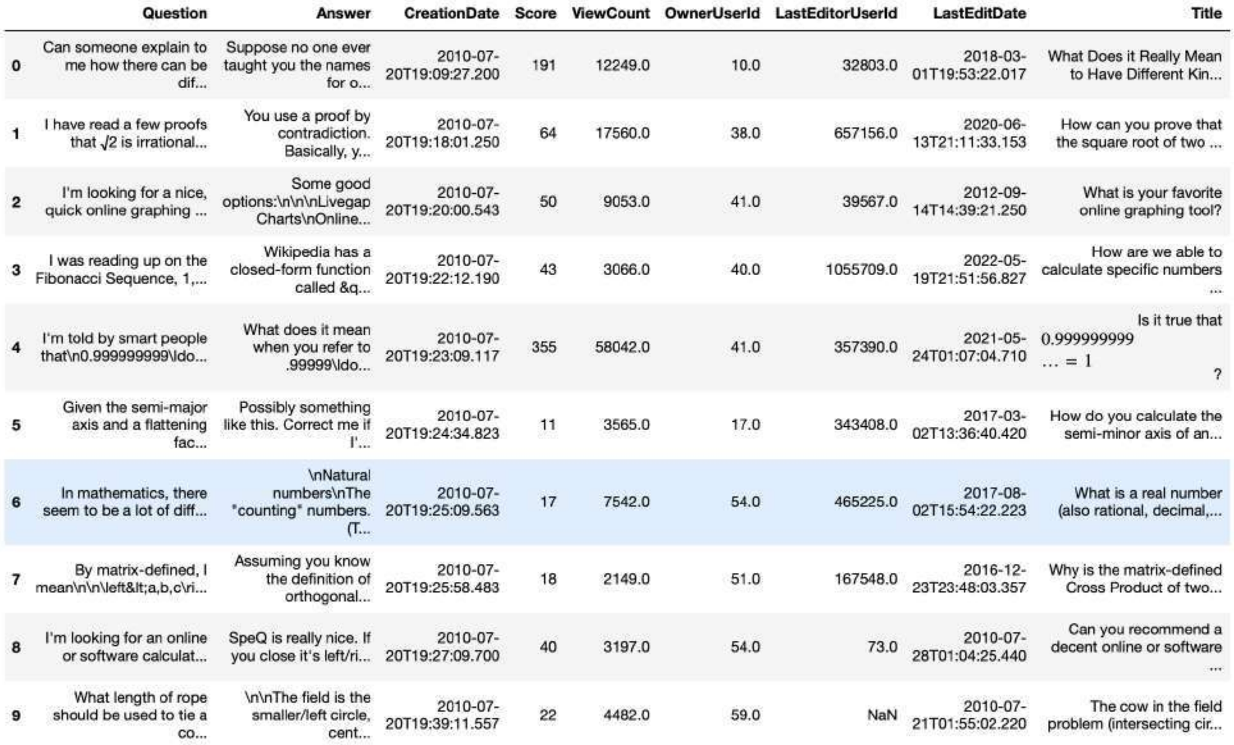


	Id	PostTypeId	AcceptedAnswerId	CreationDate	Score	ViewCount	Body	OwnerUserId	LastEditorUserId	LastEditorD
0	1	1	9.0	2010-07-20T19:09:27.200	191	12249.0	<p>Can someone explain to me how there can be ...	10.0	32803.0	
1	3	1	NaN	2010-07-20T19:12:14.353	143	81560.0	<p><a href="http://mathfactor.uark.edu/">mathf...	29.0	498.0	
2	4	2	NaN	2010-07-20T19:14:10.603	16	NaN	<p><a href="http://www.bbc.co.uk/podcasts/seri...	31.0	NaN	
3	5	1	7.0	2010-07-20T19:18:01.250	64	17560.0	<p>I have read a few proofs that $\sqrt{2}$ is...	38.0	657156.0	
4	6	1	281.0	2010-07-20T19:20:00.543	50	9053.0	<p>I'm looking for a nice, quick online graphi...	41.0	39567.0	

In order to make it readable and in understandable format, feature extraction was performed wherein two new columns were created named “Question” and “Answer”. This was performed by comparing and mapping the value of the “AcceptedAnswerId “column to the “Id” column. In figure 32 two separate columns named “Question” and ”Answer” are formed, making it more readable and appropriate for further analysis. Furthermore the NaN value in the AcceptedAnswerId column depicts that the corresponding answer was not accepted, whereas for the answer which was accepted the AcceptedAnswerId is displayed in integer format.

Figure 32

Separate columns of Question and Answer



	Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
0	Can someone explain to me how there can be dif...	Suppose no one ever taught you the names for o...	2010-07-20T19:09:27.200	191	12249.0	10.0	32803.0	2018-03-01T19:53:22.017	What Does it Really Mean to Have Different Kin...
1	I have read a few proofs that √2 is irrational...	You use a proof by contradiction. Basically, y...	2010-07-20T19:18:01.250	64	17560.0	38.0	657156.0	2020-06-13T21:11:33.153	How can you prove that the square root of two ...
2	I'm looking for a nice, quick online graphing ...	Some good options:\n\n\nLivegap Charts\nOnline...	2010-07-20T19:20:00.543	50	9053.0	41.0	39567.0	2012-09-14T14:39:21.250	What is your favorite online graphing tool?
3	I was reading up on the Fibonacci Sequence, 1,...	Wikipedia has a closed-form function called &q...	2010-07-20T19:22:12.190	43	3066.0	40.0	1055709.0	2022-05-19T21:51:56.827	How are we able to calculate specific numbers ...
4	I'm told by smart people that\n0.999999999\ldo...	What does it mean when you refer to .99999\ldo...	2010-07-20T19:23:09.117	355	58042.0	41.0	357390.0	2021-05-24T01:07:04.710	Is it true that $0.999999999\ldots = 1$?
5	Given the semi-major axis and a flattening fac...	Possibly something like this. Correct me if I'...	2010-07-20T19:24:34.823	11	3565.0	17.0	343408.0	2017-03-02T13:36:40.420	How do you calculate the semi-minor axis of an...
6	In mathematics, there seem to be a lot of diff...	\nNatural numbers\nThe "counting" numbers. (T...	2010-07-20T19:25:09.563	17	7542.0	54.0	465225.0	2017-08-02T15:54:22.223	What is a real number (also rational, decimal,...
7	By matrix-defined, I mean\n\n\left<a,b,c\ri...	Assuming you know the definition of orthogonal...	2010-07-20T19:25:58.483	18	2149.0	51.0	167548.0	2016-12-23T23:48:03.357	Why is the matrix-defined Cross Product of two...
8	I'm looking for an online or software calculat...	SpeQ is really nice. If you close it's left/ri...	2010-07-20T19:27:09.700	40	3197.0	54.0	73.0	2010-07-28T01:04:25.440	Can you recommend a decent online or software ...
9	What length of rope should be used to tie a co...	\n\nThe field is the smaller/left circle, cent...	2010-07-20T19:39:11.557	22	4482.0	59.0	NaN	2010-07-21T01:55:02.220	The cow in the field problem (intersecting cir...

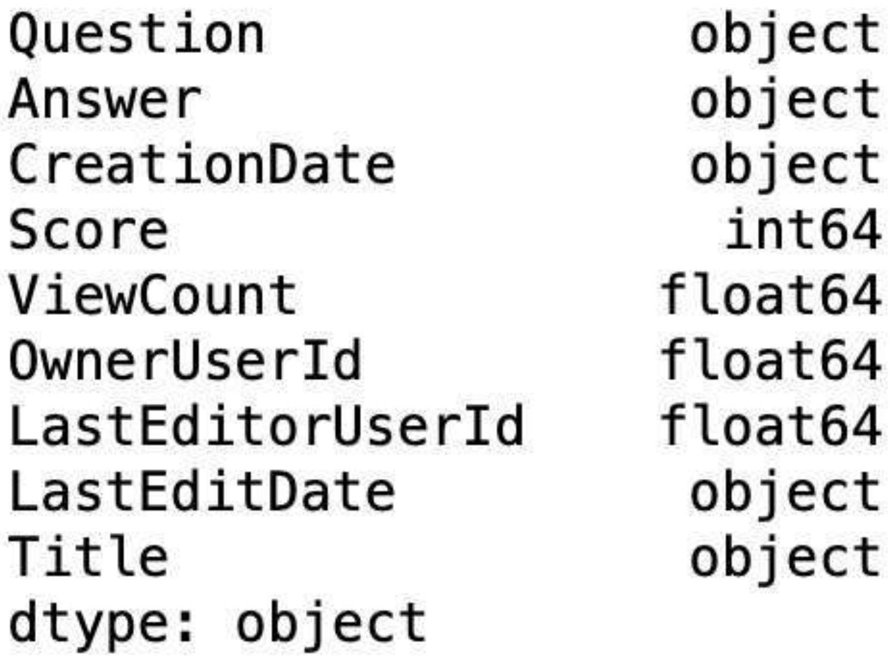
Converting data type to suitable format

Converting data types to a suitable format is a fundamental aspect of data preprocessing that plays a critical role in data analysis and modeling. It involves transforming data from one type to another to ensure that it aligns with the requirements of your analysis or machine learning algorithms. The importance of this process lies in its ability to enhance data quality and accuracy. For instance, converting numeric values that are initially stored as strings into numerical data types allows you to perform mathematical operations and statistical analyses. Similarly, changing date and time data from strings to date-time objects enables chronological analyses and time-based modeling. By converting data to the most appropriate format, not only makes it more compatible with the analytical tools but also reduce the risk of errors and discrepancies in the results, ultimately leading to more robust and reliable insights.

In the dataset of this project the datatype of the columns were inconsistent making it difficult for further analysis. The datatype of the “Creationdate” column was an object which is not appropriate, which needs to be in the datetime format which is the ideal datatype of a date column. Similar case can be seen in the LastEditDate column which is corrected by changing the datatype to datetime format. The datatype of “ViewCount”, “OwnerUserId” and “LastEditorUserId” is also changed from float64 to int64.

Figure 33

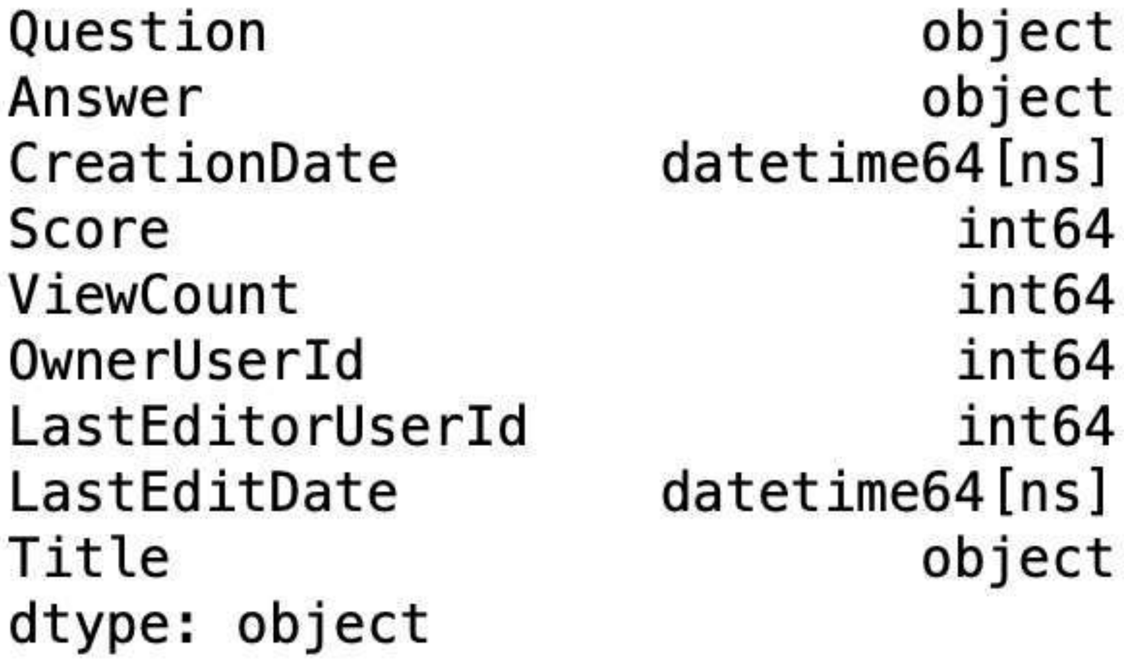
Datatypes of columns in original dataset



```
Question             object
Answer               object
CreationDate         object
Score                 int64
ViewCount           float64
OwnerUserId         float64
LastEditorUserId    float64
LastEditDate         object
Title                object
dtype: object
```

Figure 34

Datatypes of columns after changing the datatype



```
Question                        object
Answer                          object
CreationDate            datetime64[ns]
Score                            int64
ViewCount                        int64
OwnerUserId                      int64
LastEditorUserId                 int64
LastEditDate            datetime64[ns]
Title                           object
dtype: object
```

3.5. Data Preparation

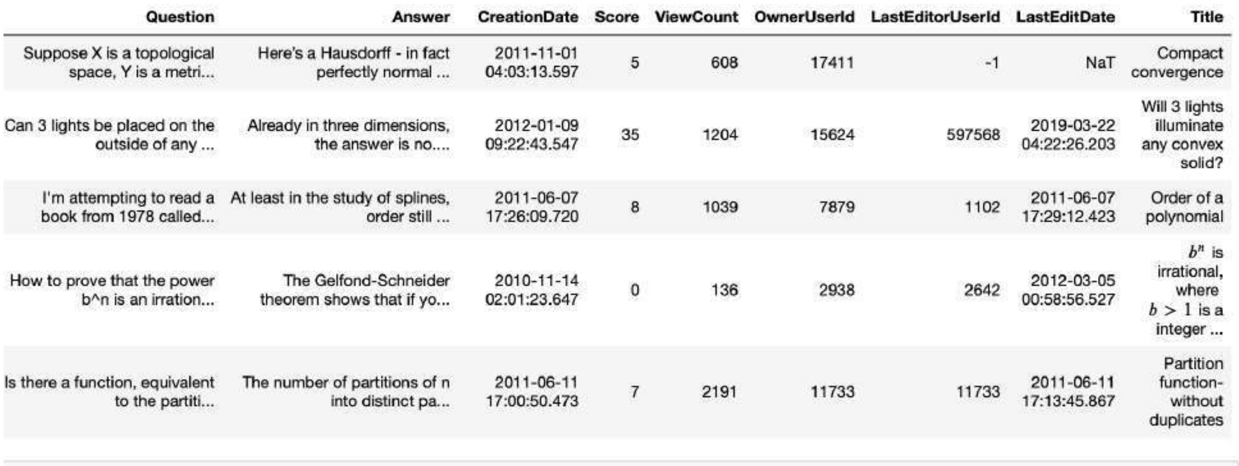
Navigating the vast landscape of data from platforms like Math Stack Overflow and other trustworthy sources requires a strategic approach to refinement and organization. The subsequent goal is to develop a machine learning model adept at ranking answers, and it hinges on the quality of this curated data.

Train Dataset

A significant portion of the accumulated data acts as the training set. This dataset equips the machine learning model to discern relationships between metrics such as user profiles, upvotes, downvotes, and solution completeness. With this set, the ranking algorithm hones its sensitivity to various facets of valuable answers. Illustrative samples of this dataset can be seen in Figure 35.

Figure 35

Sample training dataset



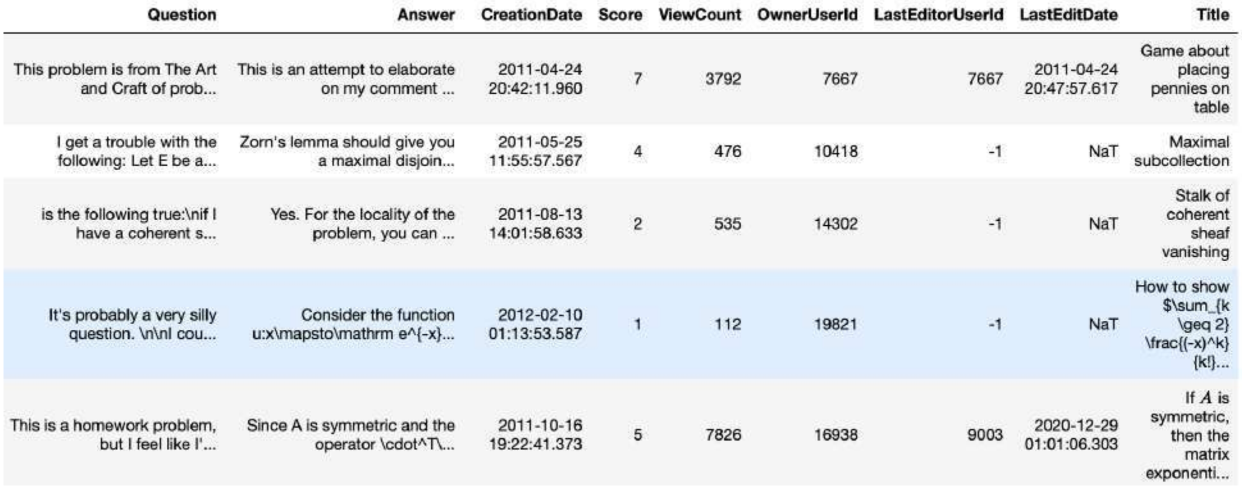
Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
Suppose X is a topological space, Y is a metri...	Here's a Hausdorff - in fact perfectly normal ...	2011-11-01 04:03:13.597	5	608	17411	-1	NaT	Compact convergence
Can 3 lights be placed on the outside of any ...	Already in three dimensions, the answer is no....	2012-01-09 09:22:43.547	35	1204	15624	597568	2019-03-22 04:22:26.203	Will 3 lights illuminate any convex solid?
I'm attempting to read a book from 1978 called...	At least in the study of splines, order still ...	2011-06-07 17:26:09.720	8	1039	7879	1102	2011-06-07 17:29:12.423	Order of a polynomial
How to prove that the power b^n is an irration...	The Gelfond-Schneider theorem shows that if yo...	2010-11-14 02:01:23.647	0	136	2938	2642	2012-03-05 00:58:56.527	b^n is irrational, where $b > 1$ is a integer ...
Is there a function, equivalent to the partiti...	The number of partitions of n into distinct pa...	2011-06-11 17:00:50.473	7	2191	11733	11733	2011-06-11 17:13:45.867	Partition function-without duplicates

Validation Dataset

To gauge the adaptability and robustness of the model, a distinct validation set is carved out. This data subset refines the ranking model's parameters, ensuring its relevance across diverse data scenarios. The structure and metrics of this validation set come to the fore in Figure 36.

Figure 36

Sample Validation dataset



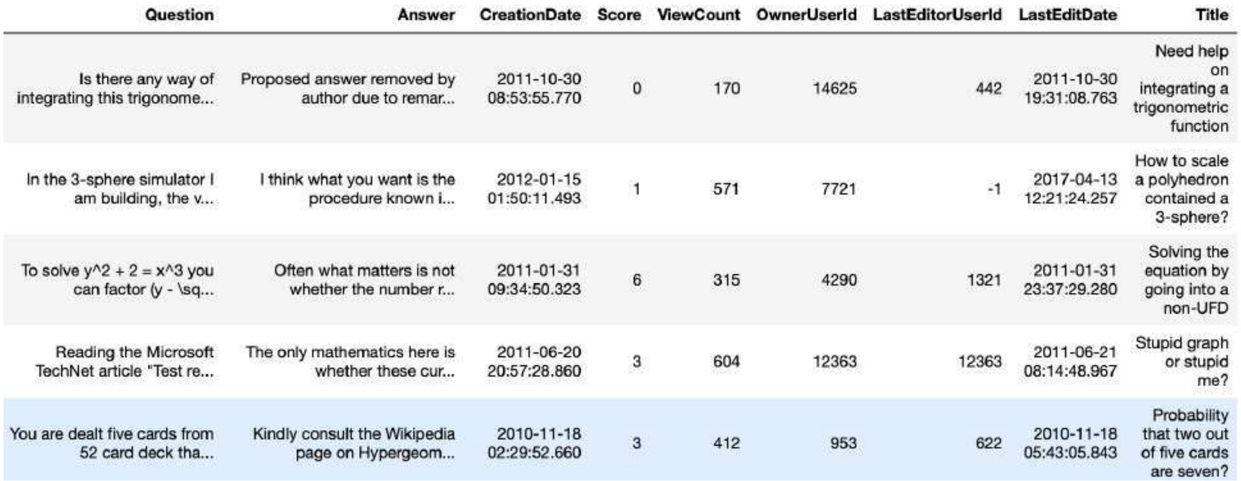
Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
This problem is from The Art and Craft of prob...	This is an attempt to elaborate on my comment ...	2011-04-24 20:42:11.960	7	3792	7667	7667	2011-04-24 20:47:57.617	Game about placing pennies on table
I get a trouble with the following: Let E be a...	Zorn's lemma should give you a maximal disjoin...	2011-05-25 11:55:57.567	4	476	10418	-1	NaT	Maximal subcollection
is the following true:\nif I have a coherent s...	Yes. For the locality of the problem, you can ...	2011-08-13 14:01:58.633	2	535	14302	-1	NaT	Stalk of coherent sheaf vanishing
It's probably a very silly question. \n\nI cou...	Consider the function u:x\mapsto\mathrm e^{-x}...	2012-02-10 01:13:53.587	1	112	19821	-1	NaT	How to show $\sum_{k \geq 2} \frac{(-x)^k}{k!}...
This is a homework problem, but I feel like I'...	Since A is symmetric and the operator \cdot^T\...	2011-10-16 19:22:41.373	5	7826	16938	9003	2020-12-29 01:01:06.303	If A is symmetric, then the matrix exponenti...

Test Dataset

Post refinement through training and validation, the model encounters the test dataset. This dataset, kept separate from the development phase, offers a measure of the model's efficacy in real-world scenarios. Its performance on this dataset provides insights into its ranking prowess in practical settings. Figure 37 offers a glimpse into the test dataset and its multifaceted metrics. Through rigorous data orchestration, the accumulated data undergoes strategic divisions. An initial allocation earmarks 85% for training and 15% for testing. Further strategizing splits the training subset into 70% for primary training and 15% for validation. This culminates in a balanced distribution: 70% for training, 15% for validation, and 15% for testing.

Figure 37

Sample testing dataset



Question	Answer	CreationDate	Score	ViewCount	OwnerUserId	LastEditorUserId	LastEditDate	Title
Is there any way of integrating this trigonome...	Proposed answer removed by author due to remar...	2011-10-30 08:53:55.770	0	170	14625	442	2011-10-30 19:31:08.763	Need help on integrating a trigonometric function
In the 3-sphere simulator I am building, the v...	I think what you want is the procedure known i...	2012-01-15 01:50:11.493	1	571	7721	-1	2017-04-13 12:21:24.257	How to scale a polyhedron contained a 3-sphere?
To solve y^2 + 2 = x^3 you can factor (y - \sq...	Often what matters is not whether the number r...	2011-01-31 09:34:50.323	6	315	4290	1321	2011-01-31 23:37:29.280	Solving the equation by going into a non-UFD
Reading the Microsoft TechNet article "Test re...	The only mathematics here is whether these cur...	2011-06-20 20:57:28.860	3	604	12363	12363	2011-06-21 08:14:48.967	Stupid graph or stupid me?
You are dealt five cards from 52 card deck tha...	Kindly consult the Wikipedia page on Hypergeom...	2010-11-18 02:29:52.660	3	412	953	622	2010-11-18 05:43:05.843	Probability that two out of five cards are seven?

3.6 Data Statistics

Data statistics serves as a fundamental component in the realm of data analysis and research. Key elements of data statistics include calculating measures like central tendencies (mean, median, mode) and dispersions (variance, standard deviation, range), as well as exploring relationships between variables through techniques like correlation and regression. Furthermore, data visualization techniques, such as heatmap and pairplot are employed to visually represent data patterns. In essence, data statistics plays a pivotal role in summarizing data and deriving meaningful conclusions, making it a cornerstone in making evidence-based decisions across diverse fields, including science, business, and the social sciences.

Table 9

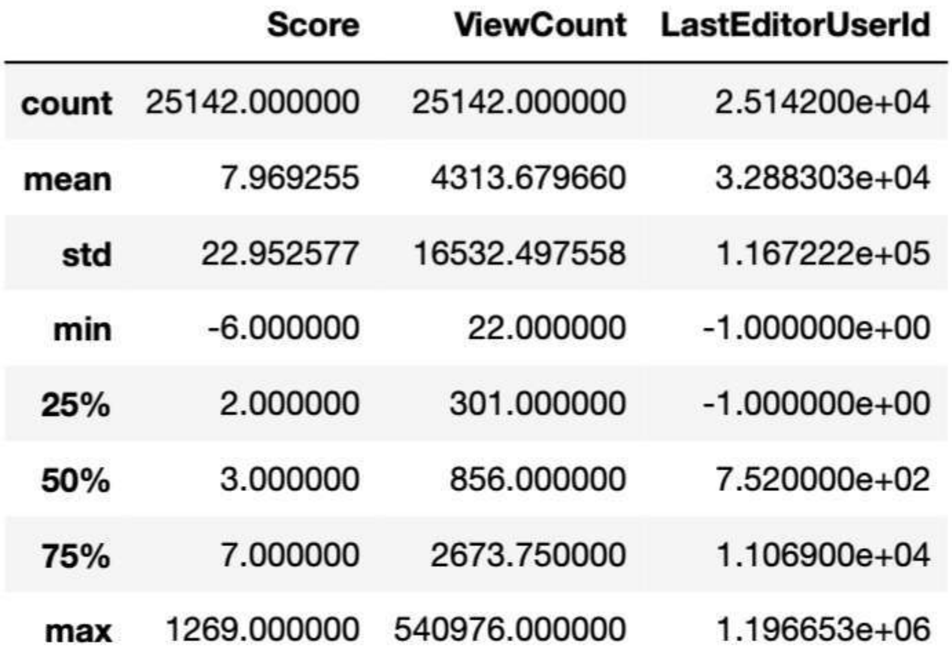
Dataset size after each stage

Stage name	Process	Rows X Columns
Raw Data	Extracting data from Math Stack Exchange	3708430 X 17
Pre-Processing	Pre-processing the dataset	741686 X 9
Transformation	Transforming rows and columns and summing up null values.	741686 X 9

Figure 38 finds a representation of essential dataset statistics. Notably, the "Score" column showcases an average score with a mean value of 6. Equally significant, both the "Score" and "View Count" columns reveal a shared count of 25142. These statistics provide valuable insights into the central tendencies of the "Score" column and the total number of entries present in both "Score" and "Viewcount" columns. These summary statistics serve as pivotal indicators of the dataset's underlying characteristics and the distribution of values, offering essential information for subsequent data analysis and interpretation.

Figure 38

Data description

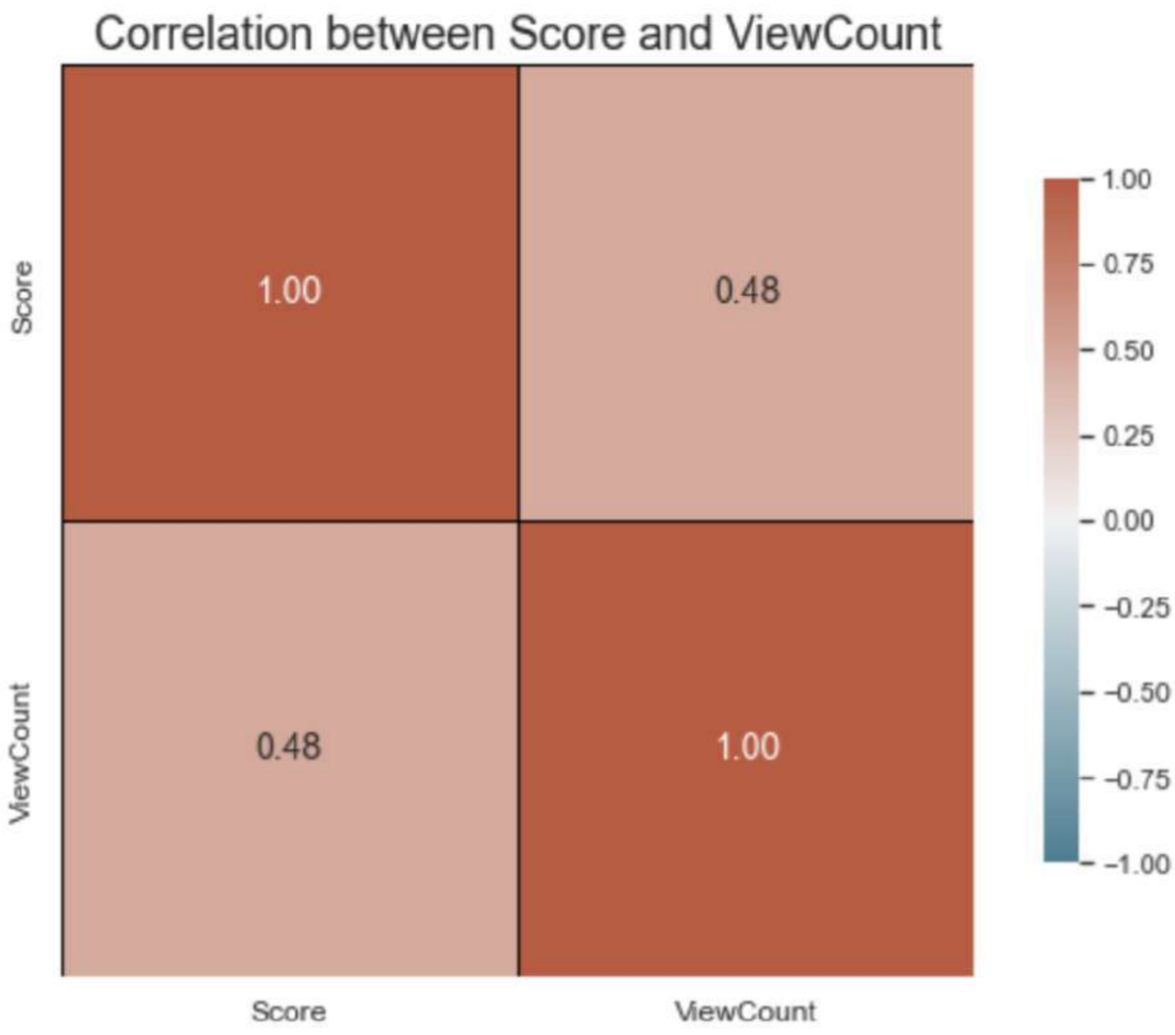


	Score	ViewCount	LastEditorUserId
count	25142.000000	25142.000000	2.514200e+04
mean	7.969255	4313.679660	3.288303e+04
std	22.952577	16532.497558	1.167222e+05
min	-6.000000	22.000000	-1.000000e+00
25%	2.000000	301.000000	-1.000000e+00
50%	3.000000	856.000000	7.520000e+02
75%	7.000000	2673.750000	1.106900e+04
max	1269.000000	540976.000000	1.196653e+06

Figure 39 shows a heatmap visualization that effectively conveys the correlation existing between the "Score" and "Viewcount" columns within the dataset. Heatmaps serve as valuable tools for visually depicting the strength and direction of relationships between variables. The darkness of the shades signifies the strength of the correlation, with darker tones indicating a stronger positive correlation and lighter tones suggesting weaker or negative correlations. By examining this heatmap, patterns within the data emerge, offering insights into whether higher scores tend to correspond with increased view counts or if any unexpected relationships exist.

Figure 39

Heatmap of Score and ViewCount


Correlation between Score and ViewCount
Score
ViewCount
1.00
0.48
0.48
1.00
Score
ViewCount
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Modeling

Stack Overflow, a key platform in the programming and data science community, is poised for a significant enhancement in how users engage with its content. The project titled "Ranked Stack Overflow: Mathematics and Statistical Analysis" focuses on employing sophisticated machine learning models to refine answer ranking and to automate tag generation for new questions. This initiative is designed to streamline the problem-solving process, ensuring that users can easily find the most relevant and high-quality answers.

The project utilizes a comprehensive dataset from the Stack Exchange network, encompassing features such as questions, answers, tags, post IDs, and parent IDs. Prior to any analytical work, this dataset underwent a rigorous cleaning process to ensure it was primed for machine learning applications.

Five distinct machine learning models form the backbone of this analysis, each selected for its effectiveness in processing and interpreting the complex text data encountered on Stack Overflow. These models include TF-IDF with Logistic Regression, Naive Bayes Classification, TF-IDF with SVM (Support Vector Machine), TF-IDF with Random Forest, and an Ensemble Method using a Bagging Classifier and Decision Tree Classifier. Each of these models plays a critical role in enhancing the tagging accuracy and ranking of answers within the platform.

The dataset is strategically divided into training (80%) and testing (20%) segments, facilitating a thorough assessment of each model's capabilities. The project highlights the transformative potential of machine learning in optimizing the use of extensive knowledge bases like Stack Overflow. It aims to deliver a more intuitive and efficient user experience, thereby enriching the community of developers and data scientists.

4.1 Model Proposals

Logistic Regression

Logistic Regression shines in binary classification tasks, such as assigning a specific tag to a question on Stack Overflow. This process converts text into numerical features that signify the importance of words within a document compared to a larger corpus. In the context of Stack Overflow questions, Logistic Regression calculates the likelihood of each question belonging to a particular class, or tag, using a logistic function. This function, characterized by its S-shaped curve, maps real-valued numbers between 0 and 1, effectively representing probabilities. By setting a standard threshold, typically 0.5, the model categorizes each question: if the calculated probability surpasses this threshold, the question gets tagged; if not, it remains untagged.

Input Features: It starts with input variables, denoted as x_1, x_2, ..., x_m , which are the independent factors sourced from the dataset. For instance, in the analysis of Stack Overflow data, these might be the figures derived from the TF-IDF processing of textual content.

Weights and Bias: Corresponding to each input variable is a weight w_1, w_2, ..., w_m, signifying the influence of each feature on the predictive outcome. Additionally, there is a bias term w_0, analogous to an intercept in a linear equation, allowing for a base level of prediction in the model.

Net Input Function: Symbolized by Sigma, this function sums up all the weighted inputs together with the bias. This cumulative value is the aggregated input that serves as the foundation for the subsequent prediction.

Sigmoid Activation Function: The sigmoid function, shown as an S-curve in the diagram, is pivotal for translating the net input into a probability value ranging from 0 to 1. This ensures that the Logistic Regression model yields a bounded output that can be interpreted as the likelihood of belonging to a specific class.

Threshold Function: A decision threshold is then applied to the probability output from the sigmoid function. Typically set at 0.5, this threshold determines the class assignment. Values above this cutoff point indicate one class, while values below suggest the alternative class.

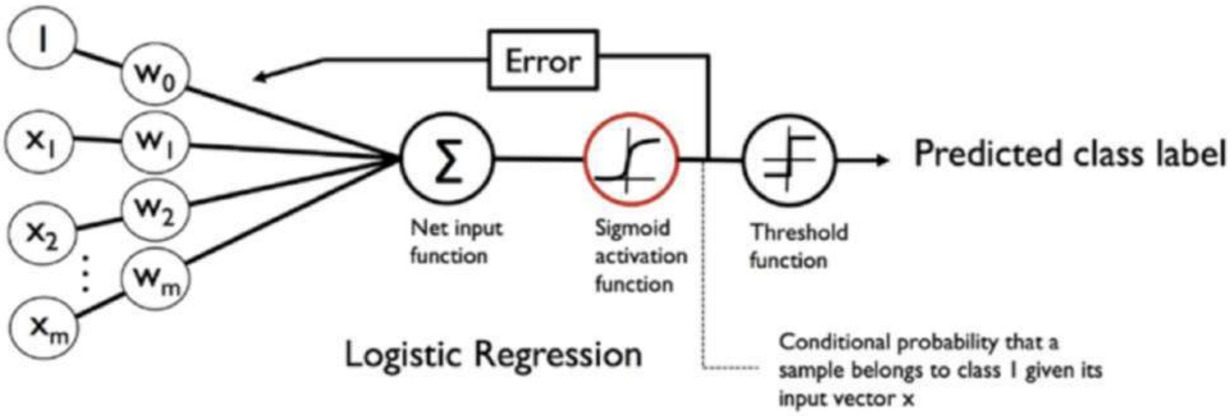
Predicted Class Label: The culmination of this process is the assignment of a predicted class label, signifying the model's verdict on classifying the input data. In a practical application, this would equate to deciding on the assignment of a specific tag to a Stack Overflow question.

Error: Referenced in the diagram is the 'Error,' indicative of the discrepancy between the model's predictions and the actual labels. During training, the model tweaks its weights and bias to minimize this error, refining its predictions over the training dataset.

In essence, the Logistic Regression architecture illustrates the method by which input data, such as questions from Stack Overflow, are translated into a probability of association with a particular tag, culminating in a clear binary decision following the threshold function. This mechanism is integral to a supervised learning approach, where the model iteratively improves its parameters for enhanced prediction accuracy.

Figure 40

Architecture of Logistic Regression



The Logistic Regression algorithm is structured to predict the likelihood of a binary outcome based on input data. Here’s an overview of the algorithm's flow:

Step 1: Establishing Probability

The initial aim is to determine the probability that an input belongs to a particular class, labeled as the default. Represented as P{X}, this is the probability for a set of features X.

Step 2: Linear Combination Calculation

The algorithm begins by calculating the linear combination of the input features, denoted by x_1, x_2, . . . , x_n , each weighted by corresponding coefficients. The formula is:

$$X = \beta_0 + \beta_1 x_1 + \beta_n x_n$$

Step 3: Sigmoid Function Transformation

Next, the computed linear combination is input into the sigmoid or logistic function, which maps the result to a probability value between 0 and 1, as shown below:

$$P\left(X\right) = \frac{e^{\beta_0+\beta_1 x_i+\beta_n x_n}}{1+e^{\beta_0+\beta_1 x_i+\beta_n x_n}}$$

The above equation can also be written as:

$$P\left(x\right) = \frac{1}{1+e^{-(X)}}$$

Step 4: Log-Odds Conversion

The output from the sigmoid function can be expressed in terms of odds, which is the probability of the event happening against it not happening. The log of these odds, or log-odds, is then calculated and is directly proportional to the input features:

$$log\left(\frac{p(X)}{1-P(X)}\right) = \beta_0 + \beta_n x_n$$

Step 5: Optimizing Model Parameters

The training process involves optimizing the model parameters, the weights, and the intercept to ensure the predicted probabilities match the observed outcomes as closely as possible, often using a method known as maximum likelihood estimation.

Step 6: Generating Predictions

With the model trained, it can generate predictions for new data. By inputting a new set of features X into the model, it yields log-odds, which are then passed through the logistic function to produce a probability. If this probability exceeds a certain threshold, typically 0.5, the input is predicted to belong to the default class.

Naive Bayes Classification

The Naive Bayes classifier operates on the principles of Bayes' Theorem, with the key assumption that the presence (or absence) of a particular feature is unrelated to the presence (or absence) of any other feature, given the classification outcome. This simplification facilitates the model's ability to swiftly process data and make accurate predictions. Applied to categorizing Stack Overflow questions, the Naive Bayes model computes the conditional probability for each tag based on the text. It assesses the likelihood of each individual word within the context of a tag, then aggregates these probabilities to determine the most likely tag for a given question. Despite its straightforward approach, Naive Bayes is known for its effectiveness in text classification tasks, particularly when dealing with substantial datasets where word frequency can be a potent predictor of content categorization. Presented next is the architecture of the model, illustrating its method of probabilistic classification (Arya et al., 2023).

The displayed diagram outlines a structured approach to text classification using the Naive Bayes method. The sequence initiates with the 'Excel Reader' node, which serves the purpose of ingesting the dataset, presumably a collection of text entries from an Excel file, such as queries or posts from Stack Overflow.

Progressing to the 'Partitioning' node, the dataset undergoes a division, segregating the data into distinct sets for the dual purposes of model training and subsequent evaluation. This split is foundational to validating the predictive accuracy of the model on data it has not previously encountered.

At the 'Naive Bayes Learner' node, the focus shifts to model training. This stage involves the model assimilating the patterns within the training set, calculating the likelihood of feature occurrences in relation to specific class designations.

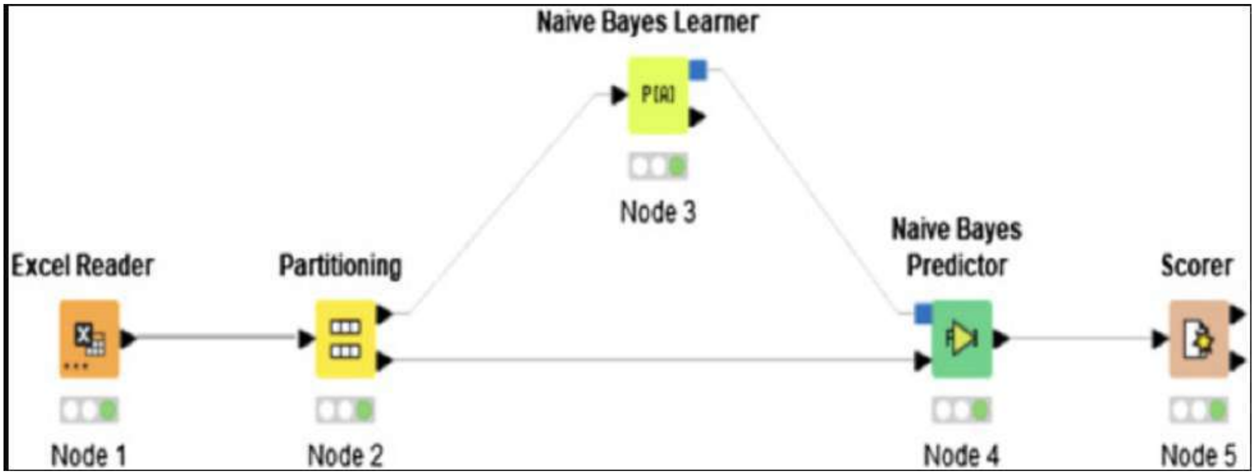
Transitioning to the 'Naive Bayes Predictor' node, the predictive capacity of the model is put into action. The trained model applies its learned probability estimates to new instances, predicting their classification based on the attributes they exhibit.

Concluding the workflow is the 'Scorer' node, where the effectiveness of the model is quantitatively assessed. By juxtaposing the predicted labels with the actual ones from the test set, the model's proficiency in accurately classifying the data is ascertained, typically through established metrics such as accuracy and F1-score.

This progression from data ingestion to performance assessment encapsulates the Naive Bayes classification workflow, representing a full cycle of machine learning from raw data input to the evaluation of a predictive model.

Figure 41

Architecture of Naive Bayes Classifier



The Naive Bayes classifier is anchored in Bayes' Theorem and presumes that all features in a dataset contribute independently and equally to the outcome. This classifier estimates the probability that an entity belongs to a particular class based on its attributes, treating each attribute separately without regard for any possible correlations. It is an efficient method, especially with high-dimensional data, due to its simplistic computational model and robustness to noise.

In practice, the Naive Bayes algorithm assesses the likelihood of a class y_i given specific features X_1, X_2, X_3 . This likelihood, represented as P(y_i | X_1, X_2, X_3), is derived by multiplying the probabilities of each feature occurring in the context of the class \(P(X_j | y_i) \) with the prior probability of the class P(y_i). When processing actual data, the denominator of Bayes' Theorem, which is the probability of the features P(X_1)P(X_2)P(X_3), remains constant, thus the model focuses on the numerator for efficiency.

The algorithm's calculation of probabilities is contingent on the feature types. For discrete data, it assumes a multinomial distribution, while for continuous data, it utilizes a Gaussian distribution. To determine these probabilities, the Naive Bayes model computes statistical parameters such as the mean and variance for each class. These parameters are then used to calculate the likelihood of the features within each class, P(X_j | y_i).

The decision-making aspect of the algorithm applies the maximum a posteriori (MAP) rule. It selects the class y_i that maximizes the product of the individual feature probabilities and the class probability:

$$y = \text{argmax}_{y_i} P(X_1|y_i)P(X_2|y_i)P(X_3|y_i)P(y_i)$$

This step is essentially picking the class that is most probable given the observed features, leading to a prediction about the entity's class based on the provided attributes.

Random Forest

A Random Forest model operates by generating a collection of decision trees during the training phase, each providing its own individual prediction. When it comes to sorting and tagging responses on Stack Overflow, this model would scrutinize textual features, such as the frequency of terms or the occurrence of specific phrases, to guide the decision-making process within each tree. Each tree in the ensemble is built on a distinct subset of features, which is randomly selected, enhancing the model's ability to generalize and minimizing the risk of overfitting to the training data. The aggregation of decisions from multiple trees leads to a final verdict on the ranking of an answer, taking into account various indicators of quality and relevance. Given its ability to manage datasets with a high number of features and to unravel intricate, non-linear associations, Random Forest is well-suited for discerning the most pertinent answers on the platform. Below is the architecture of the model, depicting its collective structure and the way it arrives at conclusions (Abdulazeez et al., 2021).

The image illustrates how a Random Forest algorithm operates when it comes to making predictions. Imagine feeding a piece of data into a system that consists of a multitude of decision-making paths — in this case, a collection of 600 decision trees. Each tree in this ensemble method takes a crack at interpreting the data independently and comes up with its own prediction.

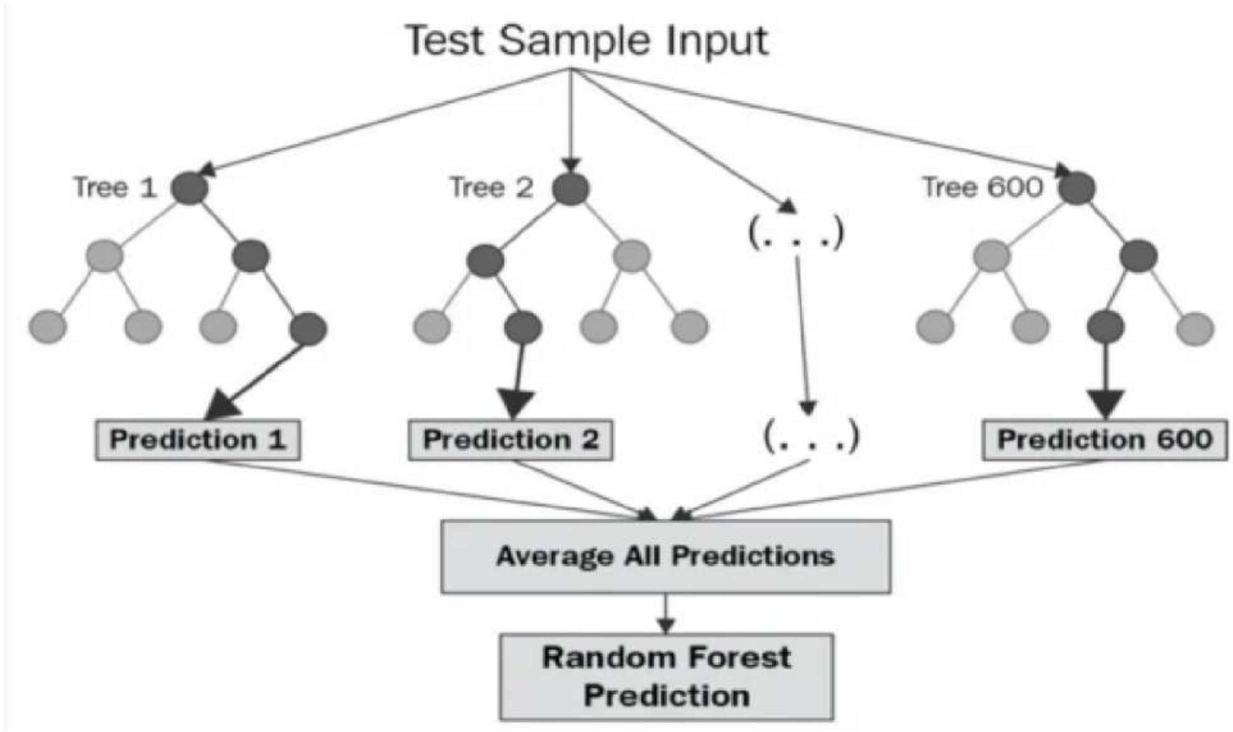
Think of each decision tree as an individual in a crowd, each with a unique perspective. Some may get a better view than others, and as a result, they all end up with different conclusions. This diversity is intentional and stems from how the Random Forest algorithm trains each tree on varied samples of the overall data.

The trees' various predictions are then combined, using an averaging method here. This suggests we're looking at a numerical prediction, as you'd find in regression problems. Were this a classification task, you might instead see a tally of votes to determine the most common prediction.

The beauty of this method lies in its collective wisdom. By averaging the outcomes from all 600 trees, the algorithm arrives at a single, more accurate prediction. This is typically more trustworthy than any individual tree's guess since it balances out the individual errors and eccentricities, providing a well-rounded final verdict.

Figure 42

Architecture of Random Forest



The algorithm in question is a blueprint for creating a Random-Forest Classifier, which is essentially a collection of decision trees that work together to classify data more effectively than any single tree could on its own. It kicks off with a for-loop that runs from 1 up to a predetermined number N, which represents the quantity of individual trees that will form the forest.

Within this loop, each pass starts by generating a bootstrap dataset, noted as B_i, by selecting data points from the original dataset D with the possibility of repetition. This new dataset D_i becomes the training grounds for one decision tree, and the data points not picked for D_i, labeled as out-of-bag (OOB), are set aside to test the model's accuracy later on.

Following the creation of the bootstrap dataset, the root node of the new tree is established, and the recursive function `Build-Tree` is put into action. This function's job is to organize the dataset at the node, using entropy as a yardstick for disorder within the data. A lower entropy reflects a more orderly set, which is the ideal scenario.

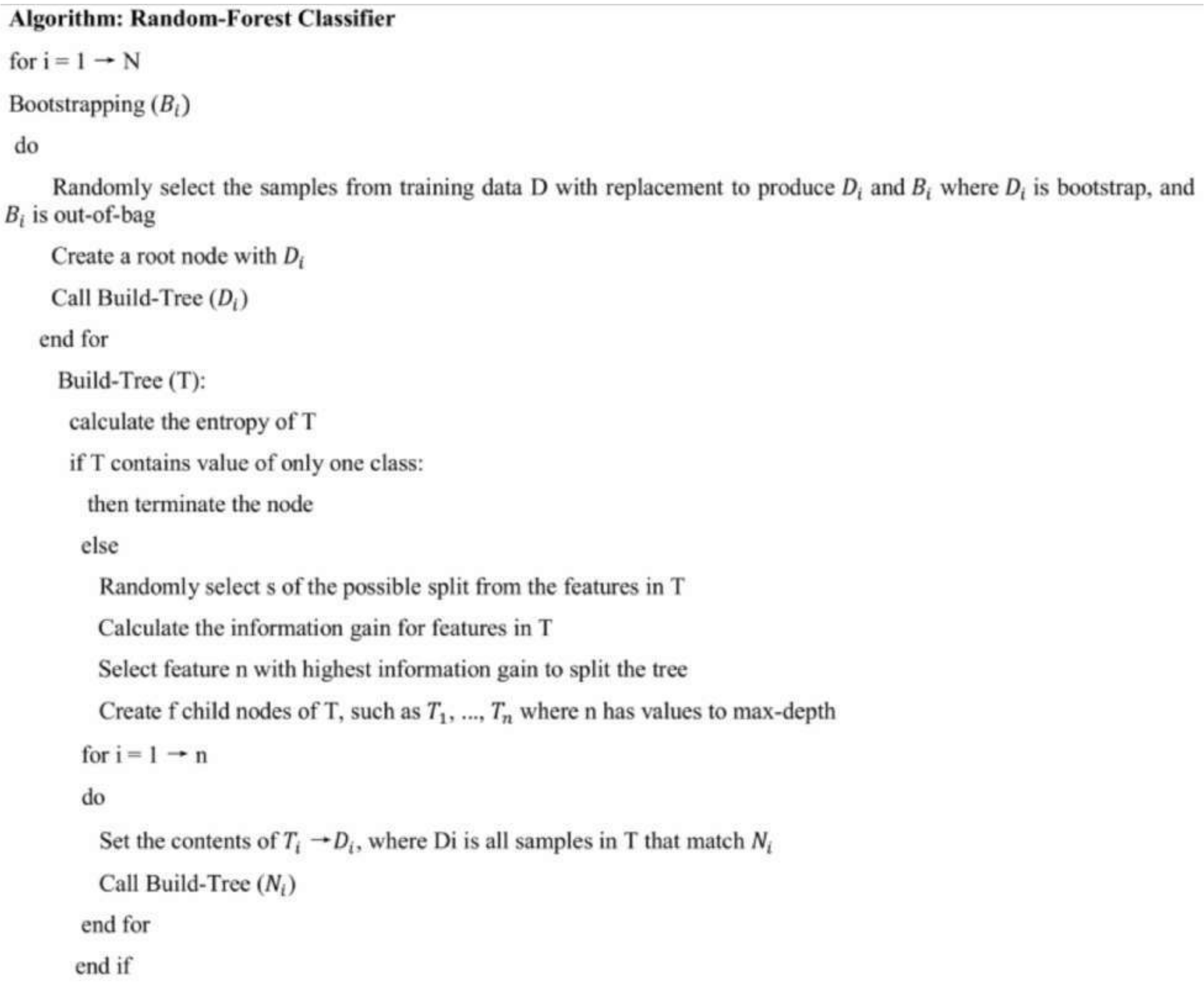
Should the data at the node be homogeneous, containing instances from only a single category, the node is deemed complete and given the label of that category. In contrast, when the node's data is heterogeneous, embodying multiple categories, the algorithm selects a subset of features at random. It then evaluates the potential of each feature to reduce entropy, a concept known as information gain, which is the difference in entropy before and after the split. The chosen feature is the one that maximizes this gain, thereby splitting the node and forming offspring nodes.

Each child node is populated with data corresponding to a specific feature value and inherits a subset of the dataset. This 'inheritance' is dictated by which feature value is present in the data. The `Build-Tree` function is then recursively applied to each child node. This recursive process is repeated, allowing the tree to expand until it reaches a predetermined level of complexity, as dictated by a maximum depth limit or when the nodes can't be split any further due to uniformity.

After all trees have been grown, they function as a committee, each casting a vote to classify new data points. The Random-Forest Classifier combines these votes, typically opting for the majority, to deliver the final verdict on the class of the new data. This ensemble approach leverages the collective intelligence of the forest, yielding predictions with higher confidence than a lone tree could provide.

Figure 43

Algorithm of Random Forest



```
Algorithm: Random-Forest Classifier
for i = 1 → N
Bootstrapping ($B_i$)
 do
     Randomly select the samples from training data D with replacement to produce $D_i$ and $B_i$ where $D_i$ is bootstrap, and
$B_i$ is out-of-bag
     Create a root node with $D_i$
     Call Build-Tree ($D_i$)
   end for
     Build-Tree (T):
       calculate the entropy of T
       if T contains value of only one class:
         then terminate the node
        else
          Randomly select s of the possible split from the features in T
          Calculate the information gain for features in T
          Select feature n with highest information gain to split the tree
          Create f child nodes of T, such as $T_1$, ..., $T_n$ where n has values to max-depth
        for i = 1 → n
        do
          Set the contents of $T_i$ →$D_i$, where Di is all samples in T that match $N_i$
          Call Build-Tree ($N_i$)
        end for
       end if
```

Support Vector Machine

TF-IDF paired with SVM forms an effective approach for categorizing text, such as assigning tags to Stack Overflow questions. Through TF-IDF, text is converted into a numerical score that reflects the significance of each word within a question while considering its commonness across all questions. This emphasizes words that are unique to a particular context, making them valuable for classification. SVM utilizes these scores to discern patterns, creating a hyperplane that optimally classifies the questions into distinct categories. In the context of Stack Overflow, this method will precisely identify and categorize questions by their content, employing the TF-IDF scores to separate tags with high-dimensional precision. Kernel functions within SVM allow this separation to be non-linear, accommodating the complex relationships within text data. Combining TF-IDF and SVM is anticipated to yield a sophisticated model capable of accurately tagging new questions, thus enhancing the organization and accessibility of information on the platform.

The displayed flowchart maps out a methodology for refining data features and subsequent classification in data science. It all starts with assembling a matrix that details how often certain terms appear across a set of documents. Following that, a calculation is applied to each term using TF-IDF, which is a technique to assess a word's relevance within a document set, balancing its frequency against how common it is across all documents.

With the recalibrated term-document matrix in hand, the next step involves sifting through these terms to pick out the ones that are statistically significant, determined by whether their TF-IDF value surpasses a preset variance threshold.

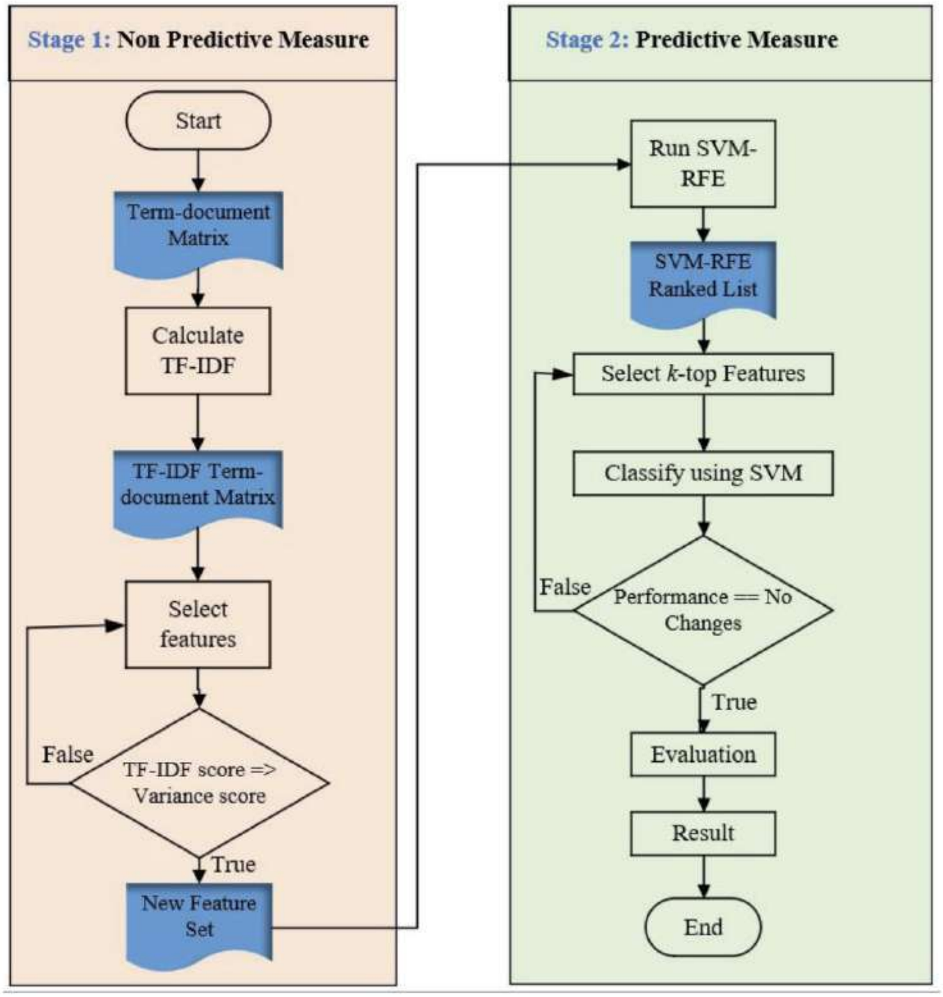
Transitioning into the predictive phase, the algorithm employs SVM-RFE, a method that uses support vector machines to iteratively evaluate and prioritize features based on their contribution to the model's accuracy. From this prioritized lineup, a predetermined number of top features are chosen.

These selected features are then employed to train a support vector machine, which is a model well-regarded for its prowess in categorizing data. The model's performance is monitored; if no improvement is observed, it suggests that the feature selection is optimal, otherwise, the process might be iterated with adjustments.

Upon reaching a plateau in performance enhancement, a thorough evaluation follows, culminating in the final results. This conclusion wraps up the procedure, yielding a set of features fine-tuned for effective data classification (Nafis & Awang, 2021).

Figure 44

Architecture of SVM



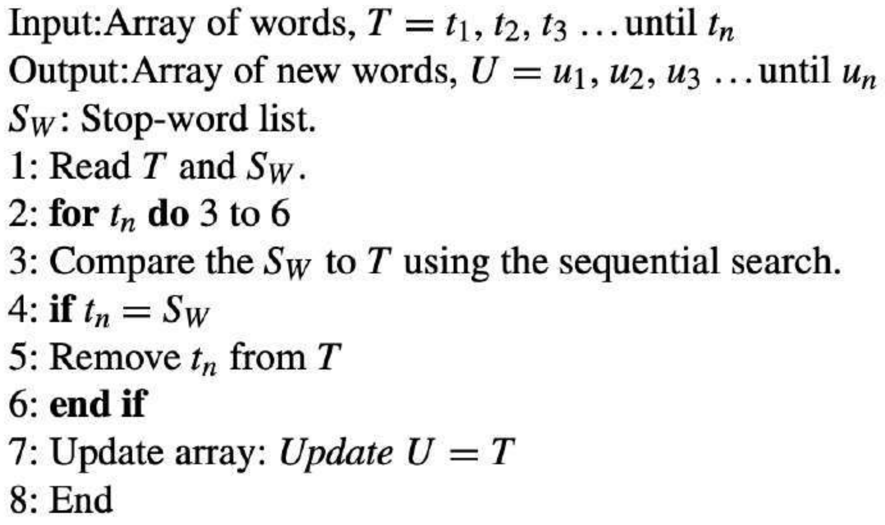
The provided algorithm takes in a list of words and sifts through it to weed out any non-essential words listed in a separate stop-word catalog. Initially, it ingests the list, referred to as T, which consists of various elements labeled t_1, t_2, t_3, ... t_n, along with a stop-word compendium, Sw. It proceeds to meticulously go through each term in T, scrutinizing it against the stop-word repository using a methodical, step-by-step comparison known as sequential search.

During this meticulous scrutiny, if it pinpoints a term in T that coincides with an entry in Sw, it is promptly excised from T. This elimination process is meticulously applied to each term, ensuring the entire list is thoroughly examined. Once this culling is complete, the algorithm compiles a new roster, U, which is populated with the refined selection of words that have survived the cull.

The end result is a pruned array, U, purged of all the superfluous stop-words, thereby streamlining the list for more advanced language processing tasks. This distilled list is now primed for use, having been cleansed of common and potentially extraneous words that could otherwise obscure meaningful analysis in language-centric computations.

Figure 45

Algorithm of SVM



```
Input:Array of words, $T = t_1, t_2, t_3 \ldots$until $t_n$
Output:Array of new words, $U = u_1, u_2, u_3 \ldots$until $u_n$
$S_W$: Stop-word list.
1: Read $T$ and $S_W$.
2: for $t_n$ do 3 to 6
3: Compare the $S_W$ to $T$ using the sequential search.
4: if $t_n = S_W$
5: Remove $t_n$ from $T$
6: end if
7: Update array: $Update\ U = T$
8: End
```

Ensemble method Bagging classifier and Decision tree classifier

The ensemble method employing bagging with decision trees is a robust technique to amplify the accuracy of predictive models. In this strategy, multiple decision trees are trained on varied subsets of the dataset, with these subsets created through resampling with replacement, a method known as bootstrap aggregating. This variety allows the ensemble to capture the breadth of the data's diversity, significantly reducing the chance of overfitting, which is a common drawback of using a single decision tree. In the context of sorting and prioritizing Stack Overflow responses, this method proves invaluable. It enables each decision tree to evaluate different segments and characteristics of the data, such as the content of an answer or the reputation metrics of its author. By consolidating the insights from each tree, the ensemble method arrives at a collective and more reliable ranking. Such a consolidated approach is especially effective in navigating the complex data landscape of textual information, guaranteeing that the final rankings accurately reflect a comprehensive assessment of an answer's relevance and utility to users.

The provided diagram outlines the bagging technique employed in creating ensemble models. It starts with a primary dataset \(T \), which forms the basis for training. This dataset undergoes a bootstrapping process that generates multiple smaller subsets, indicated by \(T_1, T_2, ..., T_n \). These subsets are created through random sampling with replacement, leading to varied yet overlapping training sets.

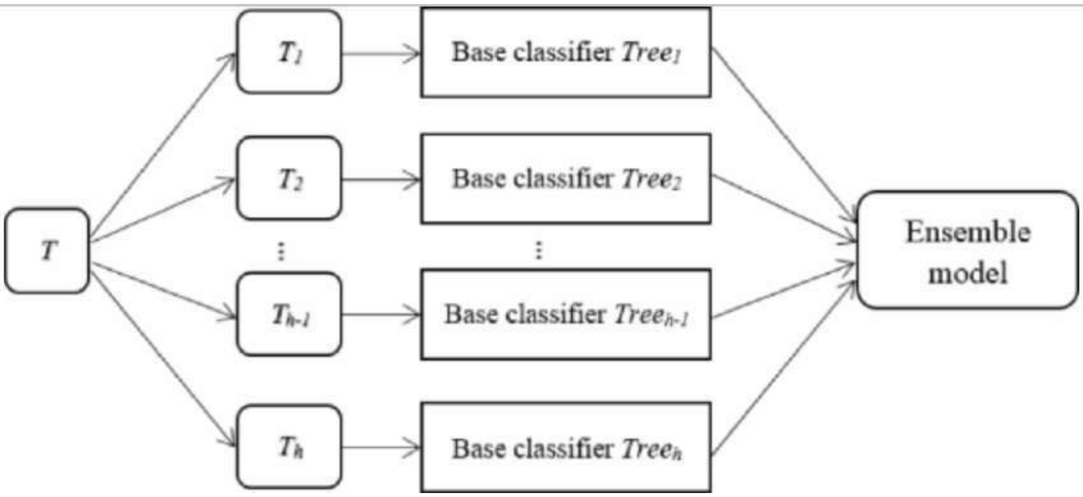
Each of these subsets is then utilized to train an individual decision tree, referred to as 'Base classifier \(Tree_i \)'. With each tree exposed to different slices of the dataset, they develop distinct rules and, consequently, unique perspectives on the data.

After the training phase, the individual predictions from each base classifier are combined. For classification tasks, this typically involves a voting system where the most frequently predicted class by the trees is selected as the final output. In contrast, for regression tasks, it might involve averaging the predictions.

This amalgamation results in a composite model that captures a broader range of data characteristics than a single decision tree could. By harnessing the predictive power of multiple classifiers, the ensemble model enhances accuracy and reduces the risk of fitting too closely to the training data. The ensemble's final output is expected to be more stable and reliable, reflecting the combined understanding of all the individual trees (Ma et al., 2017).

Figure 46

Architecture of Bagging



The algorithm depicted outlines a systematic procedure for creating simple belief decision trees as part of a bagging ensemble. The process commences with a root node that encompasses all instances within a specified uncertain training set. The initial step involves evaluating a stopping criterion that, if satisfied, designates the current node as a terminal point, returning a tree with the root node as its sole element.

If the stopping criterion is not fulfilled, the algorithm selects an attribute that maximizes the belief from the available features. This belief metric gauges the confidence level in the classification accuracy at that node. Upon identifying the optimal attribute for division, the data is split into subsets accordingly.

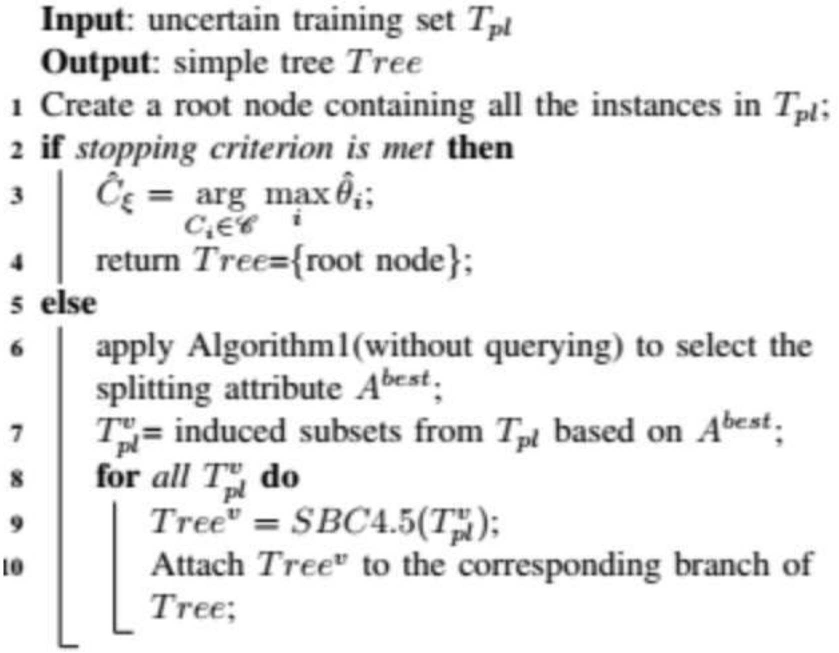
Subsequently, the algorithm recursively invokes itself on each subset, employing a belief decision tree learning subroutine to pinpoint the next attribute for splitting without additional querying. This recursive invocation leads to the expansion of the decision tree, with new nodes being integrated into the appropriate branches of the developing tree.

This recursive expansion persists until all nodes satisfy the stopping condition. The culmination of this process is a belief decision tree that has been meticulously constructed, accounting for uncertainties and optimizing belief at every decision juncture.

When set within the bagging framework, the tree induction is replicated across various bootstrap samples from the initial dataset, assembling multiple trees. These trees are collectively leveraged, often through a majority voting mechanism for classification tasks, to produce the ensemble model's final prediction. This method enhances the predictive stability and diminishes error variance by synthesizing insights from multiple decision trees.

Figure 47

Algorithm of Bagging with Decision Tree



```
Input: uncertain training set $T_{pl}$
Output: simple tree $Tree$
Create a root node containing all the instances in $T_{pl}$;
if stopping criterion is met then
    $\hat{C}_{\xi} = \underset{C_i \in \mathscr{C}}{\arg} \max_{i} \hat{\theta}_i$;
    return $Tree$={root node};
else
    apply Algorithm1(without querying) to select the
    splitting attribute $A^{best}$;
    $T_{pl}^{v}$= induced subsets from $T_{pl}$ based on $A^{best}$;
    for all $T_{pl}^{v}$ do
        $Tree^{v} = SBC4.5(T_{pl}^{v})$;
        Attach $Tree^{v}$ to the corresponding branch of
        $Tree$;
```

Sentence Transformer all-MiniLM-L6-v2

The all-MiniLM-L6-v2 model is a part of the MiniLM series of models which is a distilled version of larger transformer-based models BERT. The "L6" in the name indicates that this particular model has 6 layers, which is fewer than BERT model. This model is ideal for embedding tasks in natural language processing (NLP), like semantic search, clustering, and text similarity assessment. The model operates based on a distilled and optimized version of the transformer architecture, which is fundamental in modern natural language processing (NLP). The model workflow has two main components namely transformer mechanism and the distillation process.

Transformer Architecture Basis. Text input is first tokenized into a series of tokens. In transformers, these tokens are then converted into numerical vectors through an embedding process. Additional embeddings, like positional embeddings, are added to retain the sequence information of the tokens. The core of the transformer is the attention mechanism. It allows the model to focus on different parts of the input sequence when processing each token. This is crucial for understanding the context and relationships between words in a sentence. In a standard transformer, the attention mechanism is replicated multiple times in what's called multi-head attention. This allows the model to capture various aspects of the context in parallel. After attention processing, the output goes through feed-forward neural networks. These networks further process the data, applying nonlinear transformations. Transformers consist of several layers of multi-head attention and feed-forward networks. Each layer takes the output of the previous one and processes it further.

MiniLM Distillation Process. In distillation, a large, pre-trained model is used to guide the training of a smaller model. The smaller model here is the MiniLM. The smaller model learns to mimic the larger model. This involves aligning the output distributions of the smaller with those of the large model. The idea is to retain as much of the large model’s performance as possible. The all-MiniLM-L6-v2 models have fewer layers than their respective large models. This reduction is a key to their efficiency. Despite having fewer layers, the smaller model benefits from the distilled knowledge of the larger model, maintaining high performance. The training objective often involves matching the representations (like attention distributions or hidden states) produced by the small model with those from the large model, alongside traditional NLP training objectives. The all-MiniLM-L6-v2 model leverages the efficiency of the transformer architecture and the effectiveness of the distillation process to provide a powerful yet resource-efficient tool for semantic understanding.

Sentence Transformer all-mpnet-base-v2

All-mpnet-base-v2 is a language representation model from the Sentence Transformers library, and is intended to provide high-quality sentence embeddings. It is based on Microsoft's innovative pre-training method called MPNet architecture, which combines elements of permuted and masked language modeling to efficiently capture deep semantic meanings. This model performs exceptionally well in tasks such as semantic textual similarity, information retrieval, clustering, and text classification. It is optimized to produce embeddings that closely map semantically related sentences in the embedding space.

Operational Flow. An enhanced MPNet architecture created for semantic search jobs is the all-mpnet-base-v2 model. By employing a special pre-training technique that includes masking and permuting words to efficiently learn contextual information, it combines the advantages of BERT and XLNet. It is highly skilled at comprehending and contrasting textual context since it is specifically tailored for semantic search and produces dense phrase embeddings that capture semantic meanings. It works by processing queries to produce embeddings, which are subsequently analyzed using metrics like cosine similarity for pertinent document retrieval against a database of pre-computed embeddings. Because of its exceptional efficiency and deep comprehension of language, this model is well suited for tasks like matching queries to responses in a Q&A platform that are based on semantic content rather than just keywords.

4.2 Model Supports

Environment and Platform

Table 10

Environment and Platforms

Platform	Version	Purposes

Google Collab	2023-11-08 3.10	Google Colab runs on default Python version is 3.10.
Python	3.10	Data Preprocessing, Data Cleaning, Data Analysis with visualizations, Model Building.
GitHub	3.10.3	version management and monitor code modifications over time.

Tools

Table 11

Tools

Library		Method	Usage
Pandas	dataFrame	shape, head, describe, info, iloc	Loading the data, manipulating the data, finding data statistics for the modeling
Scikit-Learn	sklearn.feature_extraction.text	TfidfVectorizer	Converting text to a useful numerical

Library		Method	Usage
			representation for ML models.
	sklearn.model_selection	train_test_split	Split the data for training, validation and testing
	Sklearn.multiclass	OneVsRestClassifier	Modifying binary classifiers to handle tasks for multi-class classification tasks.
	Sklearn.linear_model	LogisticRegression	binary classification tasks.
	sklearn.preprocessing	MultiLabelBinarizer	transforming multiclass, multilabel target variables into a binary format
	sklearn.ensemble	RandomForestClassifier	performing both classification and

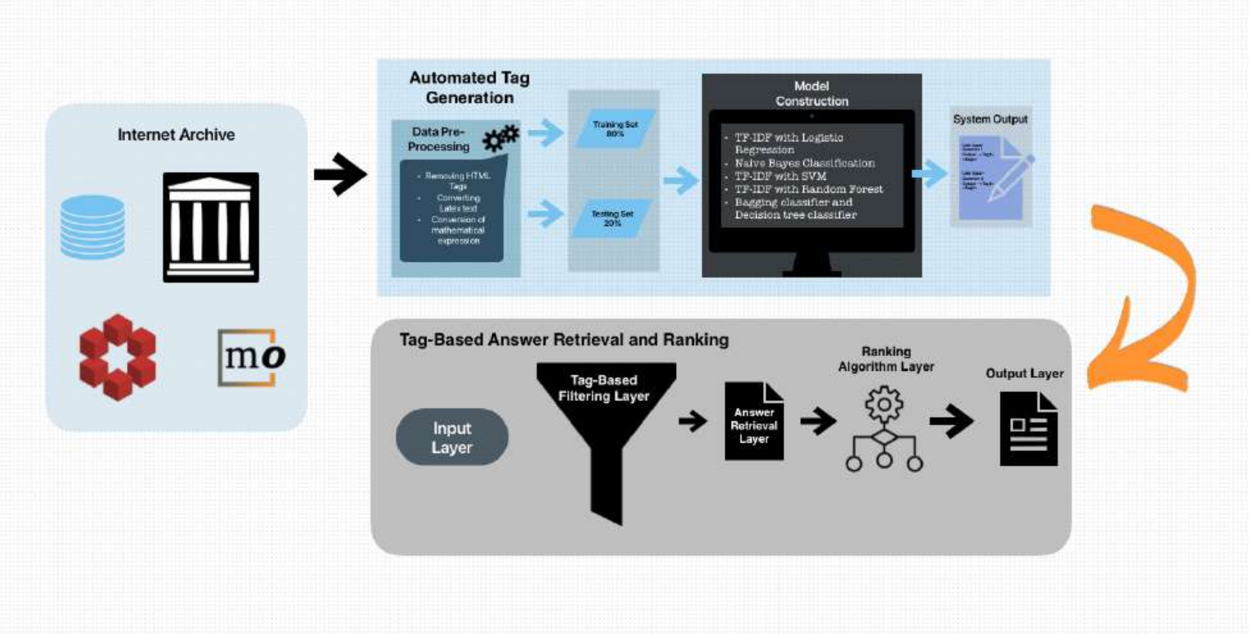
Library		Method	Usage
			regression tasks.
		BaggingClassifier	Reducing overfitting in ML models.
	sklearn.tree	DecisionTreeClassifier	Simple, interpretable classification tasks.
	sklearn.svm	LinearSVC	linear classification
	sklearn.naive_bayes	MultinomialNB	classification tasks involving discrete features, especially in text categorization.
	sklearn.metrics	precision_score, recall_score, hamming_loss, classification_report, accuracy_score, f1_score	Used for performance evaluation of models.
Seaborn		barplot	Used for data visualization

Library		Method	Usage
Numpy	numpy.ndarray	std, min, mean, abs, sqrt, sum, max, reshape, concatenate	Numerical calculations, Resize the array to the appropriate size, Concatenate several arrays into one array
Matplotlib	pyplot	plot, show, grid, figure, bar, title	Plotting numerous graphs

System Data Flow

Figure 48

System Data Flow



The diagram represents a complex system designed to enhance answer retrieval and ranking on a platform similar to Stack Exchange by first predicting tag generation. The system runs in multiple discrete but linked phases:

First, the system gathers a lot of information from the folders of Internet Archive.org namely, Math Stack Exchange, Math Educators, Mathematica and MathOverflow datasets. This data repository consists of several parts, including questions, responses, and related information, which comprises HTML text and mathematical formulas typed in LaTeX. The main goal is to gather a large dataset that will be used to power the prediction system's next phases. The prediction of tags portion of the system commences after the completion of data collection. This step, which includes the thorough data pre-processing necessary to clean and standardize the data, is fundamental to the system's operation. In this step, LaTeX text used for mathematical formulas is transformed into a machine-readable format and HTML tags are removed to isolate the text. This translation is essential because it guarantees that the machine learning algorithms that follow can understand the mathematical expressions.

To make sure models can generalize to new data, machine learning practitioners frequently separate pre-processed data into training and validation sets. 20% is set aside for validation and the remaining is 80%, used to train the model. This enables the machine learning models to be trained iteratively and the parameters to be adjusted based on performance metrics evaluated on the validation set.

Several machine learning models are used during the model creation phase, which is the central component of the tag generation phase. These include vectorization techniques for text classification using TF-IDF (Term Frequency-Inverse Document Frequency) in conjunction with logistic regression and SVM (Support Vector Machines) models. In addition, a decision tree classifier with bagging, a Random Forest method, and a Naive Bayes classifier are employed. These algorithms are trained to identify data patterns that match the right tags for the queries.

The system moves to the tag-based response retrieval and rating phase when tags are generated. After the tag-based filtering layer receives a query from the input layer, it sorts through the database and filters responses based on pertinent tags. The filtered responses are then retrieved by the answer retrieval layer and sent through the ranking algorithm layer. In order to guarantee that the most relevant answers are given priority, an algorithm ranks each response according to quality, relevancy, or other predetermined criteria. The ranking algorithms use the sentence transformer models namely, “all-MiniLM-L6-v2” and “all-mpnet-base-v2”. Another custom layer is used to compare the answer scores from the 2 transformer models and the original dataset.

The output layer is the last level of this multi-stage process, where the system shows the user a well-structured list of answers for the question, arranged according to a ranking the algorithm has established. This final depiction represents the main goal of the system, which is to perform the classification and retrieval of data based on accuracy and relevance, hence streamlining the user's information search.

4.3 Model Comparison and Justification

Logistic Regression

A statistical technique called logistic regression can be used to examine a dataset in which one or more independent factors affect the outcome. It is an efficient tool that can aid in understanding and foretelling systems or process behavior. One benefit of logistic regression is that it can handle a high number of predictor variables and is simple to execute and analyze. It can be used to handle nonlinear interactions between variables as well. The objective is to identify the link between the independent and dependent variables and to generate predictions about the future course of events using this relationship. A useful technique for assessing and forecasting binary outcomes is logistic regression. It can handle a lot of predictor variables and has a lot of benefits, including being simple to use and analyze (G et al., 2023). Logistic regression is a type of statistical analysis that is frequently brought into use for forward looking analytics and modeling and expands for implementation in machine learning and its widened applications. Logistic Regression is generally used in those instances where the dependent variable (target) is categorical in nature. In most cases, it is observed that linear regression is not the best-suited approach for classification problems because linear regression is limitless, and hence the logistic regression comes into the frame (Agarwal et al., 2022).

Naive Bayes Classifier

Bayes is a simple probabilistic prediction technique based on the Bayes rule with the assumption of strong independence. The strong independence in Bayes (especially Naive Bayes) on features is that a feature in a data is not related to the tone or not other features in the same data [8], choosing to use the Naïve Bayes algorithm because it has efficiency on both large and small data., and is not affected by irrelevant attributes, the Naïve Bayes Classifier can be applied to a fairly large number of data. The Naïve Bayes Classifier works well with isolated noise points and irrelevant attributes, and missing values are handled regardless of instances during probability estimation. The Naïve Bayes classification has better resistance to missing data than the supporting vector machine classification (Herlambang et al., 2021).

Support Vector Machines

Support Vector Machines (SVMs) are a set of supervised learning methods used for classification, regression, and outliers' detection. They are particularly known for their effectiveness in high-dimensional spaces and their versatility due to the kernel trick. In SVM, a hyperplane is a decision boundary that separates data points from different classes. For two-dimensional data, this hyperplane is a line. In higher dimensions, it becomes more complex (e.g., a plane in three dimensions). In its simplest form, the SVM algorithm creates a line or a hyperplane that separates the data into classes. When the data is not linearly separable, SVM uses a method known as the kernel trick to transform the input space into a higher-dimensional space. Here, a linear separator can be found. The goal of the SVM algorithm is to find a hyperplane that has the maximum margin, i.e., the greatest distance between data points of both classes. Optimization techniques are used to find the hyperplane that maximizes this margin.

SVM is a well-known machine learning algorithm that is capable of handling high-dimensional data and achieving promising performance such as text documents. SVM is a non-probabilistic algorithm which is able to separate data linearly and nonlinearly. However, in solving the binary class problem, a linear SVM is sufficient. SVM commits classification by constructing several *N*-dimensional hyper-planes that optimally split the data into two categories. Among the possible hyper-planes, the one where the distance of the hyper-plane from the closest data points (the ‘‘margin’’) is as large as possible is selected.

Random Forest Classifier

Random forests are the ensemble learning methods used for classification and regression problems. Random forest is an ensemble collection of multiple decision trees that are constructed from the bootstrap samples. While constructing each tree as high variance, random forests take the average of multiple different trees, which in turn reduces the high variance and leave us with a powerful classifier. Tree-based models like random forest are constructed from samples out of the dataset, picking a less features, and finding the value that makes the best split in our dataset. Breiman has shown that bagging in general, and Random Forest specifically, reduce the variance of a biased learner. Thus, for optimal performance, individual trees should minimize the bias error, which implies that they should not be restricted in size (Buschjäger et al., 2018).

Bagging with Decision Tree Classifier

"Bagging," combined with Decision Tree classifiers, forms a robust ensemble learning method. This approach is used to improve the stability and accuracy of machine learning algorithms, particularly decision trees, which are prone to overfitting on their own. A decision tree is a flowchart-like tree structure where an internal node represents a feature(or attribute), the branch represents a decision rule, and each leaf node represents the outcome. They are popular due to their simplicity and interpretability. While decision trees are intuitive, they tend to overfit the training data, especially when they are deep, capturing noise along with the signal. Ensemble methods like bagging often yield more accurate models than individual decision trees. This approach can be effective in handling datasets with imbalanced classes.

Sentence Transformer all-MiniLM-L6-v2

The all-MiniLM-L6-v2 is a distilled version of a larger transformer model BERT. It is designed to be small and efficient while retaining much of the performance of larger models. It utilizes the knowledge distillation process, where it learns to mimic the behavior of a larger, more complex 'teacher' model. It is highly efficient in terms of computational resources And has faster inference times due to fewer layers.

Sentence Transformer all-mpnet-base-v2

The all-mpnet-base-v2 model is designed for a broader range of NLP tasks and generally aims for higher performance at the cost of size and computational load. It uses a novel pre-training method that combines masked language modeling (like BERT) and permuted language modeling, aiming to understand the context better. It offers higher performance in a broader range of NLP tasks.

Justification

Logistic Regression is straightforward and easy to interpret, making it an excellent choice for situations where understanding the influence of individual variables is crucial. It also serves as an excellent baseline model in many classification problems due to its simplicity and speed in training and prediction. It can understand the relationship between the dependent binary variable and one or more independent variables, often used in situations where variables are linearly separable.

Naive Bayes classifiers are highly scalable and can handle large datasets efficiently. It is exceptionally effective in text classification problems, due to their ability to handle high-dimensional data. It is often used as a benchmark in Natural Language Processing tasks due to its simplicity and surprisingly good performance. While the independence assumption is a limitation, it can sometimes lead to surprisingly good performance, especially in cases where features are conditionally independent.

SVMs are known for their robustness, especially in avoiding overfitting, making them suitable for scenarios where precision is more important than interpretability. They tend to generalize well on unseen data, which is crucial for practical applications. They are effective in both classification and regression tasks, making them a good choice for many machine learning problems. The kernel trick allows them to solve non-linear problems, making SVM incredibly versatile and powerful for a wide range of classification problems.

Random Forest can capture complex interactions between features, suitable for both regression and classification tasks. They often yield high accuracy and work well with both categorical and numerical data. They provide insights into feature importance, which can be valuable in understanding the data. They can also handle large datasets with a higher dimensionality.

Bagging with Decision Trees reduces the risk of overfitting, a common problem with single decision trees, also combining multiple decision trees increases the overall accuracy of the model. They provide more stable and robust predictions than individual decision trees. They are also effective in handling datasets with high variance.

Logistic Regression and Naive Bayes are generally preferred for their simplicity and efficiency, especially in linearly separable or high-dimensional data scenarios. SVM is chosen for its robustness and effectiveness in high-dimensional spaces. Random Forest and Bagging with Decision Trees are favored for their accuracy and ability to handle complex datasets without overfitting.

The all-MiniLM-L6-v2 is suitable for environments with limited computational resources. Despite its smaller size, it delivers strong performance on various NLP tasks, especially in embedding generation for tasks like semantic search or text clustering.

The all-mpnet-base-v2 offers higher performance in a broader range of NLP tasks. Its pre-training approach allows it to understand context and semantics more effectively, which can be critical for complex NLP tasks. It can handle more complex language understanding tasks, including those involving nuanced contexts or subtleties in language. It is suitable for applications where the primary focus is on maximizing performance and accuracy in a variety of NLP tasks.

Table 12

Model Comparison

Model	Function	Strength	Type of Data	Space Complexity	Time Complexity	Training Time
Logistic Regression	Predicts probability of binary outcomes; good for yes/no type questions	Simple, efficient, interpretable	Linearly separable	Low; requires storage for coefficients for each feature	Low; training is fast, but can increase with the number of features	Fast for small to medium datasets, efficiency decreases with large datasets or large

						number of features
Naive Bayes Classification	Classifies data based on applying Bayes' theorem with strong independence assumptions	Scalable, works well with high-dimensional data	High-dimensional, text data	Low for simple implementations; higher for models with many features or categories	Low; generally fast due to the simplicity of calculations, but can increase with the number of features	Fast due to the simplicity of the algorithm, even with large datasets
SVM	Performs classification and regression by constructing hyperplanes in high-dimensional space	Effective in high-dimensional spaces, robust	High-dimensional, non-linear	High; needs to store support vectors and coefficients, increases with more support vectors	High; training time increases significantly with the number of samples and features, especially with kernel tricks	Slow, especially with large datasets and when using complex kernels
Random Forest Classifier	Ensemble of decision trees for classification and regression, reducing variance and avoiding overfitting	High accuracy, handles large datasets	Large datasets, mixed feature types	High; needs to store multiple decision trees, which increases with the number of trees and tree depth	Moderate to High; grows with the number of trees and the depth of each tree, but trees can be built in parallel	Moderate, training involves building multiple trees; faster with parallel computing

Bagging with Decision Tree	Combines multiple decision tree classifiers to improve prediction accuracy and control overfitting	Reduces overfitting, improves stability	Large datasets, high variance	Moderate to High; storage depends on the number of trees and their complexity	Moderate to High; depends on the number of trees and the complexity of the dataset, training can be parallelized	Moderate, depends on the number of trees; parallel processing can speed up training
all-MiniLM-L6-v2	Efficient language representatio, particularly for embeddings	High efficiency and speed; good performance with fewer resources	Suitable for various NLP tasks; excels in embedding generation	Lower due to fewer layers and parameters	Generally lower has faster inference due to simpler architecture	Generally low, benefits from distillation process
all-mpnet-base-v2	Advanced language understanding and representation	High accuracy and performance in diverse NLP tasks	Broad range of NLP tasks, including complex contextual understanding	Higher, more layers and parameters than MiniLM	Higher due to more complex model architecture	Longer due to the size and complexit y of the model

4.4 Model Evaluation Methods

To evaluate the efficacy and performance of machine learning models, model evaluation techniques are crucial. These methods involve metrics such as Precision, Recall, Hamming Loss, Classification Report, Accuracy, F1 scores and Cosine Similarity.

Precision

Precision is used as a metric to assess the proportion of accurate positive identifications to total identified positives. Essentially, it expresses how accurate a model or system is by measuring the proportion of correctly detected examples to all instances that are identified as positive, providing information on how discerning the classification or detection process is.

Recall

Recall plays a crucial function as a metric that indicates how well a system or model captures and accurately identifies all relevant positive examples from the total set of real positives. It captures the model's sensitivity by displaying the percentage of positives that are accurately detected in comparison to the total number of real positive cases.

Hamming Loss

Hamming Loss is employed when evaluating the difference between expected and observed binary or multilabel results. It provides a concise depiction of classification error by quantifying the percentage of wrongly predicted labels. Basically, Hamming Loss is used to estimate how accurate predictions are made when an instance can have several labels given to it, capturing the total label assignment difference.

Accuracy

By measuring the proportion of properly predicted instances to total instances, accuracy is a key metric used to evaluate the overall correctness of a model. It offers a clear and thorough assessment of the model's ability to accurately predict all classes or outcomes.

F1 score

Precision and recall are balanced in classification tasks by the F1 Score, which acts as a combined metric. By taking into account both false positives and false negatives, it provides a fair evaluation and a single numerical number that unifies the trade-off between precision and recall.

Cosine Similarity

Cosine similarity is a metric used to determine how similar two vectors are in machine learning and information retrieval. By calculating the cosine of the angle formed by these vectors, a value ranging from -1 to 1 is obtained. A value of -1 denotes a complete opposite direction, 0 orthogonality (no similarity), and a value of 1 shows the identical orientation.

Table 13

Evaluation Metrics Formulas

Name	**Formula**	**Variable**
Precision	$TP\ /\ (TP\ +\ FP)$	**True Positives (TP):** Events that the model accurately predicts as positive but are in fact positive. **False Positives (FP):** Occurrences that the
Recall	TP / (TP + FN)	**True Positives (TP):** Instances where the model accurately predicts a positive outcome even though the data is truly

Name	Formula	Variable
		positive. **False Negatives (FN):** When a model predicts negatively when it is actually positive.
Hamming Loss	$1/N\ [\sum i = 1\ to\ N]\ 1/L\ [\sum j = 1\ to\ L]\ I(y\ ij = \hat{y}\ ij)$	$\boldsymbol{N}$: Number of instances in the dataset. $\boldsymbol{L}$: Total number of labels. y ij : Binary indicator of if label j is associated with instance i in the true data. $\hat{y}\ ij$: Binary indicator of if label j is predicted for instance i by the model. $\mathbf{I(\cdot)}$: Indicator function, which returns 1 if the condition inside is true and 0 otherwise.

Name	Formula	Variable
Accuracy	(TP + TN) / (TP + TN + FP + FN)	**True Positives (TP):** Instances that are positive & correctly predicted as positive by the model. **True Negatives (TN):** Instances that are negative & correctly predicted as negative by the model. **False Positives (FP):** Instances that are negative but are incorrectly predicted as positive by the model. **False Negatives (FN):** Instances that are positive but are incorrectly predicted as negative by the model.
F1 Score	2 * Precision * Recall / (Precision + Recall)	**Precision** : Ratio of true positives to the sum of

Name	Formula	Variable
		true positives and false positives. **Recall** : Ratio of true positives to the sum of true positives and false negatives.
Cosine Similarity	CS = A.B / (||A|| ||B||)	A, B are vectors. A.B is dot product of vectors. ||A|| ||B|| are the magnitudes of the vectors.

Various criteria such as accuracy, F1 score, hamming loss, precision, and recall are employed to evaluate the effectiveness of classification models. The precision metric highlights the model's ability to prevent false positives by focusing on the ratio of accurately predicted positive cases to the total anticipated positives. Conversely, recall emphasizes the model's capacity to catch all positive instances by calculating the ratio of correctly predicted positive instances to all real positives. Hamming loss is appropriate for multi-label classification issues since it calculates the percentage of wrongly predicted labels over all instances. The F1 score is especially helpful in circumstances when maintaining a balance between precision and recall is essential since it integrates the two metrics into a single metric and provides a balanced evaluation by taking false positives and false negatives into account.

References

Prediction of Graduation with Naïve Bayes Algorithm and Principal Component Analysis (PCA) on Time Series Data. (2021, August 3). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9527443/

Nafis, N. S. M., & Awang, S. (2021). An enhanced hybrid feature selection technique using term Frequency-Inverse document frequency and support vector Machine-Recursive feature elimination for sentiment classification. *IEEE Access*, *9*, 52177–52192. https://doi.org/10.1109/access.2021.3069001

Ma, L., Sun, B., & Li, Z. (2017). Bagging likelihood-based belief decision trees. *Bagging Likelihood-based Belief Decision Trees*. https://doi.org/10.23919/icif.2017.8009664

Herlambang, W. D., Laksitowening, K. A., & Asror, I. (2021). Prediction of Graduation with Naïve Bayes Algorithm and Principal Component Analysis (PCA) on Time Series Data. *Prediction of Graduation With Naïve Bayes Algorithm and Principal Component Analysis (PCA) on Time Series Data*. https://doi.org/10.1109/icoict52021.2021.9527443

G, H. V., Preethi, S., P, S. R., & S, E. B. C. (2023). Flood Prediction Using Logistic Regression. *Flood Prediction Using Logistic Regression*. https://doi.org/10.1109/iccpct58313.2023.10245832

Buschjäger, S., Chen, K., Chen, J., & Morik, K. (2018). Realization of Random Forest for Real-Time Evaluation through Tree Framing. *Realization of Random Forest for Real-Time Evaluation Through Tree Framing*. https://doi.org/10.1109/icdm.2018.00017

Arya, A., Sehgal, M., Bhatia, N., Juneja, S., & Koundal, D. (2023). Heart disease prediction with machine learning and virtual reality. In *Elsevier eBooks* (pp. 209–228). https://doi.org/10.1016/b978-0-323-98381-5.00011-8

Agarwal, N., Srivastava, R., Srivastava, P., Sandhu, J., & Singh, P. P. (2022). Multiclass Classification of Different Glass Types using Random Forest Classifier. *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*. https://doi.org/10.1109/iciccs53718.2022.9788326

Abdulazeez, A. M., Falah, Y., Ahmed, F. Y. H., & Zeebaree, D. (2021). Intrusion detection systems based on machine learning algorithms. *ResearchGate*. https://www.researchgate.net/publication/353906529_Intrusion_Detection_Systems_Based_on_Machine_Learning_Algorithms