**Sophie Dang**

**C964 Capstone: SVM Spam Filter Proposal**

**Student ID: 009522974**

**06/15/23**

**Part A: Project Proposal for Business Executives**

Sophie Dang

Data Scientist

Chewy, Inc.

1110 112th Ave NE,

Bellevue, WA 98004

Satish Mehta

Chief Technology Officer

Chewy, Inc.

7700 West Sunrise Blvd.

Plantation, FL 33322

Dear Mr. Mehta,

Subject: Letter of Transmittal - Proposal for Implementation of Data Project

I am pleased to present this proposal implementation of a data product that will revolutionize our decision-making process by implementing an advanced spam filter using machine learning techniques, specifically the Support Vector Machines (SVM) algorithm. As senior, non-technical managers and executives, your support and understanding are crucial to the success of this project. This letter aims to provide a concise overview of the proposal's key components and its significant benefits.

The proposed solution offers benefits, including reduced exposure to spam emails, enhanced communication security, streamlined decision-making, and improved collaboration and organizational efficiency. The spam filter will seamlessly integrate into our existing email infrastructure and continuously learn and adapt to evolving spam patterns. The project methodology includes data collection, preprocessing, model development, evaluation, integration, and deployment. The estimated budget for the project will be an initial amount of $26,630.00 to create the proposed solution and subsequently, an estimated $10,000 per month for a machine learning engineer to provide maintenance, support, fine-tune the spam filter, and address any technical challenges that may arise.

The implementation of the spam filter will have positive impacts on stakeholders while adhering to ethical and legal considerations. As the data scientist leading the project, I have the necessary expertise and experience in implementing machine-learning solutions. My proficiency in machine learning algorithms, including SVM, and my ability to handle complex data environments make me well-suited to execute this project successfully. With the support of non-technical managers and executives, I am confident in the successful implementation of this data product.

In conclusion, the implementation of the proposed spam filter data product will provide our organization with a robust solution to combat the ever-increasing problem of spam emails. By investing in this project, we will improve productivity, communication security, and decision-making processes, ultimately leading to a more efficient and competitive organization. I invite you to review this proposal and consider the significant benefits it offers. Your support and endorsement are crucial for the successful implementation of this project.

Thank you for your time and consideration. I am available to discuss any questions or concerns you may have.

Sincerely,

Sophie Dang

Data Scientist

Chewy, Inc.

1110 112th Ave NE,

Bellevue, WA 98004

**Project Proposal**

**Overview:**

The project aims to develop and implement a spam filter using the Support Vector Machine (SVM) algorithm. The main objective is to identify and distinguish between spam (illegitimate) and ham (legitimate) messages accurately and efficiently. The implementation of the spam filter will have positive impacts on various stakeholders within the organization. Employees will enjoy increased productivity and reduced distractions from spam emails, while management will benefit from enhanced data security and improved resource allocation. The organization will experience an increase in collaboration and more efficient communication and gain a competitive edge in the industry.

**Summary of the Problem:**

In today's data-driven world, we face the increasing challenge of managing and extracting value from large amounts of information. We are inundated with data, especially in the realm of email communications. The issue we face is the growing threat of spam emails, which hinders productivity, exposes our systems to potential security risks, and undermines the efficiency of our communication channels.

**Description of the Data Product and Customer Benefits:**

Our proposed data product is an advanced spam filter that uses state-of-the-art machine learning techniques, specifically the Support Vector Machine (SVM) algorithm. This solution will provide our organization with a highly accurate and efficient mechanism to differentiate between spam and legitimate emails, ensuring that our employees receive only the messages that matter. By implementing this data product, we will experience the following benefits:

* Drastically reduced exposure to spam emails, leading to improved productivity and reduced time spent sorting through irrelevant messages.
* Enhanced communication security, safeguarding our systems, and protecting sensitive information.
* Streamlined decision-making process, allowing our employees to focus on critical tasks and essential correspondences.
* Increased confidence in our email communication channels, leading to improved collaboration and organizational efficiency.

**Outline of the Data Product:**

The project aims to develop a highly accurate and efficient spam filter by using the Support Vector Machine (SVM) algorithm. This algorithm will be trained using a supervised reinforcement approach, leveraging a dataset that includes labeled examples of both spam and non-spam emails/messages. By training on such data, the model will effectively learn the distinctive characteristics of each class (GeeksforGeeks, 2023). The successful implementation of this spam filter brings numerous advantages to the company, the most significant being its ability to distinguish between spam and legitimate emails/messages with exceptional accuracy and efficiency, ensuring that our employees receive only essential and relevant communications and minimizing the disruption caused by unwanted messages.

**Description of the Data Used for Construction:**

The project uses the "SMS Spam Collection Dataset" obtained from kaggle.com, which contains 5,574 English SMS messages classified as either legitimate (ham) or illegitimate (spam) (SMS Spam Collection Dataset, 2016). A new column named "spam" was added to the dataset to facilitate binary classification. This column assigns a value of 1 to rows labeled as 'spam' and 0 to rows labeled as 'ham', effectively distinguishing between spam and non-spam messages.

The dataset initially shows a significant class imbalance, with 87% of the messages labeled as ham and only 13% labeled as spam (SMS Spam Collection Dataset, 2016). To address this issue, the Synthetic Minority Oversampling Technique (SMOTE) was employed. SMOTE oversamples the minority class (spam messages) to balance the class distribution. The dataset was divided into training and testing sets. SMOTE was then applied exclusively to the training data. By oversampling the training data, the model can learn from a more balanced representation of spam and ham messages. SMOTE was only applied to the training data and not the testing data. This approach ensures that the testing data retains its original class distribution, allowing for an accurate evaluation of the performance of the spam filter model (Maddali, 2022). Additionally, the dataset underwent a privacy-focused step to remove any sensitive or personally identifiable information. This precautionary measure ensures compliance with legal and ethical considerations related to data privacy and protection.

**Objectives and Hypotheses of the Project:**

The primary goal of this project is to create and deploy an effective spam filter that significantly reduces the volume of spam emails reaching our employees' inboxes. The hypotheses of the project include:

* The SVM algorithm will accurately classify emails as either spam or non-spam.
* The implementation of the spam filter will lead to a substantial decrease in the number of spam emails received by our employees.
* The spam filter will have a negligible false positive rate, ensuring that legitimate emails are not mistakenly classified as spam.

**Outline of the Project Methodology:**

**The project methodology will consist of the following steps:**

1. **Data collection:** Gather a diverse dataset of labeled emails/messages for training and evaluation purposes. Ensure all sensitive or personally identifiable information is removed to ensure privacy and adhere to legal and ethical considerations.
2. **Data preprocessing:** Cleaning and transforming the data to ensure its compatibility with the SVM algorithm. Data preprocessing techniques include applying lemming to the dataset, removing non-alphabetic characters, converting characters to lowercase, tokenizing the text, and removing stop words. Furthermore, address any class imbalance within the dataset to ensure the spam filter can learn effectively from both classes and make accurate predictions (Maddali, 2022).
3. **Model development:** Splitting the dataset into training/testing sets: the training set will include 80% of the dataset, and the test set will contain 20%. Train the SVM model using the preprocessed data and fine-tune the hyperparameters for optimal performance if necessary.
4. **Model evaluation:**

* Calculate evaluation metrics such as accuracy, precision, recall, and F1 score to assess the model's performance (Maddali, 2022).
* Predict the classes (spam or non-spam) for the test set using the trained SVM spam filter model.
* Examine the confusion matrix to analyze the model's performance in predicting unseen spam/non-spam messages from the test data.
* Fine-tune the model, if needed, based on the evaluation results to achieve the desired performance (Maddali, 2022).
* Generate a classification report summarizing the performance metrics for both classes (Maddali, 2022).
* Gather feedback from end-users (e.g., employees within the organization) regarding the accuracy, usefulness, and spam filter's usability.

1. **Integration and deployment**: Integrate the trained and tested spam filter into the company’s existing email infrastructure and conduct thorough testing to ensure seamless functionality.

**Funding Requirements:**

We estimate that the initial funding required for this project, including data collection, preprocessing, model development, and deployment, will be $26,630. This total includes the salaries and costs of laptops for software developer I ($3,500 for 70hrs), machine learning engineer I ($4,800 for 80hrs), data scientist I ($4,800 for 80hrs), project manager I ($3,000 for 60hrs), laptop for software developer I ($2,500), laptop for machine learning engineer I ($3,000), laptop for a data scientist I ($3,000), and laptop for project manager I ($2,000). Additionally, Microsoft Suite costs $30.00 monthly for three licenses and is used to create and edit documentation.

The development environment for this project is Google Codelabs. We will use Python and machine learning libraries such as scitkit-learn and NLTK. Google Codelabs, Python, and machine learning libraries are all open-sourced and free, so there’s no cost in using them. The initial development cycle is expected to last four weeks, from the start of the project to its deployment. Also, it’s estimated to cost roughly $10,000 per month to pay a machine learning engineer to monitor the spam filter, ensuring its smooth functioning and addressing any technical issues that arise. This investment will yield significant returns in terms of improved productivity, enhanced communication security, and streamlined decision-making processes.

**Impact on Stakeholders:**

The implementation of the spam filter will positively impact multiple stakeholders within our organization. Employees will experience increased productivity and reduced distractions caused by spam emails. Management will benefit from enhanced data security and improved resource allocation. Additionally, the organization will achieve improved collaboration, more efficient communication, and a competitive edge in the industry.

**Benefits that stakeholders in the organization will enjoy including the following:**

* **Enhanced Data Privacy:** The spam filter will help safeguard sensitive information by minimizing the risk of phishing attacks and unauthorized access to confidential data ensuring data privacy and protecting the organization from potential security breaches.
* **Time and Cost Savings:** With fewer spam emails reaching employees' inboxes, they can allocate their time and energy to important tasks, leading to increased productivity. Moreover, the reduced time spent sorting through spam emails translates into cost savings for the organization.
* **Improved Reputation:** By effectively filtering out spam emails, the organization can maintain a professional image and reputation. Clients, partners, and customers will perceive the organization as efficient and trustworthy in their communication practices.
* **Regulatory Compliance:** Implementing a robust spam filter can help the organization comply with industry-specific regulations and data protection laws, ensuring adherence to legal requirements and minimizing the risk of non-compliance penalties.
* **Reduced Stress and Frustration:** Constantly being bombarded with spam emails can be frustrating for employees. Implementing the spam filter alleviates this frustration, leading to a more positive work environment and increased job satisfaction.
* **Streamlined Workflows:** The spam filter's accurate classification of emails allows for streamlined workflows, as employees can quickly identify and prioritize important messages without being overwhelmed by irrelevant content. This facilitates efficient collaboration and decision-making processes.
* **Business Continuity:** Spam emails can pose a significant threat to the stability and continuity of business operations. By minimizing the impact of spam, the organization can ensure uninterrupted workflow and mitigate potential disruptions caused by malicious content.

**Ethical and Legal Considerations:**

We understand the importance of handling sensitive data responsibly and ethically. Throughout the project, we will strictly adhere to all legal and regulatory requirements regarding data protection and privacy. Measures will be implemented to anonymize and sanitize the data, ensuring that no personally identifiable information is at risk. Communication about the project will be conducted with utmost care, ensuring the confidentiality of sensitive information.

**When working with and communicating about sensitive data, the project team will:**

* **Ensure Compliance:** Adhere to relevant data privacy regulations and guidelines (e.g., GDPR, HIPAA) while handling customer reviews containing personal information (Data Protection and Privacy: How to Protect User Data, 2023).
* **Anonymize Data:** Remove or encrypt personally identifiable information (PII) from customer reviews to protect customer privacy (Data Protection and Privacy: How to Protect User Data, 2023).
* **Secure Storage:** Employ appropriate security measures to protect sensitive data during storage and transmission, such as encryption and access controls (Data Protection and Privacy: How to Protect User Data, 2023).
* **Limited Data Access:** Limit access to sensitive data only to authorized personnel who need it for the project's execution, ensuring confidentiality and integrity (Data Protection and Privacy: How to Protect User Data, 2023).

**Your Expertise Relevant to the Proposed Solution:**

As an experienced data scientist with a track record of successfully implementing machine learning solutions, I possess the necessary expertise to oversee the development and deployment of the spam filter data product. My extensive knowledge of machine learning algorithms, including the SVM technique, and my ability to navigate complex data environments equip me with the skills to execute this project effectively. I am confident that my expertise, combined with the support of our non-technical managers and executives, will lead to the successful implementation of this data product.

**Part B: Project Proposal**

**Executive Summary: Data Product for IT Professionals**

This executive summary provides an overview of our proposed data product, highlighting its benefits and outlining the primary requirements and considerations for its design, development, and implementation. As IT professionals, your expertise is crucial in evaluating and supporting the success of this endeavor.

**Decision-Support Problem or Opportunity:**

We aim to address the challenge of efficiently managing and classifying incoming email communications within our organization. The current process is time-consuming and prone to errors, resulting in reduced productivity and potential security risks. Our data product, an advanced spam filter using Support Vector Machines (SVM) algorithm, presents an opportunity to streamline email classification, improve productivity, and enhance communication security.

**Customers and Fulfilling Their Needs:**

Our primary customers are our organization's employees, who receive a significant volume of emails daily. This data product will fulfill their needs by significantly reducing their exposure to spam emails, enabling them to focus on important messages and critical tasks. The improved email filtering will enhance productivity, communication efficiency, and overall job satisfaction.

**Existing Gaps in Data Products:**

The current email filtering system lacks the accuracy and efficiency required to effectively differentiate between spam and legitimate emails. Existing solutions often result in false positives or false negatives, leading to frustration and decreased confidence in email communication. Our data product aims to bridge these gaps by leveraging advanced machine learning algorithms to provide a more accurate and reliable email classification system.

**Data Availability and Collection:**

The project uses the "SMS Spam Collection Dataset" from kaggle.com, containing 5,574 English SMS messages categorized as ham or spam (SMS Spam Collection Dataset, 2016). A new column named "spam" was added, assigning 1 to spam messages and 0 to ham messages for binary classification. To address the class imbalance issue, SMOTE (Synthetic Minority Oversampling Technique) was used to oversample the minority class (spam). The dataset was split into training and testing sets, and SMOTE was applied only to the training data to balance the class distribution (Maddali, 2022). All sensitive or personally identifiable information was removed from the dataset to ensure compliance with legal and ethical considerations related to data privacy and protection.

**Methodology for Design and Development:**

The project will use the Agile model, specifically the Scrum framework, to implement the spam filter. Scrum is a suitable methodology for this project due to its emphasis on flexibility, collaboration, and continuous improvement. It allows for adaptive planning and encourages regular feedback from stakeholders, which is particularly beneficial when developing a spam filter that may require evolving requirements and continuous refinement.

**The Scrum framework is divided into nine main steps:**

**1. Define product vision:** Clearly establish the goals and purpose of the spam filter. The primary objectives of the spam filter are to distinguish between spam and legitimate emails effectively and efficiently, ensuring that employees receive only relevant messages (Pr, 2023). To achieve this, the filter will be developed using machine learning techniques, specifically utilizing the Support Vector Machine (SVM) algorithm. The expected benefits of the spam filter include:

* Drastically reduced exposure to spam emails, leading to improved productivity and reduced time spent sorting through irrelevant messages.
* Enhanced communication security, safeguarding our systems and protecting sensitive information.
* Streamlined decision-making process, allowing our employees to focus on critical tasks and important correspondences.
* Increased confidence in our email communication channels, leading to improved collaboration and organizational efficiency.

**2. Create Product Backlog:** Create a product backlog by gathering requirements from our stakeholders. Each requirement should be represented as a user story, focusing on the value it brings to the end users (organization's employees). Prioritize the user stories based on their importance and estimated value (Pr, 2023).

**3. Sprint Planning:** Select a set of user stories from the product backlog to be implemented in the upcoming sprint. Collaborate with the development team to determine the tasks required to complete each user story. Estimate the effort required for each task (Pr, 2023).

**4. Sprint Execution:** The development team works on the selected user stories during the sprint. Daily Scrum meetings are held to discuss progress, address challenges, and plan the day's work. The development team self-organizes and collaborates to complete the tasks (Pr, 2023).

**5. Sprint Review:** At the end of the sprint, conduct a sprint review meeting to demonstrate the completed user stories to stakeholders. Gather feedback and incorporate it into the product backlog for future sprints. The focus is on obtaining feedback and validating the work done (Pr, 2023).

**6. Sprint Retrospective:** Hold a sprint retrospective meeting to reflect on the sprint and identify areas for improvement. Discuss what went well, what could be done better, and any adjustments needed for future sprints. The focus is on continuous improvement and learning (Pr, 2023).

**7. Repeat and Iterate:** Start a new sprint by selecting additional user stories from the product backlog and repeating the sprint planning, execution, review, and retrospective steps. Continuously refine and reprioritize the product backlog based on feedback and changing requirements (Pr, 2023).

**8. Release and Deployment:** When the spam filter reaches a stable and valuable state, plan its release and deployment. This includes preparing the release package, conducting final testing, and deploying the spam filter to the production environment. User training and communication should accompany the release (Pr, 2023).

**9. Ongoing Maintenance and Support:** After deployment, provide ongoing maintenance and support for the spam filter. This includes addressing bugs and issues, monitoring performance, applying security updates, and providing user support. Continuously gather feedback and make improvements based on user needs (Pr, 2023).

**Deliverables and Implementation Plan:**

**Project Deliverables:**

The following deliverables will be provided upon the completion of this project:

1. **Trained Spam Filter Model:** Spam filter model that accurately and efficiently differentiates between spam and legitimate messages. It scans incoming emails and assigns a spam probability score to each message. Based on this score, the system will automatically filter out spam emails, ensuring that only legitimate and relevant messages reach our employees' inboxes. The filter will continuously learn and adapt to evolving spam patterns, ensuring optimal performance and accuracy.
2. **Machine Learning Model Code:** The code to train the machine learning model for spam detection. This code will include data preprocessing, feature extraction, model training, and evaluation.
3. **User Interface Code:** The code for creating a user interface, which will be a user-friendly and simple command line interface that allows users to interact with the spam filter. The interface allows users to enter messages and have the filter determine whether the message is spam or legitimate (ham).
4. **Training Data:** The dataset used to train the machine learning model. This dataset will consist of labeled emails, including both spam and legitimate messages, ensuring a diverse and representative training sample.
5. **Documentation:** Comprehensive documentation that explains the data collection process, data preprocessing techniques, machine learning model architecture, and user interface functionality. This documentation will provide a clear understanding of the solution's implementation details.
6. **User Manual:** A user manual that guides users on how to effectively use the spam filter. It will include instructions on installation, setup, and best practices for maximizing the performance and accuracy of the filter.

**Implementation Plan:**

**The implementation of the spam filter will follow the steps outlined below:**

1. **Data Collection:** The project will collect a labeled email dataset from kaggle.com, ensuring diversity and representation.
2. **Data Preprocessing and Cleaning:** The collected data will undergo preprocessing and cleaning to enhance its quality and integrity, which includes removing stop words, applying lemming to reduce words to their base or dictionary form, handling missing values, and normalizing the text data. Feature extraction techniques will be used to represent emails in a suitable format for training the model. These steps ensure data is refined and ready for effective model training and analysis. Moreover, any class imbalance within the dataset will be addressed using SMOTE to either oversample the minority class or under sample the majority class to ensure that the spam filter model can accurately classify both the majority and minority classes, leading to improved performance and better real-world applicability (Maddali, 2022).
3. **Model and User Interface Development:** The code development phase involves creating the necessary code to train a machine-learning model for spam detection. The chosen machine learning algorithm, Support Vector Machines, will be used to train the model. Furthermore, a user-friendly command line interface will be designed and developed, allowing users to input messages and obtain spam or legitimate classifications from the model.
4. **Model Training and Evaluation:** Split the preprocessed data into training and testing sets. The training set will include 80% of the dataset, and the testing set will contain 20%. Train the machine learning model using the training data and evaluate its performance using the testing data. Fine-tune the model, adjust hyperparameters, and assess its accuracy, precision, recall, and F1 score. Iterate as necessary to improve results.
5. **Documentation and Solution Finalization:** Prepare comprehensive documentation that details the data collection process, data preprocessing techniques, machine learning model architecture, and user interface functionalities. Provide clear explanations and code examples where necessary. Review and refine the documentation to ensure its accuracy and completeness.
6. **User Manual Creation:** Develop a user manual that provides step-by-step instructions on how to install, configure, and use the spam filter. Include guidelines on interpreting spam probability scores, managing filters, and customizing settings for individual users.
7. **Delivery and Deployment:** Compile all the deliverables, including the machine learning model code, user interface code, training data, documentation, and user manual. Perform a final review to ensure the solution's completeness and accuracy. Deliver the package to the client or deploy it within the organization's infrastructure. Provide any necessary support during the deployment process.

Throughout the implementation process, regular communication, and collaboration with stakeholders, including users and IT personnel, will be essential to gather feedback, address concerns, and ensure the spam filter meets the organization's requirements and expectations.

**Validation and Verification:**

**To ensure that the developed spam filter meets requirements and end-users needs, the following evaluation metrics and methods will be used:**

1. **Accuracy:** Accuracy measures the overall correctness of the spam filter's predictions by comparing the number of correct predictions with the total number of predictions. It helps assess the effectiveness of the filter in correctly classifying emails as spam or non-spam (Maddali, 2022).
2. **Precision, Recall, and F1 Score:** Precision, recall, and F1 score are commonly used metrics to evaluate a spam filter's performance. Precision measures the filter's ability to correctly identify spam without classifying legitimate emails as spam. Recall measures the filter's ability to identify all spam emails correctly. The F1 score provides a balanced measure of both precision and recall. These metrics assess the accuracy and effectiveness of the spam filter in identifying spam while minimizing false positives and false negatives (Maddali, 2022).
3. **User Feedback:** Gathering feedback from end-users (the company’s employees) is crucial to evaluating the spam filter's performance and usability. Feedback can be collected through surveys, user interviews, or user testing sessions, focusing on the accuracy of spam classification, usefulness in reducing unwanted emails, and the overall user experience of the filter.

By utilizing these evaluation metrics and gathering user feedback, the performance of the spam filter can be assessed comprehensively, ensuring that it meets the requirements and addresses the needs of the end users.

**Programming Environments, Costs, and Resources:**

|  |  |  |
| --- | --- | --- |
| **RESOURCES** | **DESCRIPTIONS** | **COSTS** |
| Python programming language | Python is widely used in the machine learning community and offers a rich ecosystem of libraries and tools for natural language processing and sentiment analysis. | Open-sourced. No cost. |
| Machine learning libraries (scikit-learn and NLTK) | The project will use machine learning libraries such as scikit-learn and NLTK for implementing the Multinomial Naive Bayes algorithm and performing data preprocessing tasks. | Open-sourced. No Cost |
| Data Scientist 1 | Collaborate with Machine Learning Engineer 1 to:   * Collect and preprocess a large dataset of spam/non-spam emails. * Train and evaluate the spam filter SVM algorithm’s ability to classify spam/non-spam emails. * Assess the model’s accuracy, precision, recall, and F1 score.   Monitor the deployed model’s performance, track key metrics, and make updates as needed.  Maintain the spam filter by regularly updating the model and monitoring its performance. | $60hr x 80hrs = $4,800 |
| Machine Learning Engineer 1 | Work with Data Scientist 1 to:   * Gather, clean, and preprocess datasets to be used for training and evaluation. * Build and fine-tune the spam filter. * Assess the model’s accuracy, precision, recall, and F1 score.   Collaborate with Data Scientist 1 and Software Developer 1 to integrate the spam filter into the company’s current email infrastructure.  Document the design, implementation, and deployment process for reproducibility and knowledge sharing. | $60hr x 80 hrs = $4,800 |
| Software Developer 1 | Design and implement the core logic of the spam filter algorithm.  Collaborate with Data Scientist 1 and Machine Learning Engineer 1 to:   * integrate the spam filter into the company’s current email infrastructure. * ensure the filter’s code is well-documented and maintainable for future development.   Write and execute test cases to ensure the functionality and accuracy of the spam filter.  Handle user feedback and make necessary updates to improve the spam filter’s effectiveness. | $50hr x 70hrs = $3,500 |
| Project Manager | Develop a project plan outlining the goals, scope, and deliverables of the spam filter project.  Identify and allocate the necessary resources, including human resources, budget, and equipment, to support the project.  Facilitate collaboration among the team members, like data scientist 1, software developer 1, and machine learning engineer 1 to ensure effective communication/coordination.  Implement quality control measures to ensure the spam filter meets the desired performance and accuracy standards, conducting regular reviews and tests.  Maintain project documentation, including project plans, progress reports, and any necessary compliance documentation.  Facilitate effective communication channels among team members and stakeholders, ensuring everyone is aligned/informed about project progress, changes, and decisions. | $50hr x 60hrs = $3,000 |
| Laptop for Data Scientist | Laptop with a powerful processor and ample storage because data science work is memory-intensive, especially when working with large datasets.  It should also have good battery life since data science tasks can be demanding. The laptop should have a battery life of 8 hrs or more. | $3,000.00 |
| Laptop for Software Developer 1 | Laptop with a fast and modern processor for efficient code execution and running machine learning frameworks.  Must also have ample memory capacity for running intensive tasks and sufficient space to store the project files, datasets, and development tools. | $2,500.00 |
| Laptop for Machine learning engineer 1 | Laptop that meets the requirements for running machine learning frameworks and tools. It should have good processing power, memory, and storage capacity to support development tasks and experimentation. | $3,000.00 |
| Laptop for Project Manager 1 | Laptop with a powerful processor, ample storage, and capable of running project management software well. Battery life should be at least 6-8 hours for uninterrupted work. | $2,000.00 |
| Microsoft Suite License | To create/edit documentation. | $10/month x 3 licenses = $30 |
| Total |  | = $26,630 |

**Timeline and Milestones:**

**The following timeline offers a brief overview. More details for each are contained in the “Milestone Schedule” below.**

* June 20: Project approval.
* June 22: Start the project with a meeting with the team members.
* June 23: Work for the project starts.
* June 30th: Train the spam filter model.
* July 7th: Test the spam filter model and conduct the second team meeting for a status/progress update.
* July 12th: Product data solution review.
* July 19th: Final review of the entire project.
* July 21st: Final team meeting for a status update.
* July 25th: All deliverables and solutions are delivered to COF and stakeholders.
* July 30th: The spam filter is integrated into the company’s current email infrastructure.

**Milestones Schedule**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Milestone/Event** | **Start** | **End** | **Tasks** | **Prerequisites/Dependencies** | **Assigned Team Member** | **Total # of Hours** |
| Project Approved | June 20th | June 21st | 1. The project was approved by the COF and stakeholders. | Detailed written project proposal. | Data Scientist | 10hrs. |
| Project Started | June 22nd | June 22nd | 2. First team meeting to discuss project’s details and assign work to members. | Task: 1 | All team members. | 6hrs. |
| Project Work Starts | June 23rd | June 30th | 3. Collect the necessary data.  4. Clean/transform the data to make it suitable for model training/testing.  5. Write code to train the spam filter model.  6. Write code for the command-line interface. | Task: 2 | Task 3: Data Scientist and Machine Learning Engineer.  Task 4: Data Scientist.  Task 5-6: Software Developer. | 80 hrs. |
| Train the Spam Filter Model | June 30th | July 7th | 7. Train the spam filter using the collected/cleaned dataset.  8. Data documentation. | Tasks: 3, 4, 5, 6 | Task 7: Data Scientist and Machine Learning Engineer.  Task 8: Data Scientist and Machine Learning Engineer. | 60 hrs. |
| Test the Spam Filter Model. | July 7th | July 11th | 9. Test the spam filter model and ensure it’s functional.  10. Evaluate the model.  11. Second team meeting for a status update. | Task: 7 | Task 9: Data Scientist and Machine Learning Engineer.  Task 10: Data Scientist and Machine Learning  Task 11: All members. | 70 hrs. |
| Solution Review | July 12th | July 17th | 12. Code documentation.  13. Solution evaluation. | Tasks: 9, 10 | Task 12: Software Developer.  Task 13: Project Manager. | 45hrs. |
| Final Model Review. | July 19th | July 21st | 14. Documentation on all reviews.  15. Final team meeting.  16. Review the model and results against the requirements. | Tasks: 8, 12 | Tasks 14, 15: All members of the team.  Task 16: Project Manager. | 40 hrs. |
| Solution Delivered. | July 21st | July 25th | 17. Solution and all deliverables are delivered. | Tasks: 13, 14, 15 | Project Manager. | 0 hrs. |
| Spam Filter is integrated into current email infrastructure. | July 26th | July 30th | 18. The spam filter is implemented into the current email infrastructure. | Tasks: 12, 17 | Software Developer | 10hrs. |

**Part D: Post-implementation Report**

**Project Purpose:**

This project addresses the challenges of spam email filtering by introducing an objective and efficient approach. Existing methods rely on subjective rules, leading to risks like missed legitimate emails and exposure to threats. The project aims to develop a robust spam filter using advanced machine learning algorithms like SVM. It will enhance accuracy, protect against spam, and minimize the misclassification of legitimate emails while incorporating industry-appropriate security features. The project aims to provide Chewy and its employees with a reliable and efficient spam filter model that reduces unwanted emails. It prioritizes user-friendliness and integration with Chewy's email infrastructure. The spam filter is developed using Python and data science libraries, with continuous monitoring and optimization for improved accuracy and adaptability over time.

**Datasets:**

The project uses the "SMS Spam Collection Dataset" obtained from kaggle.com, which contains 5,574 English SMS messages classified as either legitimate (ham) or illegitimate (spam) (SMS Spam Collection Dataset, 2016). A new column named "spam" was added to the dataset to facilitate binary classification. This column assigns a value of 1 to rows labeled as 'spam' and 0 to rows labeled as 'ham', effectively distinguishing between spam and non-spam messages.

The dataset initially shows a significant class imbalance, with 87% of the messages labeled as ham and only 13% labeled as spam (SMS Spam Collection Dataset, 2016). To address this issue, the Synthetic Minority Oversampling Technique (SMOTE) was employed. SMOTE oversamples the minority class (spam messages) to balance the class distribution. The dataset was divided into training and testing sets. SMOTE was then applied exclusively to the training data. By oversampling the training data, the model can learn from a more balanced representation of spam and ham messages. SMOTE was only applied to the training data and not the testing data. This approach ensures that the testing data retains its original class distribution, allowing for an accurate evaluation of the performance of the spam filter model (Maddali, 2022). Additionally, the dataset underwent a privacy-focused step to remove any sensitive or personally identifiable information. This precautionary measure ensures compliance with legal and ethical considerations related to data privacy and protection.

The raw “SMS Spam Collection Dataset” is publicly available and can be found at the following link:<https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset>

**The Following Figures Provide Examples of the Dataset and the Code Used to Cleaned/Transformed it.**

**The dataset prior to cleanup/preprocessing:**

A screenshot of a computer

Description automatically generated with medium confidence

**The dataset after preprocessing:**

A screenshot of a chat

Description automatically generated with medium confidence

**Code used to clean up and preprocess the dataset:**

A screenshot of a computer program

Description automatically generated with medium confidence

**Product Code:**

The dataset, “SMS Spam Collection Dataset”, retrieved from kaggle.com contains 5,574 SMS messages classified as spam (illegitimate) or ham (legitimate) (SMS Spam Collection Dataset, 2016). A new column “spam” was added to the dataset, and it assigns a value of 1 to rows labeled as “spam” and 0 to rows labeled as “ham” to differentiate between spam and ham messages. As mentioned above, to address the significant class imbalance in the “SMS Spam Collection Dataset”, SMOTE was applied to oversample the minority class (spam messages). Addressing class imbalance is crucial for building a robust and effective model that can accurately classify both the majority and minority classes, leading to improved performance and better real-world applicability.

SMOTE was only applied to the training data and not the testing data. This approach ensures that the testing data retains its original class distribution, allowing for an accurate evaluation of the performance of the spam filter model (Maddali, 2022). Additionally, the dataset underwent a privacy-focused step to remove any sensitive or personally identifiable information. This precautionary measure ensures compliance with legal and ethical considerations related to data privacy and protection.

**Steps to clean and transform the text data in the “Message” column of the data frame:**

1. **Importing the necessary libraries:** The code imports the WordNetLemmatizer class from the nltk.stem module. The WordNetLemmatizer is used for lemmatizing words in the preprocessing step.
2. **Function definition:** The preprocess\_text () function was created to preprocess the text data.
3. **Stopwords and Lemmatization initialization:** Inside the preprocess\_text () function, it initializes a set of stopwords and a lemmatizer using the NLTK library. Stopwords are common words that don’t add much value to the meaning of a text and after usually removed during text preprocessing. The lemmatizer is used to reduce wods to their base or root form (Maddali, 2022).
4. **Cleaning and transformation:** The text is cleaned and transformed by removing non-alphabetical chars by using regular expressions (re.sub('[^A-Za-z]', ' ', text.lower())),  and converting the text to lowercase. This helps in standardizing the text data (Maddali, 2022).
5. **Tokenize:** The text is tokenized into individual works using the word\_tokenize function from NLTK. Tokenization breaks the text into smaller units (words) for further processing (Maddali, 2022).
6. **Stopwords removal and Lemmatization:** stopwords are removed from the tokens using a list comprehension (tokens = [lemmatizer.lemmatize(token) for token in tokens if token not in set\_of\_stopwords]). Lemmatization is applied to the remaining tokens using the lemmatizer initialized earlier. This helps in reducing words to their base or root form, which can improve the accuracy of the text analysis (Maddali, 2022).
7. **Joining tokens:** The preprocessed tokens are joined back into a single string using ' ‘. join(tokens). This creates a preprocessed version of the original text (Maddali, 2022).
8. **Applying preprocessing function:** Finally, the preprocess\_text function is applied to the 'Message' column of the spam\_ham\_df DataFrame using the apply function. This applies the text preprocessing steps to each message in the column and updates the column with the preprocessed text (Maddali, 2022).

By applying this preprocessing function to the 'Message' column, the text data is cleaned, standardized, and transformed in a way that removes stop words and performs lemmatization, making it more suitable for model training.

**Hypothesis Verification:**

**The hypotheses for this project were:**

1. The SVM algorithm will accurately classify emails as either spam or non-spam.
2. The implementation of the spam filter will lead to a substantial decrease in the number of spam emails received by our employees.
3. The spam filter will have a negligible false positive rate, ensuring that legitimate emails are not mistakenly marked as spam.

**The results of the spam filter using the SVM algorithm are as follows:**

A screenshot of a computer program

Description automatically generated with medium confidence

A picture containing text, receipt, screenshot, font

Description automatically generated

Based on the results, it shows that the hypotheses I formulated have been supported by the performance of my spam filter using the SVM algorithm.

**Hypothesis 1:** The SVM algorithm will accurately classify emails as either spam or non-spam.

The high accuracy, precision, recall, and F1-score values indicate that the SVM algorithm is indeed effective in accurately classifying emails as spam or non-spam. The model achieved an accuracy of 97.94%, indicating a high level of correctness in its predictions.

**Hypothesis 2:** The implementation of the spam filter will lead to a substantial decrease in the number of spam emails received by your employees. While the provided results do not directly measure the decrease in the number of spam emails received, the high precision and recall values for the spam class suggest that the spam filter is effective in correctly identifying and filtering out spam emails. This indicates that the implementation of the spam filter is likely to have led to a substantial decrease in the volume of spam emails reaching your employees' inboxes.

**Hypothesis 3:** The spam filter will have a negligible false positive rate, ensuring that legitimate emails are not mistakenly marked as spam. The precision value of 1.00 or 100% for the spam class means that all instances predicted as spam (class 1) were correctly classified as such, with no false positives. This means that the filter will not mistakenly mark legitimate emails as spam, ensuring that important emails are not incorrectly classified and filtered out.

**Effective Visualization and Reporting:**

Visualization and reporting were crucial parts of development and provided feedback on how the model was performing and whether adjustments need to be made.

**First Visual:**

A close-up of words

Description automatically generated with low confidence

* The first visual is a word cloud that shows the most frequently occurring words in spam messages. It provides a quick and intuitive way to understand the prominent terms in spam text. A word cloud can help us identify common patterns, keywords, or phrases that are often associated with spam. It can give us insights into the language or content used in spam messages and help us understand the characteristics of such messages (Shah, 2022).
* By visualizing the most frequent words in spam messages through a word cloud, we can easily spot the dominant terms that appear larger and more prominently in the cloud. This can help us focus the analysis and potentially identify specific words or phrases that are strong indicators of spam.

**Second Visual:**

A picture containing text, font, graphics, graphic design

Description automatically generated

* The second visual is a word cloud that shows the most commonly occurring words in ham messages. It provides a quick and intuitive way to understand the prominent terms in ham text. A word cloud can help us identify common patterns, keywords, or phrases that are often associated with ham (legitimate emails). It can give us insights into the language or content used in ham messages and help us understand the characteristics of such messages (Shah, 2022).
* By visualizing the most frequent words in ham messages through a word cloud, we can easily spot the dominant terms that appear larger and more prominently in the cloud. This can help us focus the analysis and potentially identify specific words or phrases that are strong indicators of ham (legitimate emails).

**Third Visual:**

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

* The third visual is a bar plot that shows the class distribution of the dataset before and after applying SMOTE for resampling the training data (GeeksforGeeks, 2021). SMOTE was used to oversample the minority (spam) class in the dataset because of the significant class imbalance. The dataset initially contained 87% ham emails and only 13% spam.
* The bar plot provides a clear and useful visualization of the difference in class distribution before and after applying SMOTE (GeeksforGeeks, 2021). Before applying SMOTE, the blue bar had a huge difference between them. But after applying SMOTE, the orange bars were the same and thus, class distribution became balanced.

**Fourth Visual:**

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

* The fourth figure is a confusion matrix, which provides a visual representation of the spam filter's performance metrics on the test data. It shows the counts of true positives, true negatives, false positives, and false negatives (Indeed Editorial Team, 2022).

**The confusion matrix of the test data provided above represents the following:**

* True Positives (TP): 126
* True Negatives (TN): 966
* False Positives (FP): 0
* False Negatives (FN): 23

**From the confusion matrix, the spam filter’s performance on the test data is evaluated using various metrics:**

**Let's calculate each metric:**

1. Accuracy = (TP + TN) / (TP + TN + FP + FN) = (126 + 966) / (126 + 966 + 0 + 23) ≈ 0.9794, or approximately 97.94%.
2. Precision = TP / (TP + FP) = 126 / (126 + 0) ≈ 1.00, or approximately 100%.
3. Recall = TP / (TP + FN) = 126 / (126 + 23) ≈ 0.8456, or approximately 84.56%.
4. F1-score = 2 \* (Precision \* Recall) / (Precision + Recall) = 2 \* (1.00 \* 0.8456) / (1.00 + 0.8456) ≈ 0.9174, or approximately 91.74%.

**Therefore, the calculated metrics are:**

* Accuracy: approximately 97.94%
* Precision: approximately 100%
* Recall: approximately 84.56%
* F1-score: approximately 91.74%.

These metrics provide a comprehensive evaluation of the spam filter's performance, including its accuracy, precision, recall, and overall effectiveness in classifying spam and non-spam emails.

**Fifth Visual:**

A picture containing text, screenshot, diagram, plot

Description automatically generated

* The fifth figure is a bar plot to visualize the performance metrics of the spam filter model. The resulting plot shows three bars for each class (Ham and Spam) representing the precision, recall, and F1-score metrics. Each bar has a different color (blue for precision, pink for recall, teal for F1-score) and is labeled accordingly. The x-axis represents the classes (Ham and Spam), the y-axis represents the score, and the legend indicates the metrics (GeeksforGeeks, 2021).
* The purpose of this bar plot is to visually compare and analyze the performance metrics of the spam filter model for the spam and ham classes.

**Accuracy Analysis:**

The metrics of the spam filter’s performance on the test data show that the model’s accuracy is 0.9794, which demonstrates that the spam filter can correctly classify emails as spam or non-spam with a high level of accuracy. An accuracy of 0.9794 means that out of the total of around 1,114 never seen before emails that were set aside for the test set, the model correctly classified approximately 97.94% of them. This is a high accuracy rate, suggesting that the spam filter is effective in distinguishing between spam and non-spam emails (Indeed Editorial Team, 2022).

The precision of approximately 100% suggests that when the spam filter classifies an email as spam, it is highly likely to be correct. A high precision indicates a low rate of false positives, meaning that there are very few instances where legitimate emails are incorrectly classified as spam. The recall of approximately 84.56% indicates that the spam filter successfully identified 84.56% of the actual spam emails in the dataset. Recall, also known as sensitivity or true positive rate, measures the ability of the model to correctly identify positive instances (spam) from the total actual positive instances. The F1-score of approximately 91.74% is a measure that combines both precision and recall. It provides a balance between these two metrics and is useful when you want to consider both false positives and false negatives. A higher F1-score indicates a better overall performance of the spam filter (Indeed Editorial Team, 2022). Overall, the spam filter performs well with high accuracy and precision.

**Application Testing:**

The spam filter underwent extensive testing to ensure its functionality. This involved both running the program and manually testing each feature. Whenever an error was encountered, it was carefully examined, identified, and promptly resolved. Throughout the development process, the model's functionality was continuously tested to ensure that it performed as expected. This testing occurred regularly as the features were being created. Furthermore, a comprehensive final testing phase was conducted at the end of the project to ensure that all intended functions were present and available.

To evaluate the filter's effectiveness, various examples of spam and ham messages/emails were inputted through the command-line interface. This rigorous testing aimed to confirm the filter's ability to correctly classify spam and ham emails. The results of these tests demonstrated that the spam filter performed very well.

**Application Files:**

1. C964: Capstone-Final-Ver. ipynb: this file contains the entire spam filter code. The project was created using Google Codelabs as the development environment, so this file has all the code and data for the spam filter model.
2. spam.csv: CSV file retrieved from kaggle.com containing spam and non-spam (ham) emails. It was uploaded to my Google Drive and then downloaded into the notebook because Google Codelabs doesn’t provide permanent storage for files uploaded by users because it’s primarily designed as an interactive platform for creating and sharing coding tutorials or workshops.

**User’s Guide:**

**Please use the following steps to access and run the application:**

1. Access the Jupyter Notebook through the Binder environment by clicking on the link below and waiting for the environment to load. When you try to access the application the first time, Binder may take some time to load it, so please be patient.

* <https://mybinder.org/v2/gh/sophiedang0101/WGU-Capstone-Project/HEAD>

1. Once the Jupyter Notebook is loaded, clear the “Kernel” of all potential outputs.

* To do this, go to the window’s top left corner🡪 click on the “Kernel” menu 🡪 click “Restart Kernel”.

A screenshot of a computer

Description automatically generated with medium confidence

1. After the Jupyter Notebook is loaded and the Kernel cells have been reset, click on the file named “C964-Capstone-Project.ipynb” to launch it, then select the first cell (packages imports) and run it by using the commands (SHIFT+ENTER).

* You can also run a cell by clicking on the “Run” button. It looks like an arrow.

A screen shot of a computer

Description automatically generated with low confidence

* NOTE: [\*] 🡪 This means that the cell is currently running, wait for it to finish before running the next cell to avoid bottlenecking the kernel.

1. After running the first cell, you will automatically select the next cell, therefore, at this point, just continue to run each cell by using the command (SHIFT+ENTER) and observe the outputs. Please be careful not to run a selected cell twice.
2. Once you get to the cell with the function call, check\_user\_input (), you will have the ability to try the spam filter. First, run the cell with the function call, check\_user\_input (), then you will be prompted to enter an email/message and have the spam filter classify it as spam or ham. If you want to quite the program, type “exit”.

A picture containing text, screenshot, font, white

Description automatically generated

**Summary of Learning Experience:**

I had very little machine-learning experience before creating the spam filter, so this project provided me with an opportunity to venture into the field of machine learning. Initially, I expected it to be a complex undertaking, but it turned out to be more manageable than anticipated. Although my programming background is primarily academic rather than professional, my previous academic experience equipped me with valuable skills in problem-solving and breaking down complex tasks into smaller, manageable components. This knowledge enabled me to approach the project systematically, designing, implementing, and testing the software in a structured manner.

To complete this project, I had to acquire a substantial amount of knowledge about machine learning and its implementation. I devoted considerable time to studying guides, explanations, and video tutorials that elucidated the intricacies of spam classification and related topics. This learning experience, though not the most demanding project I have undertaken, proved to be one of the most knowledge intensive. I gained a comprehensive understanding of the Support Vector Machine algorithm and the intricacies of data processing. Engaging in this project has reaffirmed my commitment to lifelong learning and bolstered my confidence in my ability to acquire new skills and overcome challenges. I now possess a greater sense of assurance in my capacity to achieve my goals as I continue to expand my knowledge and gain further experience.

**Works Cited**

*Data Protection and Privacy: How to Protect User Data*. (2023, June 2). Cloudian. <https://cloudian.com/guides/data-protection/data-protection-and-privacy-7-ways-to-protect-user-data/>

GeeksforGeeks. (2023). Support Vector Machine SVM Algorithm. GeeksforGeeks. <https://www.geeksforgeeks.org/support-vector-machine-algorithm/>

GeeksforGeeks. (2021). Bar Plot in Matplotlib. GeeksforGeeks. <https://www.geeksforgeeks.org/bar-plot-in-matplotlib/>

Indeed Editorial Team. (2022). What Is a Confusion Matrix? (Plus How To Calculate One). Indeed.com. <https://www.indeed.com/career-advice/career-development/confusion-matrix>

Maddali, S. (2022, December 20). What Are the Most Important Preprocessing Steps in Machine Learning and Data Science? Medium. <https://towardsdatascience.com/what-are-the-most-important-preprocessing-steps-in-machine-learning-and-data-science-a7606d18f32a>

SMS Spam Collection Dataset. (2016, December 2). Kaggle. <https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset>

Shah, R. (2022). How to Build Word Cloud in Python? Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2021/05/how-to-build-word-cloud-in-python/>

Pr, N. (2023). What Is Scrum Methodology? & Scrum Project Management. Nimblework. <https://www.nimblework.com/agile/scrum-methodology/>