College of Engineering and Computing
DATA ANALYTICS ENGINEERING
George Mason University

Fall 2024

# Proactive Identification of Product Safety Issues

DAEN 690

# Project Report

Aakash Boenal
Jonathan Stewart King
Saiphani Chandra Vuppala
Utkarsh Ganjihal

# About the Cover

James Baldo is an associate professor and serves as the Director of the George Mason University (GMU), College of Engineering and Computing (CEC), Volgenau School of Engineering (VSE), MS Data Analytics Engineering (DAEN) program. Dr. Baldo has served in this position since Fall 2018 and recently returned this past May after a 2-year sabbatical applying state-of-the-art data analytics engineering concepts and technologies to solve real world problems.

Prior to becoming director of the DAEN program he served 19 years as a CEC adjunct professor while working as a practicing engineer. His career has over 45 years of industry and government experience with roles as a data analytics engineer and software engineer.

His interest in large scale data, data management, analytics, and tools has provided him with opportunities to engage in assessing and applying new technologies across a diverse range of problems. The adoption of new technologies is exciting; however, adoption of new technologies requires careful planning and in addition to the technology, taking special care for planning how to successfully address for both organizational and cultural factors of the enterprise. This requires not only engineering knowledge and skills, but the ability to work on teams that interact across different corporate stakeholders.

Dr. Baldo continues to consult part-time with industry and leverages this knowledge and experience as feedback to the DAEN program. With technology moving at a lightening pace and technology adoption rates increasing, Dr. Baldo closely monitors the skillset needs of industry and how the DAEN program can provide graduates to fulfill these needs.

Dr. Baldo is currently researching applications of data mesh and data fabric for data management and exploring how analytics will be integrated into agentic systems.

# Contents

## Table of Contents

# Table of Figures

## Table of Tables

# Abstract


## Abstract

Within the retail industry, early detection of products not meeting safety standards is an enduring pursuit. All too often, an unsafe product has made its way far into the market before the flaw is discovered, and a Consumer Product Safety Commission (CPSC) safety recall is forced to reach deep throughout the market. This study is taking on the goal of utilizing crowdsourcing methods through Amazon review data to attempt early prediction of products not meeting standards.

Centrally, is there a machine-readable comparison between CPSC safety recalls and Amazon reviews/ratings? Amazon review data is a multitude of data types, but predominantly free-text format, thus this study will focus much on Natural Language Processing (NLP) techniques against the reviews themselves. This is analogous to research conducted in the food safety industry. [31] We will expand on the training of a domain-specific model for product safety, but examining a holistic approach of a combined sentiment and a “dirty word” search. In addition, this study utilized not just product recalls, but focused on product incidents from the CPSC dataset to provide additional incident context as well.

To do so, starting on baby products to build a pipeline, we merged all datasets on the Amazon Seller Identification Number (ASIN), model ID, and Universal Product Codes (UPC) to identify specific products that were involved in an incident and/or recalled, comparing those and the balances of sentiment, and positive/negative reviews. In addition, we utilized SpaCy’s WordNet with an initial list of 523 “Safety Words” that would provide the specific effects verbiage (“fire hazard”, “choking”, “chemical”, etc) from incidents that, when fed against Amazon reviews, would trigger the classification “Potentially unsafe product.” We attempted a Logistic Regression, XGBoost, Decision Tree, and a RandomForest classification model against the merged datasets, using Baby Products as the example dataset.

Our findings are that, due to the disparity and non-standardized CPSC data, data merging was problematic and creating a very poorly balanced dataset. A supervised classification model of labeled data returned poor results. Logistic regression and Randomforest classification models returned the best results at a 14% recall against the test dataset (XGBoost producing a 0% recall), which reinforces a need for alternative analysis.

Alternative analysis included a heuristic “rules-based” approach, a filtered dataset based on quantile thresholds: Density of Safety Words, average review, average sentiment score, and a match rate of review text matching Safety Words against matching incidents. This more simplistic approach proved to produce more viable results. When fed against all 34 Amazon categories (450million rows in total), produced an average of 23 flagged products per category.

# Report

## Section 1: Introduction

### 1.1 Report Purpose

With the ever-deepening pools of online consumer review data, this project examines the potential to enhance product safety standards through user-based reviews. Specifically, it addresses the challenge of predicting Consumer Product Safety Commission (CPSC) product safety recalls using an extensive database of Amazon ratings/reviews. It aims to leverage Natural Language Processing (NLP), sentiment analysis, time series, patterns and correlations between customer product reviews and subsequent safety recalls. It will detail methodologies, data processing, model development techniques and tuning, and effectiveness evaluation of the predictive analysis. These findings will be provided to enhance consumer safety standards for early detection of potential product hazards. [1]

### 1.2 Report Readership

This report is addressed to data scientists, machine learning engineers, and professionals in consumer safety/regulatory agencies, particularly those affiliated with the CPSC. With a viable algorithm, product manufacturers, quality assurance teams, and data-driven decision-makers within e-commerce platforms like Amazon, Walmart, Shopify, etc, will find value in this report as well. The report is geared towards researchers in the field of data analytics, NLP and public safety, but with data-driven decision-making being increasingly used across the full manufacturing cycle, a means of early warning for safety recalls would be of interest to the full sector. [1]

### 1.3 Report Structure

This report is broken down within eight major sections. After the introductory first section, section two defines the problem set. This provides the definition of the problem, problem and solution spaces, and a vision of the solution. Section three describes the dataset and dataset processing. This includes descriptors, contextualism, quality assessments, and processing/storage requirements. Section four is the explanation of the analysis including algorithm design, classification analysis and solution approaches. Section five is the interpretation of the results of the analysis to include visualizations, testing, and validation of the results. Section six, seven and eight is an explanation of our findings, a summarization, and details of future works.

# Section 2: Problem Definition

## 2.1 Problem Space

This study is focused within the retail sector. Workflow was specifically focusing on baby products and product safety, but the programmatic pipeline was designed for every sector. The current system for identifying safety risks in products is often reactive, catching issues only after the product has already entered the hands of consumers. This reactive approach results in delayed recalls, putting consumers at risk and exposing businesses to higher costs. [1][2]

Unsafe products can pose serious risks, particularly for vulnerable consumers such as infants, and can result in costly recalls, injuries, or even fatalities. These risks not only affect consumers but also lead to reputational damage for companies involved. [2][3]

A proactive approach could utilize historical recall data and advanced analytics to identify product risks early in the distribution cycle, thus reducing the number of defective products reaching the market to ensure safety of consumers and minimize manufacturer impacts. [2][3][4]


Figure 1: Common Baby Products Subject to Potential Safety Risks [32]

## 2.2 Research

### 2.2.1. Data Research

Our project focuses on identifying product safety issues proactively by analyzing unstructured data sources, including government recall databases, customer reviews, and incident reports. Our research aimed to find a way to help manufacturers and safety authorities act quickly before a product defect becomes a serious hazard. [1]

We began by researching public datasets from SaferProducts.gov and other governmental recall databases. These datasets provided rich information about product recalls across various industries, highlighting recurring safety issues that led to recalls. Among the most common problems were manufacturing defects, faulty designs, and hazardous materials that could pose risks to consumers, especially vulnerable populations like children and the elderly. This helped us understand the common safety concerns we should be looking for in our analysis. [2][3][4]

For example, products such as household appliances and children's toys were frequently recalled due to risks of fire, choking hazards, and electrical malfunctions. This helped us identify key patterns in the types of incidents that led to recalls. This domain-specific research was essential in guiding our analysis, as it provided a foundation for understanding which incidents and consumer complaints are likely to result in recalls. [5][6][7][8][9][10]

In addition to recall data, customer reviews from platforms like Amazon gave us a wealth of unstructured data that offered insights into real-world product usage and potential safety risks. Our focus was on extracting insights from low-star reviews, typically rated 1- or 2-stars. These reviews often contained early warning signs of safety issues, such as overheating appliances, malfunctioning parts, or poor product durability, that had not yet escalated to an official recall. [1]

As part of our research, we also faced challenges in merging different datasets, particularly because product model numbers were inconsistently reported across various sources. For example, a product's model number in a recall report might not exactly match how it was listed in a customer review, even though they referred to the same product. This led us to explore techniques like Named Entity Recognition (NER) and fuzzy matching which helped us align the product data from various sources. The domain research helped us recognize that manual efforts to clean and align the data were crucial to ensuring the accuracy of our analysis. With these challenges in mind, we developed a workflow that leveraged Natural Language Processing (NLP) techniques to streamline and enhance our product safety identification process. [1][18][33][36][39]

### 2.2.2. Literature Review on Product Safety

Our approach to leveraging NLP and machine learning to identify product safety risks was further guided by reviewing relevant literature:

#### 2.2.2.1. Auto-Detection of Safety Issues in Baby Products (2018):

This study focused on using machine learning and NLP to detect safety issues in baby products based on online reviews. The research compiled data from Amazon reviews, SaferProducts.gov complaints, and recall descriptions from the U.S. Consumer Product Safety Commission (CPSC). Various classifiers were tested, including Logistic Regression, SVMs, Naive Bayes, and Random Forests, with Logistic Regression achieving 66% precision in the top 50 reviews. The study emphasized the gap between consumer-reported safety issues and formal recalls, underscoring the need for automated solutions to detect potential risks early. [5]

#### 2.2.2.2. What's Wrong with this Product? Detection of Product Safety Issues (2023):

This research examined product safety issues in online consumer feedback, particularly in e-commerce. Using a combination of data from the European Union Safety Gate (EUSG) and online product reviews,

Logistic Regression emerged as the best-performing model for identifying hazardous products. This study's framework reinforced the importance of using large volumes of online data to provide early warnings of potential hazards across product categories, offering valuable insights for consumer safety efforts. [6]

### 2.2.3. NLP Techniques Employed based on Research

Given the complexity of working with unstructured text from reviews and incident reports, we made use of various NLP techniques.

#### 2.2.3.1. Tokenization and Stop Word Removal

By breaking down reviews and incident reports into individual words and removing common, non-informative words (e.g., "the," "and"), we were able to focus on key terms that are more indicative of safety concerns. [34][35]

#### 2.2.3.2. Named Entity Recognition (NER)

NER was particularly useful for extracting product names, model numbers, and safety-related terms from unstructured text. We researched NER models capable of identifying critical product features and linking them to specific incidents or reviews. [36]

#### 2.2.3.3. TF-IDF and Bigram/Trigram Analysis

We used TF-IDF (Term Frequency-Inverse Document Frequency) to quantify the importance of specific terms within the dataset. Additionally, we applied bigram and trigram models to capture semantic relationships between words, improving the accuracy of keyword identification and enabling better context analysis of safety-related terms. [37]

#### 2.2.3.4. Sentiment Analysis

Sentiment analysis allowed us to determine the tone of customer reviews. Negative reviews, particularly those that mention safety terms (e.g., "fire hazard," "dangerous," "broken"), were flagged for further analysis. The Twitter RoBERTa model helped classify reviews as either positive or negative based on their sentiment, allowing the team to identify patterns in customer dissatisfaction that could point to potential hazards. [19][38]

#### 2.2.3.5. Fuzzy String Matching

Due to the inconsistency of product names and model numbers across datasets, we applied fuzzy string matching (using the FuzzyWuzzy library). This method allowed us to identify and match similar product names or model numbers despite slight variations in how they were presented in different sources (e.g., Amazon vs. CPSC). For instance, if a product model number in the review slightly differed from the one in the recall database (due to a typo or variation), fuzzy matching enabled us to link the two. We set an 85% similarity threshold for fuzzy matching, which ensured that most discrepancies were accounted for without introducing too many false positives. [39]

#### 2.2.3.6. Bag of Words and Synonym Analysis

The Bag of Words model helped us extract key terms from the incident reports and reviews, focusing on frequently occurring words related to product defects and safety concerns. Synonym analysis, using tools like WordNet, further enhanced this by matching related terms (e.g., "fire" and "flame"), allowing for a more comprehensive analysis of recurring safety issues. [40][41]

### 2.2.4. Model Exploration and Implementation

As part of our team's ongoing implementation, we adopted the Random Forest Classification model to conduct further analysis and prediction tasks. This involved hyperparameter tuning and breaking down the dataset into bigrams and trigrams to improve classification accuracy. We applied binary classification methods to identify potential safety issues based on the data we processed.

Our use of descriptive statistics helped us define baseline measures for evaluation, and we explored models like Logistic Regression, Random Forest, and more advanced techniques like XGBoost and deep learning models (e.g., LSTM, CNN). The choice of models was influenced by the structure of the data, with a focus on precision, recall, F1-score, and ROC-AUC as key performance metrics. Additionally, we developed strategies to handle class imbalances in the dataset to ensure better predictive performance.

By integrating these NLP techniques and machine learning models, we aimed to proactively identify product safety issues from the available unstructured data sources. This approach enabled our team to streamline data processing and extract meaningful insights critical for improving product safety. [42][43][44][45][46]

## 2.3 Solution Space

This study emphasizes a crowd-sourcing approach. Rather than relying on singular expert examinations of products, it examines a broad stroke approach-based from customers that purchased it. Utilizing Amazon consumer reviews, this study focuses on the verbiage associated within the free-text data from the customers. If an unsafe product has entered the market, the odds are that customers would post reviews warning other customers of the hazards, utilizing specific verbiage such has "too many small parts" or "chemical rash on my baby." At the same time, they would be reviewing the product negatively. With these data points in mind, a machine learning model can be trained to identify these points within certain thresholds to classify potentially unsafe products.

In terms of model exploration, we began by considering Logistic Regression as our baseline model due to its simplicity and interpretability. This allowed us to establish an initial benchmark for performance. From there, we explored more sophisticated models such as Random Forest, which handles both structured (e.g., product categories, star ratings) and unstructured data (e.g., review text) effectively. Random Forest also captures feature interactions and non-linear relationships, which makes it suitable for identifying complex patterns in product reviews.

For more advanced analysis, we considered applying Gradient Boosting models (such as XGBoost and CatBoost), particularly because research suggests these models often outperform Random Forest in terms of predictive accuracy. Given our dataset's imbalance—where unsafe products are far less frequent—we found that these models handle imbalanced data well by adjusting for class weights.

Our plan also involves using binary classification to identify whether a product is "potentially unsafe" or "not unsafe," with metrics like precision, recall, and F1-score guiding our model evaluation. While more complex models like LSTM or CNN could potentially offer deeper insights into text data, we are focused primarily on the application of Random Forest and XGBoost to strike a balance between performance and interpretability at this stage. [42][43][44][45][46]

## 2.4 Project Objectives

### 2.4.1. Learning Outcomes

Our project has been an incredible journey, allowing us to dive deep into the integration of large-scale datasets from Amazon product reviews and CPSC recall and incident data. Through this work, we've gained valuable insights into identifying potentially unsafe products by combining diverse sources of information. The process has given us hands-on experience with advanced data preprocessing techniques, enabling us to cleanse, merge, and enrich datasets to extract meaningful and actionable insights.

A key highlight has been applying sentiment analysis and text mining to identify safety-related terms like "burned" or "choked." These efforts have allowed us to uncover potential hazards, bridging the gap between raw data and real-world implications. Additionally, we've focused on designing and fine-tuning machine learning algorithms to predict product safety concerns effectively. By evaluating models and calibrating thresholds, we've developed tools that can proactively assess risks, significantly contributing to consumer safety.

This project has expanded our understanding of practical machine learning and other aggregation applications in consumer protection. We've also sharpened our expertise in heuristic and threshold-based classification methods, making us interdisciplinary experts in e-commerce, product safety, and data science. Ultimately, our goal has been to create a tool with proactive detection capabilities to ensure safer consumer products and drive improved industry standards. It's been an exciting and rewarding experience to work on something that has the potential to make a meaningful impact on consumer safety worldwide.

### 2.4.2. Solution outcomes

By the end of this project, we aim to deliver a powerful and functional command-line interface tool designed to proactively identify potentially hazardous products. This tool will analyze Amazon product reviews and CPSC recall and incident data, leveraging real-time feeds and preprocessed datasets. Using sentiment analysis and natural language processing, it will extract meaningful safety-related insights, flagging products that frequently include terms like "burned" or "choked." These flagged terms will then be cross-referenced with known incident and recall data to pinpoint at-risk products.

Our solution is designed to go beyond detection, offering scenario-driven outputs. For instance, it can generate alerts for manufacturers when warning signs surpass predefined thresholds and deliver actionable insights to guide product improvements and ensure compliance with safety standards. This tool is envisioned as a robust, scalable solution for organizations like the Consumer Product Safety Commission (CPSC) or manufacturers, empowering them to detect and address product safety issues proactively and effectively, ultimately contributing to safer products and enhanced consumer protection.

### 2.4.3. Understanding of the problem space

By the completion of this project, we expect to have gained a deep understanding of the complexities involved in identifying potentially unsafe products from large-scale textual datasets. Through our work, we aim to uncover deeper connections between consumer reviews, product recall data, and safety incidents. This includes understanding how patterns in language, sentiment, and specific hazard-related terms can reveal underlying risks associated with products.

We anticipate this project will enhance our ability to merge and analyze diverse data sources—such as Amazon reviews and CPSC reports—to generate actionable insights. Additionally, it will provide us with valuable experience in handling unstructured text data, addressing data quality challenges, and applying machine learning

techniques to predict risk. These learnings will not only refine our technical expertise but also equip us with a stronger perspective on tackling similar real-world problems in the future that require integrating and analyzing large, disparate datasets to derive meaningful solutions.

### 2.4.4. Value provided

The research team believes this project will have a significant impact by delivering substantial benefits to consumers, manufacturers, and regulatory agencies through the proactive detection of unsafe products. For consumers, the tool offers enhanced safety by identifying products that exhibit risk patterns from reviews and incident data, potentially preventing injuries or harm before they occur.

Manufacturers will benefit from robust market research and quality control tool. By providing actionable insights into consumer feedback, the tool highlights areas that require product improvement, allowing manufacturers to address safety concerns proactively. This ensures compliance with safety standards, mitigates risks, and fosters greater consumer trust.

For regulatory agencies such as the Consumer Product Safety Commission (CPSC), the tool provides a powerful platform to monitor and analyze safety-related trends in large datasets. This enables faster, more accurate identification of products that may require recalls or safety notifications, enhancing their ability to protect the public.

In essence, this project not only ensures safer consumer products but also strengthens trust within the e-commerce ecosystem. It represents a significant step forward in leveraging data analytics and machine learning to advance public safety and improve industry standards.

## 2.5 Primary User Stories

Based on the user context and value proposition, we developed the following primary user story to guide our project:

### 2.5.1. User Story 1: Data Cleaning

**Description:** As a Developer, I want to clean the dataset by handling null values, removing stop words, processing bigrams/trigrams, and designing a schema for model creation and feature extraction, so that we can ensure the dataset is pre-processed correctly and is ready for accurate analysis, specifically focusing on Baby Products.

**Subtasks:**

i. Amazon Dataset: Tokenization
ii. Amazon Dataset: Stop word removal
iii. Amazon Dataset: Named Entity Recognition
iv. Amazon Dataset: TF-IDF Vectorization
v. Same Process for Recall Dataset

**Exit Criteria:** Dataset is cleaned, schema is established and there is no missing or inaccurate data.

### 2.5.2. User Story 2: Data Exploration (Part 1)

**Description:** As a Developer, I want to perform exploratory data analysis to understand the structure, distribution, and key characteristics of the datasets, so that I can gain insights and identify any potential issues or patterns for further analysis.

**Subtasks:**

i. Load and Inspect Datasets
ii. Generate Summary Statistics
iii. Visualize Data Distributions
iv. Identify Data Gaps and Anomalies

**Exit Criteria:** EDA is completed, and an internal report detailing key insights and potential.

### 2.5.3. User Story 3: Data Exploration (Part 2)

**Description:** As a developer, I want to perform data exploration by generating descriptive statistics and identifying the most common bigrams and trigrams in the Recall Dataset so that I can focus on key phrases like "fire hazard" or "choking hazard" and better understand safety issues related to products.

**Subtasks:**

i. Review the dataset for bigrams and trigrams.
ii. Perform exploratory data analysis.
iii. Identify key verbiage related to product safety issues.
iv. Generate statistical insights for key phrases.

**Exit Criteria:** Pattern recognition of recall/amazon dataset is established, and a method of sentiment analysis or key verbiage search is established for prediction model.

### 2.5.4. User Story 4: Domain Research

**Description:** As a team member, I want to conduct research on product safety, particularly in baby products, so that I can understand common safety issues and recall triggers to better inform the analysis and identification of safety risks.

**Subtasks:**

i. Review Literature on Product Safety
ii. Analyze Historical Recall Data
iii. Identify Common Issues in Baby Products
iv. Summarize Findings in a Report

**Exit Criteria:** A research document summarizing key safety issues and recall triggers in baby products is created, with related works and an introduction to the overall project.

### 2.5.5. User Story 5: Learn NLP

**Description:** As a team member, I want to learn and understand Natural Language Processing (NLP) techniques so that I can apply them effectively in tasks such as text processing, tokenization, and feature extraction for our datasets.

**Subtasks:**

i. Review fundamental NLP techniques (e.g., tokenization, stop word removal).
ii. Study advanced NLP models and applications (e.g., Named Entity Recognition, TF-IDF, sentiment analysis).
iii. Implement basic NLP techniques on a sample dataset.
iv. Document NLP learning and findings for the team.

**Exit Criteria:** The team has a solid understanding of NLP demonstrated by implementing basic techniques and documenting the results for application in future phases.

### 2.5.6. User Story 6: Manual Data Cleaning of Incidents Dataset

**Description:** As a developer, I want to manually clean the Incidents dataset to ensure accurate data for model development. This involves identifying and resolving issues with messy and repeated data in the dataset.

**Subtasks:**

i. Review and identify messy data in the Incidents dataset.
ii. Clean and restructure the dataset.
iii. Verify dataset quality after cleaning.

**Exit Criteria:** The Incidents dataset is clean and consistent with all issues resolved and ready for further processing.

### 2.5.7. User Story 7: Manual Model Extraction of Recall Dataset

**Description:** As a developer, I want to manually extract model numbers from the recall dataset to ensure accurate merging with other datasets. This includes identifying relevant model numbers and formatting them for easy integration into the project.

**Subtasks:**

i. Extract relevant model numbers from the recall dataset.
ii. Format model numbers for integration.
iii. Verify accuracy and consistency of the extracted data.

**Exit Criteria:** Model numbers are successfully extracted, formatted, and ready for merging with other datasets.

### 2.5.8. User Story 8: Sentiment Analysis of Amazon Reviews Raw Data

**Description:** As a developer, I want to perform sentiment analysis on the raw textual data of the Amazon review dataset so that I can perform a binary classification of positive/negative reviews.

**Subtasks:**

i. Preprocess Amazon review dataset.
ii. Apply sentiment analysis techniques (e.g., RoBERTa model).
iii. Classify reviews as positive or negative.
iv. Document insights from sentiment analysis.

**Exit Criteria:** EDA is completed, and an internal report detailing key insights and potential.

### 2.5.9. User Story 9: Sentiment Analysis of Incident/Recall Data

**Description:** As a developer, I want to perform sentiment analysis on incident descriptions and recall summary datasets and compare them with the Amazon reviews dataset to identify patterns.

**Subtasks:**

i. Preprocess incident and recall summary datasets.
ii. Apply sentiment analysis techniques.
iii. Compare sentiment results with Amazon reviews.
iv. Document insights and findings.

**Exit Criteria:** Process is complete, and an internal report detailing key insights and potential.

### 2.5.10.User Story 10: Research Free-Text Model ID Extraction

**Description:** As the Product Owner, I want to research a methodology to extract model IDs from the summary reports of the recall dataset to establish a pipeline for identifying specific recalled products and their incidents/reviews.

**Subtasks:**

i. Research existing methods for free-text model extraction.
ii. Develop an approach for model ID extraction from free text.
iii. Test and validate the extraction pipeline.
iv. Document the findings for team review.

**Exit Criteria:** A pipeline is created that extracts model IDs and/or brand/model names from the free text in the recall dataset.

### 2.5.11.User Story 11: Modify/Fix Incidents Dataset Merge

**Description:** As a developer, I want to modify the incidents dataset merge logic to create a more accurate dataset merge for accurate model pipelining.

**Subtasks:**

i. Identify merge issues in the current incidents dataset.
ii. Modify the merge logic for accuracy.
iii. Verify the merge results for consistency.
iv. Document the modifications and findings.

**Exit Criteria:** Logic is modified for superior processing, and internal visualizations are created for team discussion.

### 2.5.12.User Story 12: Entity-Relationship Diagram

**Description:** As a developer, I want to create a simplified entity-relationship diagram of the datasets and the merge process that the team has established.

**Subtasks:**

i. Design a draft entity-relationship diagram.
ii. Validate relationships between entities based on the datasets.
iii. Refine and finalize the diagram for team presentation.

**Exit Criteria:** A finalized visualization for external production is created and shared with the team for the final report.

### 2.5.13.User Story 13: Semantic Word Similarity Matching for Sentiment Analysis

**Description:** As a developer, I want to use WordNet and/or SpaCy to match similar words from a starting bag-of-words so that I can establish a semantic word similarity matching process to enhance negative sentiment analysis in Amazon reviews and aid in model training.

**Subtasks:**

i. Develop a starting bag-of-words using relevant words from Amazon reviews and incident reports.

ii. Use WordNet and/or SpaCy to find semantically similar words for each term in the bag-of-words.
iii. Use the matched words to cross-check and refine negative sentiment analysis.

**Exit Criteria:** A comprehensive bag-of-words is created, and semantically similar words are successfully matched, documented, and cross-checked with negative sentiment analysis for further model training.

### 2.5.14.User Story 14: Automation of Review Data Download and Processing

**Description:** As a developer, I want to create a command-line interface that allows selecting different categories of product reviews from the McAuley Labs dataset, and automatically downloads the reviews and metadata, processes them (including language detection and filtering), and merges them with the incidents dataset so that I can efficiently automate the entire data gathering, processing, and integration workflow, which reduces manual effort and improves productivity.

**Subtasks:**

i. Set up and test the command-line interface to ensure it allows selection of categories from McAuley Labs dataset.
ii. Implement automatic download for review data and metadata based on selected categories.
iii. Merge filtered review data with incidents dataset for further analysis.

**Exit Criteria:** The review data processing script is integrated, tested, and functional, allowing automated download, preprocessing, and merging with incident data.

### 2.5.15.User Story 15: Integrate Model IDs from Recall Dataset

**Description:** As a developer, I want to integrate Model IDs from the Recall dataset so that we can accurately match products across datasets for further analysis.

**Subtasks:**

i. Extract Model IDs from Recall dataset.
ii. Clean and format Model IDs.
iii. Verify the integrity and accuracy of Model IDs.

**Exit Criteria:** The Recall dataset has clean, integrated Model IDs with verified accuracy across relevant datasets. The integration process is documented.

### 2.5.16.User Story 16: Model Training for Logistic Regression

**Description:** As a developer, I want to train a Logistic Regression model so that we can classify product safety issues and establish a baseline model for further refinement.

**Subtasks:**

i. Prepare the dataset for model training (split data into training and test sets).
ii. Train the Logistic Regression model.
iii. Evaluate model performance using accuracy, precision, recall, and F1-score.

**Exit Criteria:** A trained Logistic Regression model with a report on accuracy, precision, recall, and F1-score. The model's performance is documented.

### 2.5.17.User Story 17: Descriptive Stats for Logistic Regression Model

**Description:** As a developer, I want to generate descriptive statistics for the Logistic Regression model so that: we can understand key data patterns and improve the model's performance.

**Subtasks:**

i. Identify key features for descriptive statistics.
ii. Calculate statistics (mean, median, mode, variance) for each feature.
iii. Analyze the results to interpret feature importance.
iv. Document findings and suggest any necessary model adjustments.

**Exit Criteria:** A report with descriptive statistics (e.g., mean, median, standard deviation) on key features and model results, providing insights for model improvement.

### 2.5.18.User Story 18: Automation of Review Data Download and Processing

**Description:** As a developer, I want to manually extract the Brand and Manufacturer fields from the Recalls dataset so that this extracted data can be used as features in the Logistic Regression model for predicting product safety risks.

**Subtasks:**

i. Analyze the structure of the dataset and identify where brand and manufacturer details are located within the unstructured recall summaries.
ii. Extract brand and manufacturer names from each recall entry and standardize the data.
iii. Ensure consistency in how brands and manufacturers are labeled.
iv. Create new fields/columns in the dataset for "Brand" and "Manufacturer" and populate them with the extracted information.
v. Perform quality checks on the manually extracted data to ensure accuracy and completeness.

**Exit Criteria:** Brand and Manufacturer fields are correctly extracted for all relevant recall entries, and data is validated and ready for use in the model training pipeline.

### 2.5.19.User Story 19: Recall Data Scrubbing

**Description:** As a Developer, I want to clean and standardize the recall data, handling any inconsistencies or missing entries so that the data is accurate and ready for analysis and merging with other datasets.

**Subtasks:**

i. Identify and address missing values or inaccuracies in the recall dataset.
ii. Standardize brand and manufacturer names to ensure consistency.
iii. Evaluate model performance using accuracy, precision, recall, and F1-score.

**Exit Criteria:** Recall data is cleaned and validated for merging.

### 2.5.20.User Story 20: Merge Refactoring

**Description:** As a Developer, I want to refine the merging process of recall data with the main dataset, ensuring accurate mapping of records so that the merged dataset is comprehensive and supports model training effectively.

**Subtasks:**

i. Optimize merging code to handle inconsistencies and improve efficiency.
ii. Verify merge results for accuracy in matched records.

**Exit Criteria:** Merging process is optimized and validated with correct mappings.

### 2.5.21.User Story 21: Heuristic Rules-Based Model

**Description:** As a Developer, I want to design a heuristic, rules-based model for initial classification based on set patterns and keywords so that I can establish a baseline classification before implementing complex models.

**Subtasks:**

i. Define classification rules based on safety words and other indicators.
ii. Implement and test the rules-based model on sample data.

**Exit Criteria:** Heuristic model successfully identifies key patterns for classification.

### 2.5.22.User Story 22: Aggregation and Descriptive Statistics

**Description:** As a Developer, I want to perform data aggregation and generate descriptive statistics on the cleaned and merged dataset so that I can provide insights on data distribution and inform further analysis steps.

**Subtasks:**

i. Aggregate data at the product level.
ii. Calculate descriptive statistics, including mean, median, and distribution of key variables.

**Exit Criteria:** Summary statistics are generated and documented.

### 2.5.23.User Story 23: XGBoost Model Creation

**Description:** As a Developer, I want to build and train an XGBoost model to improve classification accuracy on safety indicators so that the model can leverage complex patterns for more accurate predictions.

**Subtasks:**

i. Prepare training data for the XGBoost model.
ii. Tune hyperparameters for optimal performance.
iii. Evaluate model metrics and document results.

**Exit Criteria:** XGBoost model is trained, evaluated, and documented.

### 2.5.24.User Story 24: Threshold Tuning for Logistic Regression

**Description:** As a Developer, I want to adjust and fine-tune the threshold values for the Logistic Regression model so that the model can achieve optimal scoring, particularly in balancing precision and recall for product safety classification.

**Subtasks:**

i. Research effective threshold tuning techniques for Logistic Regression.
ii. Implement threshold adjustments based on initial model performance.
iii. Evaluate the effects of threshold tuning on model metrics (accuracy, precision, recall, F1-score).

**Exit Criteria:** A tuned Logistic Regression model with performance improvements from threshold adjustments.

### 2.5.25.User Story 25: Threshold Tuning for XGBoost

**Description:** As a Developer, I want to adjust the threshold values for the XGBoost model so that we can optimize classification results, ensuring a balanced performance across metrics.

**Subtasks:**

i. Identify effective techniques for threshold tuning in XGBoost.
ii. Implement threshold changes and evaluate their impact on model metrics.
iii. Record the threshold values that yield the best balance of precision, recall, and overall accuracy.

**Exit Criteria:** An optimized XGBoost model with recorded threshold settings and performance metrics.

### 2.5.26.User Story 26: SMOTE Ingestion for Data Balancing

**Description:** As a Developer, I want to use SMOTE (Synthetic Minority Over-sampling Technique) to balance the dataset so that the Logistic Regression and XGBoost models can be trained on a balanced dataset, improving classification accuracy for minority classes.

**Subtasks:**

i. Review and select appropriate SMOTE parameters for the dataset.
ii. Apply SMOTE to the training data and confirm data balance.
iii. Analyze the effects of SMOTE on model performance metrics.

**Exit Criteria:** A balanced dataset with SMOTE applied, along with documented improvements in model performance.

### 2.5.27.User Story 27: Big Loop Script Execution and Optimization

**Description:** As a Developer, I want to execute and optimize the Big Loop script to process all 34 product categories so that we can generate category-wise outputs for the heuristic model and analyze product safety data efficiently.

**Subtasks:**

i. Run the script sequentially for each category ensuring each completes without errors.
ii. Monitor the script runtime to avoid system crashes due to high CPU usage.
iii. Evaluate the results for categories like "all beauty" and "baby products" to verify heuristics outputs.
iv. Adjust and refine the loop logic for edge cases where data might not meet quantile criteria.

**Exit Criteria:** The script successfully processes all 34 categories, producing outputs aligned with heuristic model requirements where no major errors are encountered, and runtime stability is maintained for large datasets.

## 2.6 Product Vision

After production, this model will be a command-line interface that is capable of drawing in live data feeds from the McCauley Labs Amazon Reviews dataset. After pulling pipelined data feeds from the chosen categories, it will organize, filter, pre-process, and merge the CPSC incident/recall data with the regarded contextual including incident/recall text data. For those Amazon products that have known incidents/recalls, it will then extract “safety hazard”-like terms like “burned” or “choked” from a bag-of-words with known synonyms.

With these extracted terms, it will cross-reference all Amazon review texts across the product category for products that display a similar amount of “safety hazard”-like terms to flag them as potentially unsafe products.

### 2.6.1 Scenario #1 - CPSC Manufacturer Alerts

With this operational model, the CPSC can create a threshold within a certain adjusted R square that can be used to alert manufacturers that their product is being reviewed with certain cautionary terminology. The product would essentially be a flagger, thus outputting the reviews that the manufacturer can then investigate.

### 2.6.2 Scenario #2 - Manufacturer Pre-Alerts

As above, manufacturers could use it in a similar sense for market research purposes. Is their product meeting standards, and/or what should the engineering tea focus on to meet standards?

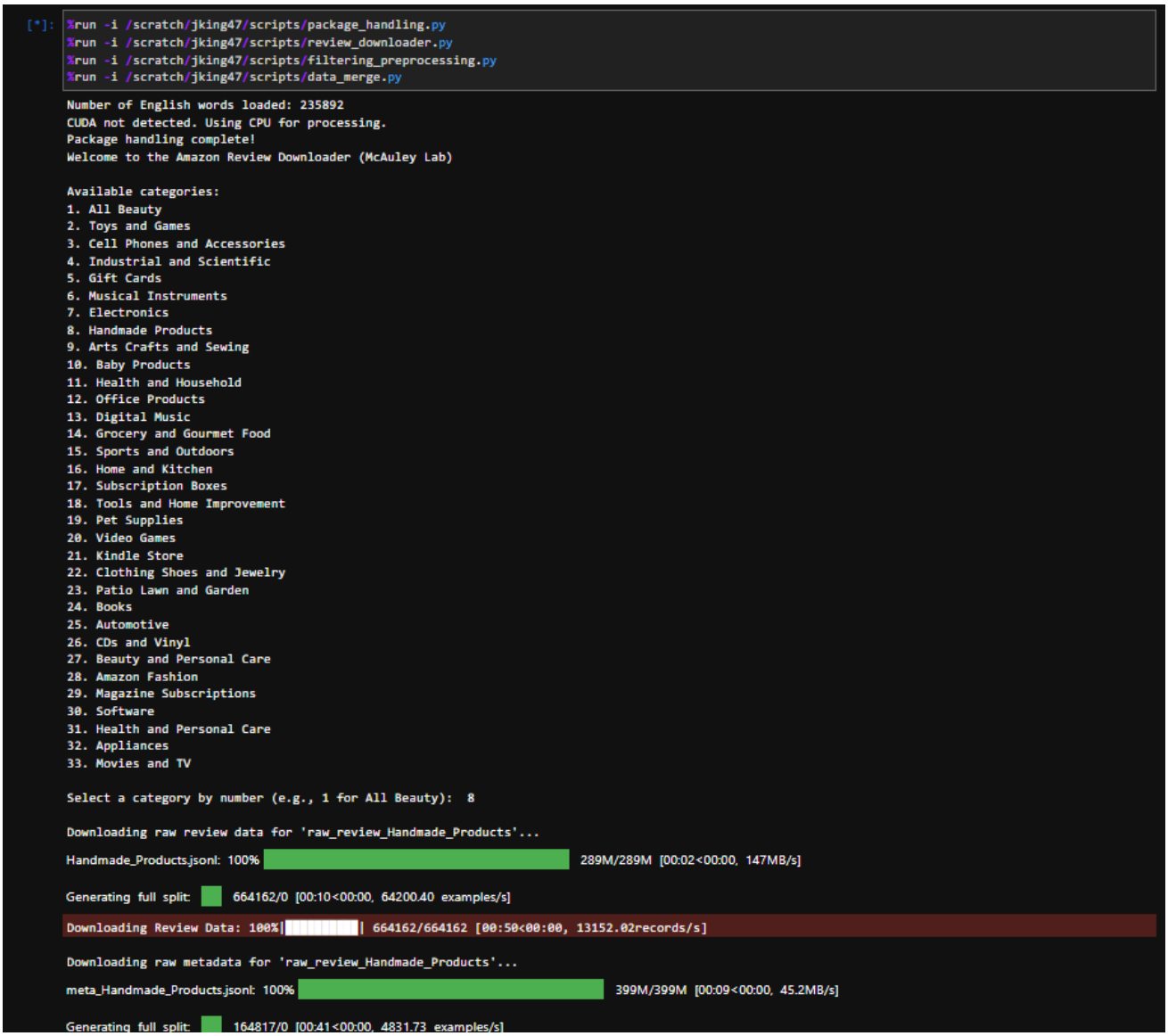



Figure 2: Command Line Interface

# Section 3: Datasets

## 3.1 Overview

This project utilizes multiple datasets focused on product reviews, incidents, and recalls to identify and analyze product safety issues. The primary sources of data include Amazon reviews, incident reports from SaferProducts.gov, and product recall data from public safety databases such as the Consumer Product Safety Commission (CPSC).

The dataset comprises various fields that capture essential information from news articles parsed using the Newspaper 3K module. Below is a detailed description of each field:

### 3.1.1. Amazon Reviews Dataset

Contains user-generated product reviews, including unstructured text reviews, star ratings, and metadata such as product model numbers.

### 3.1.2. Incident Dataset

Documents incidents reported by consumers, detailing specific product safety concerns, incidents, or injuries related to product use.

### 3.1.3. Recall Dataset

Features detailed summaries of products that have been recalled due to safety defects or hazards, including manufacturer information, recall dates, and product descriptions.

## 3.2 Field Descriptions

The dataset comprises various fields that capture essential information from news articles parsed using the Newspaper 3K module. Below is a detailed description of each field:

### 3.2.1. URL (Type: string)

For the **Amazon Review Dataset and Metadata**, the URL is: https://amazon-reviews-2023.github.io/.

For the **Recall and Incident Dataset**, the URL is: https://www.saferproducts.gov/SPDB.zip.

### 3.2.2. Review ID (Type: string)

A unique identifier for each customer review. This field ensures that each review is distinct and can be referenced separately. No two reviews share the same Review ID.

### 3.2.3. Product ASIN (Type: string)

The Amazon Standard Identification Number (ASIN) for each product reviewed. This field links the review to a specific product on Amazon, and each product has a unique ASIN. This field cannot be null as each review is tied to a specific product.

### 3.2.4. Review Text (Type: string)

The unstructured textual data provided by customers in their product reviews. This field contains the free-text feedback from users, often highlighting product features, usage experiences, or concerns. This is the core dataset used for Natural Language Processing (NLP) techniques to extract insights such as sentiment,

product safety concerns, and more. This field may be null in rare cases where a review was left without any text content.

### 3.2.5. Review Rating (Type: integer)

The numerical rating given by the customer, typically on a scale from 1 to 5 stars. This rating provides quantitative feedback that can be combined with textual data to better understand the overall sentiment of the review. This field cannot be null as each review must include a rating.

### 3.2.6. Review Date (Type: datetime)

The date and time when the review was posted, displayed in ISO 8601 format (YYYY-MM-DD Thh:mm +offset). This field helps in understanding trends over time, such as when a surge of negative reviews occurs, which might be indicative of product issues. This field is not allowed to be null.

### 3.2.7. Product Model Number (Type: string)

The model number of the product, if mentioned in the review. This field is crucial for matching reviews with specific Consumer Product Safety Commission (CPSC) recall data. In cases where the model number is not explicitly mentioned, it may be inferred using Named Entity Recognition (NER) or fuzzy matching techniques. This field can be null if not provided in the review.

### 3.2.8. Sentiment Score (Type: float)

A numerical value representing the sentiment of the review, calculated using sentiment analysis models. This field helps categorize reviews into positive, neutral, or negative sentiment. Scores typically range from -1 (most negative) to +1 (most positive). This field is derived and thus cannot be null once the analysis is complete.

### 3.2.9. Recall Status (Type: boolean)

A boolean flag indicating whether the product being reviewed has been subject to a recall, based on data from the CPSC Recalls and Unsafe Product Reports dataset. This field is added during the data integration phase and can take on a value of true (if the product has been recalled) or false (if it has not). This field may be null if no corresponding data is available.

### 3.2.10. Project Information

This project, titled Proactive Identification of Product Safety Issues, aims to utilize the datasets for deriving actionable insights through Natural Language Processing and Machine Learning. The involvement of NIRA, Inc. reflects a commitment to leveraging advanced analytics for consumer safety.

## 3.3 Data Context

The data used in this study brings together complementary information from Amazon reviews and the Consumer Product Safety Commission (CPSC) to analyze product safety. The Amazon reviews dataset, sourced from McAuley Lab's repository, includes approximately 6.6 million reviews and 213,000 metadata entries for baby products. These reviews capture consumer experiences and often highlight potential safety concerns, making them a valuable resource for assessing product risks.

The CPSC datasets include two critical components: an incidents dataset, which contains detailed consumer-reported safety events, and a recall dataset, summarizing official recalls of hazardous products. These datasets provide substantial evidence of safety issues and offer context for consumer responses to product hazards.

To create a comprehensive and harmonized dataset, the project combines these datasets using identifiers such as Amazon Seller Identification Numbers (ASINs) and model numbers. This integration allows us to explore the relationship between consumer feedback and documented safety incidents. By leveraging contextual information such as product metadata, incident descriptions, and safety-related terms from reviews, the project aims to identify patterns in complaints, predict future risks, and support proactive safety measures within the retail sector.

The dataset comprises various fields that capture essential information from news articles parsed using the Newspaper 3K module. Below is a detailed description of each field:

## 3.3.1. Data Sources

#### 3.3.1.1. Amazon Reviews Dataset

This dataset has consumer reviews from Amazon. These reviews provided critical insights into the safety, usability, and overall performance of consumer products. Reviews often include detailed feedback, which can be analyzed through NLP techniques, and can be used to detect product safety issues. However, as the dataset is composed of unstructured text, comprehending the sentiment, context, and meaning of these reviews is important for deriving meaningful insights.

#### 3.3.1.2. CPSC Recalls and Unsafe Product Reports

This dataset has structured data about product recalls and incidents, as reported by the Consumer Product Safety Commission (CPSC). The recall reports provide detailed information about model numbers, product defects, and the reasons for product recalls. This structured data is vital for identifying known safety hazards and linking them to user-reported issues in the Amazon Reviews dataset.

## 3.3.2. Interpreting the context

Each dataset's context affects how it might be interpreted and integrated with other data. For example, An Amazon review stating that a baby product "overheats" while being used, for instance, would not instantly raise concerns about safety, but when paired with past recall information for comparable goods, the situation becomes more apparent and possible dangers are highlighted. Furthermore, a consumer's concern about small parts in a toy may align with previously reported choking hazards in the CPSC dataset, thereby providing a fuller picture.

## 3.3.3. Challenges of Out-of-Context Data

Analyzing reviews without considering the context—such as product recall history, consumer incidents, and industry trends—can lead to misinterpretation of the data. For instance, a surge of negative reviews about a product may be incorrectly attributed to a manufacturing flaw without accounting for possible external factors like recent media coverage, weather conditions, or product misuse. Thus, capturing the full context from multiple sources ensures that the machine learning models can identify the true cause of product safety risks rather than confounding factors.

### 3.3.4. Use of External Data for Context

To improve the accuracy of the analysis, the project might also pull in external data sources, such as social media mentions, news articles, or even weather data, which could influence product performance (e.g., electronics malfunctioning in extreme weather conditions). Additionally, market trends and consumer demand fluctuations could provide context for spikes in product issues or recalls.

## 3.4 Data Conditioning

For this project, data conditioning is the most crucial step in preparing both the Amazon Reviews Dataset and the CPSC Recalls and Incident Dataset for effective analysis. Each dataset requires specific preprocessing steps to ensure consistency, accuracy, and relevance of the data being analyzed.

### 3.4.1. Amazon Reviews Dataset

The Amazon Reviews Dataset contains raw customer reviews, which may include text, ratings, dates, and metadata about the products being reviewed. The following data conditioning steps are necessary for this dataset:

#### 3.4.1.1. Text Preprocessing

The review text needs to be cleaned by removing special characters, HTML tags, and redundant white spaces. Additionally, common NLP steps such as tokenization, stop-word removal, and stemming/lemmatization are required to prepare the data for sentiment analysis and machine learning models.

#### 3.4.1.2. Normalization

The dataset should be normalized by standardizing date formats and ensuring consistency in rating systems (e.g., converting different numerical or textual rating scales to a uniform 5-star system).

#### 3.4.1.3. Duplicate Removal

Duplicates, such as repeated reviews or spam content, must be removed to prevent skewed results in sentiment analysis and risk identification models.

#### 3.4.1.4. Handling Missing Data

Reviews or fields with missing critical information (e.g., product ID or review text) need to be either imputed with estimated values (if applicable) or removed if imputation is not feasible.

### 3.4.2. CPSC Recalls and Incident Dataset

The CPSC dataset includes information on product recalls and unsafe product reports, which contain descriptions, dates, and product metadata. Data conditioning for this dataset involves:

#### 3.4.2.1. Standardization of Recall Descriptions

The recall descriptions may vary in format, so it is important to standardize the text for uniformity, ensuring that the data can be easily analyzed using Natural Language Processing techniques.

#### 3.4.2.2. Date Alignment

Ensure that dates of incidents and recalls are standardized into a consistent format (ISO 8601), similar to the Amazon Reviews Dataset, for easier matching of reviews with relevant recall data.

#### 3.4.2.3. Categorization of Product Types

Product categories and types need to be aligned with the categories found in the Amazon Reviews Dataset. This alignment will allow better cross-referencing of reviews and recall incidents during analysis.

#### 3.4.2.4. Handling Missing or Inconsistent Data

Like the reviews, missing or inconsistent data (e.g., missing product details or improperly formatted recall descriptions) should be cleaned or flagged for further review.

### 3.4.3. Cross-Dataset Conditioning

#### 3.4.3.1. Data Matching

After conditioning each dataset independently, data from the Amazon Reviews and CPSC datasets need to be linked based on product identifiers or descriptions. This will facilitate the matching of review sentiment to reported incidents or recalls.

#### 3.4.3.2. Labeling for Machine Learning

Labeled datasets must be created by tagging Amazon reviews that match products with CPSC recalls. This enables supervised learning models to predict potential product safety issues based on review data.

By applying these data conditioning steps, both datasets will be prepared for in-depth analysis, ensuring the accuracy and effectiveness of the machine learning models and insights derived from the project.

## 3.5 Data Quality Assessment

This section evaluates the quality of the datasets used in our project based on the following attributes: Completeness, Uniqueness, Accuracy, Atomicity, Conformity, and Overall Quality. These attributes provide a comprehensive assessment framework as outlined by Wang & Strong (1996). [53]

### 3.5.1. Amazon Review Dataset

#### 3.5.1.1. Completeness

The dataset is semi-complete due to missing model IDs. Approximately 20% of model IDs are missing from the initial metadata. However, around 80% of the dataset has model IDs, which provides a reasonably comprehensive set for analysis.

#### 3.5.1.2. Consistency

For a non-relational dataset, the Amazon Review dataset shows good consistency. It includes structured data for reviews and metadata with minimal variation in format or structure, making it reliable for further processing.

#### 3.5.1.3. Uniqueness

Metadata is non-unique, which aligns with our goals since reviews are meant to be aggregated. This non-uniqueness allows for the combination of similar reviews and the identification of broader patterns.

#### 3.5.1.4. Integrity

The dataset's integrity is rated as average, with no significant issues of manipulation but room for improvement in maintaining data fidelity.

#### 3.5.1.5. Conformity

The dataset does not always conform to expected standards. For example, model IDs may not always align between different datasets.

#### 3.5.1.6. Accuracy

Given that the dataset is complete and consistent, its accuracy is deemed acceptable. The high presence of model IDs ensures that most reviews can be reliably linked to the correct products.

### 3.5.2. CPSC Recalls and Unsafe Product Reports Dataset

#### 3.5.2.1. Completeness

This dataset suffers from a lack of completeness, with many missing values, particularly in critical fields. This issue affects the dataset's ability to offer comprehensive insights into product safety issues.

#### 3.5.2.2. Consistency

Due to a significant amount of messy user input, the dataset is not consistent. The free-text nature of the data results in inconsistent formats and irregularities that complicate data processing.

#### 3.5.2.3. Uniqueness

The dataset demonstrates uniqueness in terms of free-text input. This uniqueness allows for rich qualitative analysis, but also poses challenges for standardization.

#### 3.5.2.4. Integrity

The integrity of this dataset is low. Given the unstructured nature of the data and the possibility for manipulation (e.g., user-entered text), the reliability of this dataset is compromised.

#### 3.5.2.5. Conformity

The dataset does not conform to predefined standards, making integration with other datasets difficult. This lack of conformity impacts the overall usability of the data.

#### 3.5.2.6. Accuracy:

Due to the issues of inconsistency and integrity, the accuracy of this dataset is questionable. These factors result in a dataset that may not be fully reliable for drawing conclusive insights.

## 3.6 Other Data Sources

For this project, we looked at a variety of possible data sources, including a Web API to which we had access from cpsc.gov. However, upon examining the data accessible via the API, we discovered that it was exactly the same as the datasets we had previously included in our research. We choose not to use the API or any other data sources because of this redundancy. Rather, we focused on the datasets that were supplied, which were sufficiently extensive to satisfy the project's needs and goals.

We also looked at other taxonomy datasets to see whether there was a source where a template of model IDs were easily accessible. To enhance data processing, we set out to extract regular expressions (regex) from model IDs. Unfortunately, we were unable to locate any appropriate dataset that satisfied these requirements; thus we continued cleaning the initially given data sources manually.

## 3.7 Storage Medium

The project dataset is stored on an ORC (Office of Research Computing) cluster at George Mason University. This medium provides a high-performance, secure, and scalable infrastructure designed specifically to handle data-intensive research projects. The ORC cluster allows for efficient data storage and retrieval, which is essential given the volume and variety of data in this project.

Additionally, the ORC cluster enables seamless integration with computational resources for advanced data processing and analysis. By leveraging this storage medium, we ensure that the project data is maintained in an environment optimized for speed, reliability, and ease of access for the team members, while adhering to the data security requirements of the university's protocols.

## 3.8 Storage Security

For this project, the dataset used is publicly available, which means that the data itself is not subject to access restrictions. However, ensuring the security of data storage and computing resources is critical for maintaining the integrity and confidentiality of the project environment.

### 3.8.1. Storage Location

The dataset and all associated computational processes are stored on the **George Mason University ORC Cluster**, a high-performance computing facility that offers robust security measures. The ORC cluster is designed to handle large datasets efficiently while providing a secure environment for sensitive computations.

### 3.8.2. Security Measures for the ORC Cluster

#### 3.8.1.1. Access Control

The ORC cluster is secured with access control protocols to ensure that only authorized users can access and interact with the computing resources and data. Access is granted through secure authentication methods, such as username and password combinations or multi-factor authentication, depending on the specific policies in place

#### 3.8.1.2. Data Encryption

Data stored on the ORC cluster is encrypted to protect it from unauthorized access. This encryption ensures that the data remains confidential during both storage and transmission. Additionally, any computations or data transfers conducted within the cluster adhere to encryption standards to prevent data breaches

#### 3.8.1.3. Physical Security

The physical infrastructure housing the ORC cluster is protected by state-of-the-art security measures, such as restricted access to server rooms, surveillance, and disaster recovery protocols.

#### 3.8.1.4. Compliance with University Policies

The ORC cluster adheres to George Mason University's security policies and practices, which are designed to meet industry standards for data security and privacy. These policies ensure that data is protected from unauthorized access or potential cyber threats

#### 3.8.1.5. Network Security

The ORC cluster is connected to the university's secure network infrastructure, which employs firewalls, intrusion detection systems, and other network security protocols to protect against unauthorized access and potential attacks.

## 3.9 Storage Costs

For this project, we are using the Hopper cluster provided by the Office of Research Computing (ORC) at George Mason University. This compute cluster is free of cost to support research projects within the university. So, there were no direct storage associated with our data storage.

The Hopper Cluster is equipped with a large-scale infrastructure, We are using 12-core processor and 8gb ram per core for the project. The cluster features with A VAST flash-based storage system that provides 2PB of fast scratch storage. This storage medium is optimized for the high-performance computing and supports efficient handling of large datasets, which is essential for our project involving extensive data analysis.

Additional features of the Hopper Cluster include High-speed networking with a redundant Ethernet network (100 Gbps spine switches and 25 Gbps leaf switches) and an HDR InfiniBand network providing 100 Gbps to each node. Access through SSH connections and the ORC Open OnDemand server, which enables web-based access to tools like Matlab and JupyterLab.

Our project benefits from this infrastructure, which enables efficient data processing and storage at no cost, thanks to George Mason University's support for research activities.

# Section 4: Algorithms and Model Analysis

## 4.1 Algorithms and Analysis

### 4.1.1 Algorithms

This section provides an overview of the algorithms employed to classify products as potentially unsafe, utilizing a set of key features derived from product reviews, incidents, and ratings data. The following models were tested: Logistic Regression, XGBoost (Extreme Gradient Boosting), Decision Tree, Naive Bayes, and K-Nearest Neighbors (KNN). Each algorithm was chosen for its distinct strengths in binary classification, interpretability, or performance on imbalanced datasets. [47][49]

#### 4.1.1.1. Logistic Regression

Logistic Regression is a linear model widely used in binary classification tasks. The model applies the logistic (sigmoid) function to a linear combination of features, transforming the results into a probability score. This score is used to classify products as either "safe" or "unsafe." [47]

For this project, the Logistic Regression model was trained on the following features, chosen for their potential to correlate with product safety issues:

- **Average Rating**: Low ratings might indicate dissatisfaction related to safety.
- **Review Count**: More reviews may indicate a higher likelihood of safety-related reports.
- **Average Sentiment Score**: Negative sentiment might suggest safety concerns.
- **Safety Term Density**: Frequency of safety-related words (e.g., “defective”) in reviews.
- **Incident Review Match Rate**: Correlation between review content and official incidents.

Logistic Regression was configured with class_weight='balanced' and adjusted for regularization (C=0.05) to address class imbalance. The threshold for predicting unsafe products was raised to 0.9 to reduce false positives.

#### 4.1.1.2. XGBoost (Extreme Gradient Boosting)

XGBoost is a powerful ensemble method that builds sequential decision trees, with each tree correcting errors from the previous one. This algorithm is robust against imbalanced datasets and can handle complex feature interactions effectively. [49]

XGBoost was trained on the same features as Logistic Regression and fine-tuned with:

- **max_depth=4**: Restricting tree depth to reduce overfitting.
- **learning_rate=0.05**: Slowing down the learning process for stability.
- **n_estimators=100**: Building a moderate number of trees for robustness.

To address class imbalance, scale_pos_weight was set based on the safe-to-unsafe product ratio. Like Logistic Regression, XGBoost's prediction threshold was raised to 0.9 to minimize false positives.

#### 4.1.1.3. Decision Tree

Decision Tree is a non-linear model that splits data into branches based on feature values, which makes it more adaptable to non-linear relationships and interpretable.

The Decision Tree model was configured with parameters optimized to handle class imbalance and overfitting issues. Decision Trees inherently work well with both numeric and categorical data, and thus they provided a valuable comparison to the other linear and ensemble models in this project.

#### 4.1.1.4. Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming independence among features. While simple, Naive Bayes can be effective for text-based data like reviews, as it captures the probabilistic distribution of terms associated with safety issues.

The Naive Bayes model was applied to this dataset to check if probabilistic modeling would help in identifying potentially unsafe products by focusing on the term frequencies associated with safety concerns.

#### 4.1.1.5. K-Nearest Neighbors (KNN)

K-Nearest Neighbors is an instance-based learning algorithm that classifies products based on the proximity to other instances. For each product, KNN finds the most similar products and assigns the class based on a majority vote.

Due to its sensitivity to data imbalance and scalability limitations with large datasets, KNN was included for comparison, as it operates differently from other classifiers, focusing on instance similarity rather than feature relationships.

### 4.1.2 Analysis

This section evaluates the performance of the models, focusing on their ability to correctly classify products as safe or unsafe. Given the severe consequences of misclassifying an unsafe product, a high threshold (0.9) was used to prioritize precision over recall, reducing false positives.

#### 4.1.2.1. Feature Scaling and Preparation

To prepare the features for model training, the following preprocessing steps were performed:

- **Standard Scaling**: Applied to all features to normalize them, especially beneficial for Logistic Regression. [47]
- **Class Balancing**: Both Logistic Regression and XGBoost were configured with balancing techniques, while Naive Bayes and KNN handled class imbalance differently. [49]

#### 4.1.2.2. Model Performance Evaluation (*Refer to Table 1 on Page 45*)

#### 4.1.2.3. Confusion Matrix Analysis

The confusion matrix provides further insight into the classification performance: [54]

The confusion matrices revealed that most models excelled at classifying safe products but struggled with the minority class (unsafe products). Only the Decision Tree achieved slightly better recall for unsafe products, but this came at the expense of a higher false-positive rate.

#### 4.1.2.4. Key Observations

- **True Negatives (TN):** Most models had high true negatives, accurately identifying safe products.
- **False Positives (FP):** Logistic Regression, Naive Bayes, and XGBoost had lower false positives due to the strict threshold.

- **True Positives (TP):** All models except Decision Tree failed to detect unsafe products, resulting in near-zero recall.
- **False Negatives (FN):** All models had difficulty in correctly identifying unsafe products, with XGBoost and KNN completely missing them.

#### 4.1.2.5. Summary and Insights

- **Logistic Regression:** High precision but low recall for unsafe products. It struggled to capture unsafe cases even with class weighting.
- **XGBoost:** High accuracy for safe products, but failed to detect unsafe products due to the strict threshold.
- **Decision Tree:** Slightly better recall for unsafe products but higher false positives, showing potential for tuning.
- **Naive Bayes:** Good at detecting safe products but missed all unsafe products due to class imbalance.
- **KNN:** Performed well for safe products but failed to detect any unsafe products.

Each model showed strengths in identifying safe products but struggled with recall for unsafe products due to class imbalance. Future steps could include lowering the threshold for higher recall, applying synthetic oversampling (e.g., SMOTE) for unsafe products, or experimenting with ensemble techniques for better balance between precision and recall.

## 4.2 Machine Learning Model Exploration and Selection

### 4.2.1 Model Exploration

In this project, several machine learning models were explored to identify potentially unsafe products based on a combination of structured and unstructured data. The models chosen represent a variety of algorithms with different approaches to handling data, including linear models, tree-based models, probabilistic models, and instance-based methods. Each model's exploration was guided by the unique challenges of the dataset, particularly the class imbalance between safe and unsafe products.

**Explored Models**

#### 4.2.1.1. Logistic Regression

Chosen for its simplicity and interpretability, Logistic Regression is effective for binary classification tasks and can be adjusted for class imbalance using class weighting. It provides insights into the influence of each feature on the outcome, making it useful for understanding factors correlated with unsafe products.

#### 4.2.1.2. XGBoost (Extreme Gradient Boosting)

A powerful ensemble method, XGBoost is robust to imbalanced data and capable of capturing complex patterns through boosted decision trees. XGBoost's scale_pos_weight parameter was tuned to address the imbalance, making it a suitable candidate for handling minority classes.

#### 4.2.1.3. Decision Tree

This non-linear model was selected for its ability to handle both categorical and continuous data, as well as for its interpretability. Decision Trees work well without extensive data preprocessing and

provide an easily understandable decision-making process, especially valuable for identifying specific safety patterns in product data.

#### 4.2.1.4. Naive Bayes

A probabilistic model, Naive Bayes is often effective in text-heavy datasets like review data due to its simplicity and speed. Despite its independence assumption between features, Naive Bayes was included as it can perform well with high-dimensional, sparse data.

#### 4.2.1.5. K-Nearest Neighbors (KNN)

An instance-based method, KNN was explored as it classifies based on similarity to neighboring data points. Though sensitive to class imbalance and large datasets, KNN was included for comparison due to its unique approach compared to other models.

#### 4.2.1.6. Isolation Forest

Isolation Forest is an unsupervised machine learning algorithm used for anomaly detection or outlier detection. It is particularly effective for identifying rare or unusual data points in large datasets. The core idea behind the algorithm is that anomalies are few and different from the majority of the data, and thus, they can be isolated with fewer splits compared to normal data points.

**Exploration Insights**

The baseline model was Logistic Regression which achieved a high accuracy, but low recall for the minority class of unsafe products. XGBoost, an offshoot of logistic regression designed for imbalanced datasets based on gradient boosting, did enhance accuracy but was still unable to identify the unsafe products properly. Naïve Bayes, who could analyze texts well, performed poorly as it implicitly assumed that all the features were independent, which restricted the recall to safe but missed unsafe cases. The Decision Tree model however provided better interpretability as well as balance in precision recall tradeoff but there was still underperformance. K-Nearest Neighbors(KNN) did not cope well with higher dimensionality and class imbalance of the dataset which led to poor performance. K-Means clustering was used to perform clustering with products based on the product features. The method was successful in discovering the clusters but failed in classifying the products in the groups of safe and unsafe products.

The heuristic model approach demonstrated superior performance when applied with the specified threshold values, effectively delivering the desired results. Lastly, the Isolation Forest have performed well as rigorous anomaly detection methods based on many features to identify points that may be outliers with respect to safety. It was the most suitable model for the task due to its capacity to work with high dimensional data while detecting anomalies.

### 4.2.2 Model Selection

After evaluating various machine learning models, we found that traditional approaches like Logistic Regression, XGBoost, and Decision Tree struggled to effectively identify unsafe products due to the significant class imbalance in the dataset. These models demonstrated high precision for the majority class (safe products) but consistently exhibited poor recall for the minority class (unsafe products), which is critical for our goal of proactively identifying safety risks. As a result, we selected a heuristic model as the most effective solution for this project.

#### 4.2.2.1. Heuristic Model Approach

The heuristic model leverages specific metrics and quantile-based thresholds to classify products as potentially unsafe based on explicit indicators of safety risks in customer reviews and incident data.

This rule-based approach enabled us to achieve higher recall for unsafe products without the need for complex machine learning techniques. The key elements of the heuristic model include:

#### 4.2.2.1.1. Safety Term Density

Calculated as the average number of safety-related terms per review, helping to identify products where safety-related language (e.g., "hazard," "defective") is frequently mentioned. This metric serves as an indicator of potential safety concerns.

#### 4.2.2.1.2. Incident-Review Match Rate

Measures the overlap between terms found in official incident reports and those in customer reviews. A high match rate suggests that customer-reported issues closely align with documented incidents, increasing the likelihood of safety concerns.

### 4.2.2.2. Quantile-Based Thresholds

To effectively filter potentially unsafe products, quantile-based thresholds were set for key metrics:

#### 4.2.2.2.1. Rating Threshold

Products with an average rating in the lower 25th percentile.

#### 4.2.2.2.2. Sentiment Threshold

Products with an average sentiment score in the lower 25th percentile.

#### 4.2.2.2.3. Safety Term Density Threshold

Products with a safety term density in the upper 75th percentile.

#### 4.2.2.2.4. Incident-Review Match Rate Threshold

Products with an incident-review match rate in the upper 75th percentile.

#### 4.2.2.2.5. Defining Potentially Unsafe Products

- Average Rating Threshold: Products with average rating below the $25^{th}$ percentile were labeled as potentially unsafe.
- Sentiment Score Threshold: Products with sentiment scores below the $25^{th}$ percentile were flagged as unsafe.
- Safety Term Density: Reviews containing a high density(top 25%) of safety related terms, such as “burned” or “chocked”, contributed to labelling the products as unsafe.
- Incident-Review Match Rate: The overlap of safety-related terms between CPSC incident reports and Amazon reviews was calculated, and products with a high match rate(top 25%) were flagged unsafe.
- Presence if incidents or recalls: Products with known CPSC incidents or recalls were also labeled as unsafe.

This rule-based approach allowed us to achieve higher recall for unsafe products by focusing on specific, interpretable criteria that indicate safety risks. The heuristic model’s efficiency, interpretability, and reliance

on explicit safety signals make it the ideal choice for this project, aligning well with our objective of proactively identifying product safety risks.

#### 4.2.2.3. Isolation Forest

After evaluating multiple modeling techniques, the Isolation Forest algorithm was selected as the most suitable approach for building a robust pipeline to classify unsafe items in the Amazon baby products review dataset. This decision was based on the following strengths.

##### 4.2.2.3.1. Key Feature of Isolation Forest

**Outlier Detection Capability**

Isolation Forest is well-suited for identifying outliers that could indicate safety concerns. Its tree-based structure partitions data across multiple dimensions, making it efficient in detecting products with unusual patterns that may pose risks.

##### 4.2.2.3.2. Strengths of Isolation Forest

- **Unsupervised Approach**

  The algorithm's unsupervised nature proved valuable, especially given the lack of sufficient labeled data. It allowed the pipeline to focus exclusively on detecting anomalies without requiring predefined labels.

- **Multidimensional Analysis**

  Isolation Forest effectively incorporated various safety-related features, such as average rating, review counts, sentiment scores, safety term density, and incident review match rates. This ensured a comprehensive comparison of product behaviors across multiple parameters.

- **Scalability**

  The algorithm's tree-based structure efficiently handled the large dataset of 70,000 products, enabling the analysis of a vast number of product reviews without compromising performance.

##### 4.2.2.3.3. Role of Isolation Forest in the Pipeline

- **Anomaly Detection**

  The algorithm flagged products with unusual patterns, such as low average ratings combined with high safety term density or significant incident review match rates, identifying those likely to pose safety risks.

- **Prioritization for Human Review**

  Products flagged by the model were prioritized for further evaluation by human reviewers, ensuring that critical safety issues and edge cases were thoroughly addressed.

- **Integration with Heuristic Filters**

  Isolation Forest worked in tandem with heuristic-based filters to identify anomalies that did not conform to predefined rules, allowing the pipeline to account for unexpected and unstructured product behaviors.

#### 4.2.2.3.4. Enhanced Pipeline Design

The inclusion of Isolation Forest significantly improved the pipeline's ability to detect outliers, even in the presence of severe class imbalance and unlabeled data. This approach provided a cost-effective, scalable, and reliable solution for identifying unsafe products. It also ensured the pipeline could handle both general cases and critical edge cases efficiently.

The use of Isolation Forest enhanced the heuristic approach by accurately identifying anomalies, paving the way for future advancements in real-time safety monitoring. This strengthened pipeline now serves as a powerful tool for ensuring consumer safety and evaluating product risks.

## 4.3 Solution Approach

### 4.3.1 Systems Architecture

The system architecture for the heuristic model involves multiple layers of data preprocessing, feature extraction, and filtering to classify products as potentially unsafe. The architecture follows a modular design, ensuring scalability, efficiency, and maintainability. The key components of the system are:

#### 4.3.1.1. Data Ingestion Layer

This layer collects data from multiple sources, including product reviews, incident reports, and historical recall records. Data pipelines are designed to handle structured (e.g., ratings, review counts) and unstructured (e.g., review text) data formats.

#### 4.3.1.2. Preprocessing and Feature Engineering Layer

Data is cleaned, tokenized, and processed to extract relevant features like:

- Safety Term Count and Density.
- Incident-Review Match Rate.
- Average Rating and Sentiment Score.
- Quantile-based thresholds are calculated at this stage for filtering.

#### 4.3.1.3. Classification Layer

The heuristic model applies rule-based logic to classify products as potentially unsafe based on predefined conditions:

- Thresholds for ratings, sentiment scores, safety term density, and match rates.
- Products meeting these conditions are flagged as potentially unsafe and moved to the output layer.

#### 4.3.1.4. Output and Reporting Layer

Results are stored in a database or exported to files (e.g., checkpoint6.csv), ensuring easy access for further analysis. Dashboards or summary reports can be generated to provide insights into flagged products and their associated risks. This architecture emphasizes modularity and efficiency, allowing easy updates to rules and thresholds as more data becomes available.

### 4.3.2 Systems Security

Our model is system-agnostic and leverages open-source software, Python packages, and open-source data throughout. This model study does not require any system security.

### 4.3.3 Systems Data Flows

The system's data flow ensures seamless integration of data sources, preprocessing, and classification logic. The steps in the data flow are:

#### 4.3.3.1. Input Data Flow

- Data from multiple sources (e.g., product reviews, incident records) is ingested into the system.
- Structured data like ratings and counts flows into feature calculation modules, while unstructured text is tokenized and analyzed for safety terms.

#### 4.3.3.2. Feature Extraction Flow

- Key metrics, such as safety term density and incident-review match rate, are calculated in parallel.
- Quantile-based thresholds are derived using batch processing to filter data.

#### 4.3.3.3. Classification Flow

- Filtered features flow into the heuristic classification module.
- Rule-based logic applies thresholds and conditions to flag potentially unsafe products.

#### 4.3.3.4. Output Data Flow

- Classified results are stored in a database or exported as CSV files for further analysis.
- Data flows into reporting and visualization modules to generate actionable insights for stakeholders.

# Section 5: Visual Data Insights

## 5.1 Visualizations

### 5.1.1 Number of Reviews Over Time

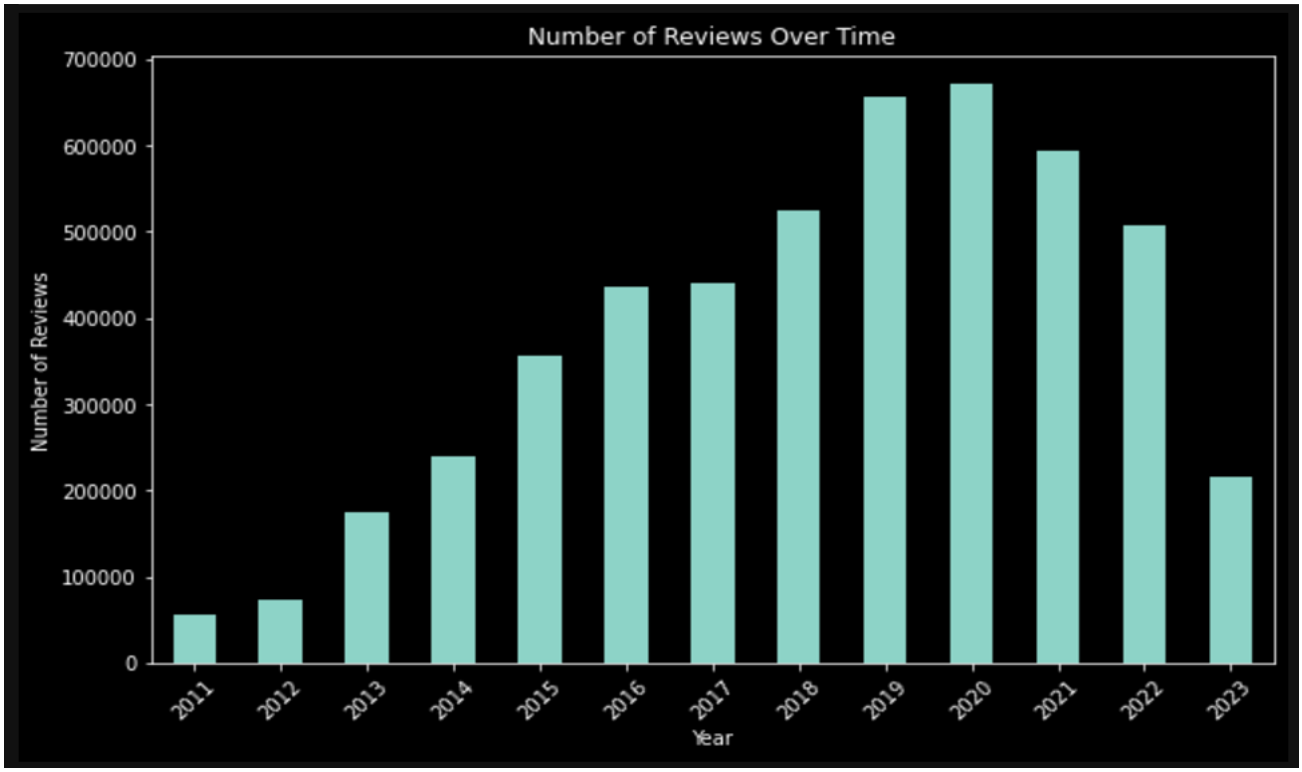



Figure 3: Analysis of Review Trends Over Time

The bar chart titled "**Number of Reviews Over Time**" provides an overview of the volume of customer reviews recorded annually from 2011 to 2023. Below is a detailed analysis of the observed trends:

**Growth Period (2011–2017):** From 2011 to 2017, the number of reviews increased steadily, indicating a growing engagement from consumers. This growth could reflect an expanding customer base, greater adoption of online review platforms, or a surge in product availability and sales.

**Peak Period (2017–2019):** The number of reviews reached its peak between 2017 and 2019, with 2019 marking the highest point. This period may signify a time of maximum platform activity or widespread consumer interest.

**Decline Period (2020–2023):** A noticeable decline in reviews occurred from 2020 onwards, with a sharper drop in recent years (2022–2023). Factors contributing to this decline might include:

- The impact of external events, such as the COVID-19 pandemic, which may have disrupted consumer habits or limited purchases.
- Changes in platform policies, such as stricter review moderation.
- Shifting consumer behavior toward other forms of product feedback.

**Recent Trend (2023):** The steep decline in reviews in 2023 raises questions about ongoing engagement levels. It may reflect platform saturation, a reduction in product diversity, or changing customer preferences.

**Implications:**

The declining trend since 2019 suggests a need to investigate underlying causes. This could involve analyzing:

- Changes in market conditions.
- Shifts in customer feedback preferences.
- Platform-related policies or technological updates.

Understanding these factors is crucial for adapting strategies to enhance customer engagement and maximize value from user-generated content.

### 5.1.2 Rating Distribution

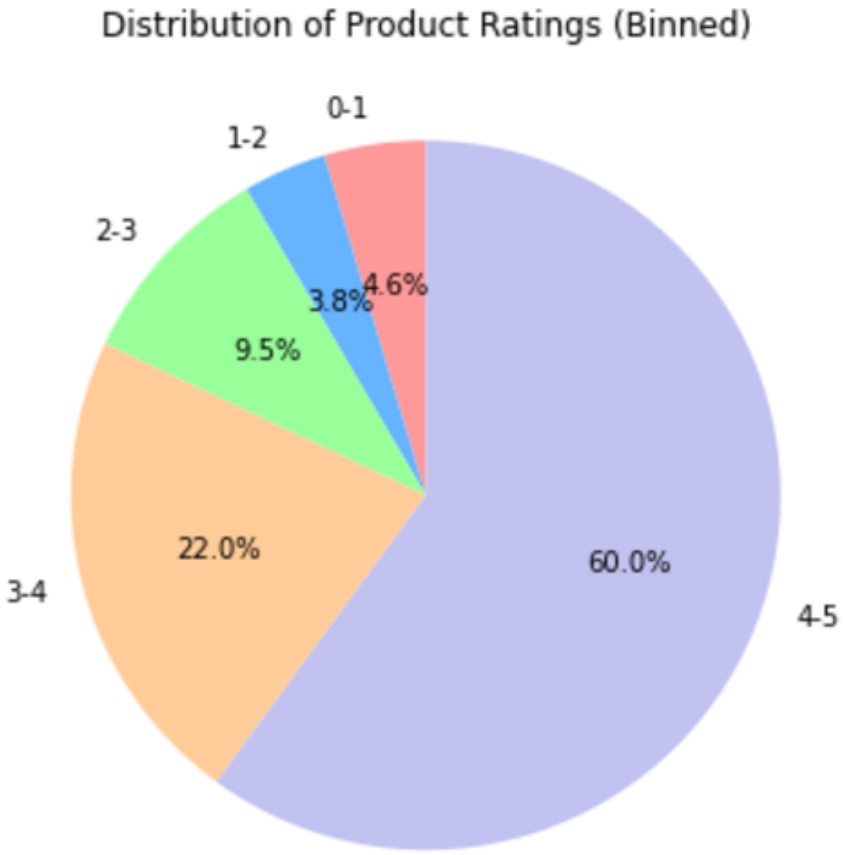



Figure 4: Distribution of Product Ratings

The **"Distribution of Product Ratings"** pie chart provides a breakdown of ratings assigned by customers, highlighting their levels of satisfaction. The following insights are observed:

**High Ratings**:

- A significant majority **60.0%** of the ratings are between **4-5**, showcasing a trend of overwhelmingly positive feedback.
- An additional **22.0%** of the ratings are between **3-4**, indicating that a combined **82.0%** of the ratings are favorable.

**Neutral to Negative Ratings:**

- Ratings between 2-3 and 1-2 account for **9.5%** and **3.8%,** respectively, suggesting limited occurrences of neutral or below-average satisfaction.
- The smallest category, 0-1 rating, comprises **4.6%**, representing strongly dissatisfied customers.

**Implications:**

The data reflects a strong inclination toward positive product experiences among customers. However, the 4.6% of strongly negative ratings and the 8.4% combined share of 0–3 ratings could warrant further investigation to identify recurring issues and improve product performance or customer experience.

### 5.1.3 Sentiment Analysis

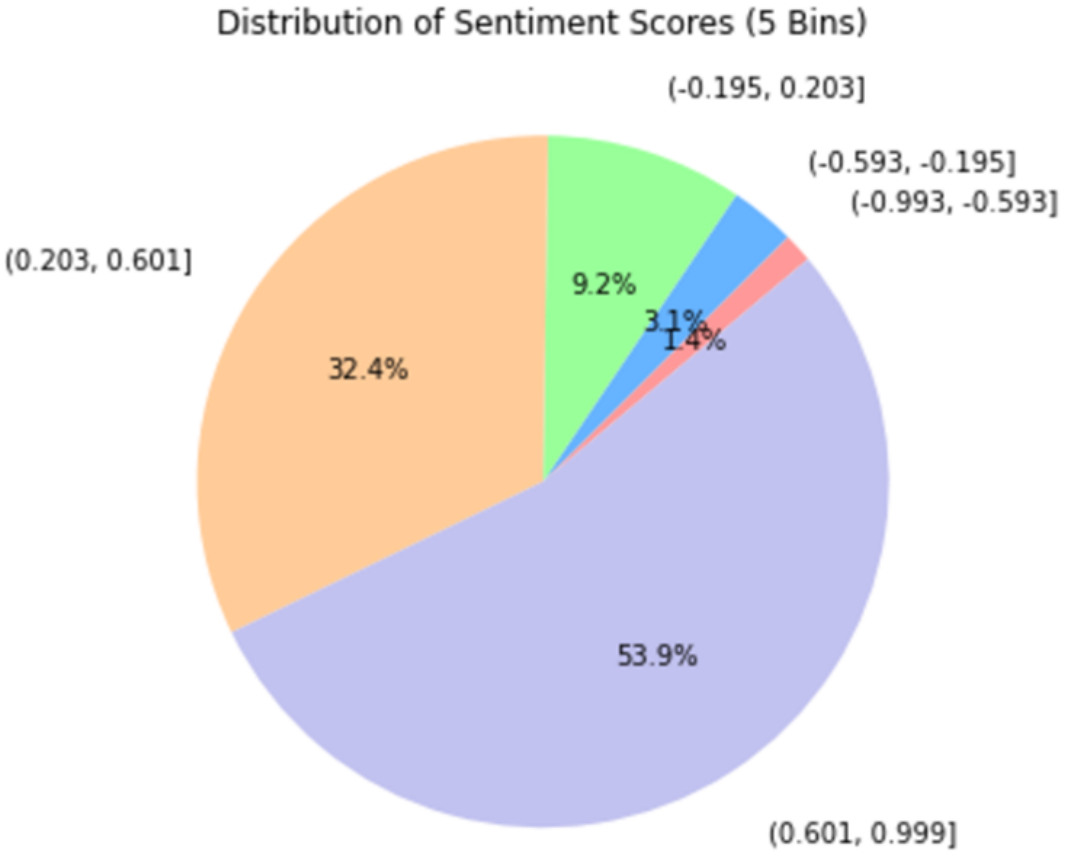



Figure 5: Sentiment Score Distribution Analysis

The **"Distribution of Sentiment Scores"** pie chart illustrates the percentage of sentiment scores across five predefined ranges. The following observations are notable:

**Strong Positive Sentiment**: The largest proportion of scores (53.9%) lies in the range **(0.601, 0.999]**, indicating a strong prevalence of highly positive sentiments.

**Moderate Positive Sentiment**: A smaller, but still significant, proportion (32.4%) falls in the **(0.203, 0.601]** range, reflecting moderately positive feedback.

**Neutral Sentiment**: Approximately **9.2%** of scores are within the **(-0.195, 0.203]** range, representing neutral sentiment.

**Slightly Negative Sentiment**: The range **(-0.593, -0.195]** accounts for **3.1%**, showing minor dissatisfaction or neutral-leaning feedback.

**Strong Negative Sentiment**: Only **1.4%** of the scores fall within the strongly negative range **(-0.993, -0.593]**, suggesting limited occurrences of strongly negative sentiment.

**Implications:**

The overwhelming dominance of positive sentiment indicates a generally favorable reception of the products or services analyzed. The relatively small proportion of neutral and negative sentiment suggests that most feedback is skewed towards satisfaction.

### 5.1.4 Histogram for Sentiment Score Distribution Analysis

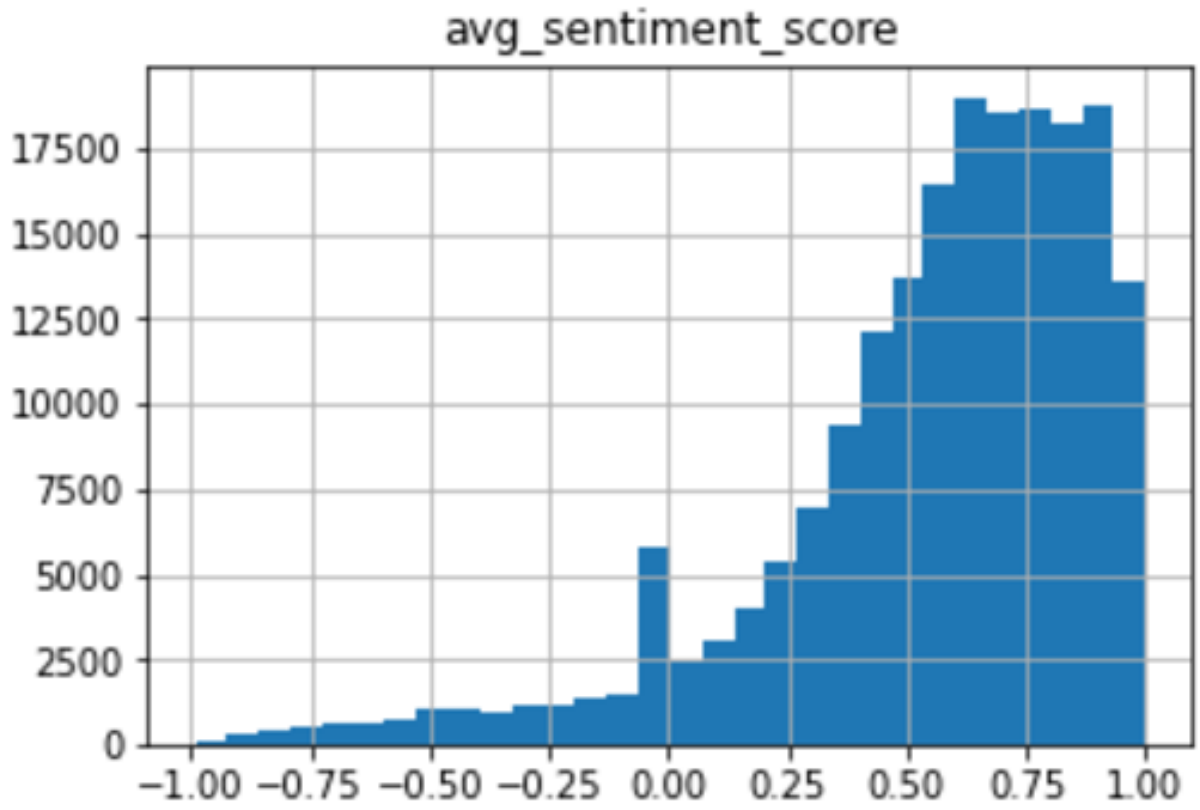



Figure 6: Average Sentiment Score Distribution Analysis

The **"Histogram of Sentiment Scores"** provides insights into the frequency of sentiment scores across various ranges. Key observations include:

**Positive Sentiment Dominance:** Sentiment scores in the range of **0.5 to 1.5** exhibit the highest frequencies, with a peak in the upper positive sentiment range (**1.0 to 1.5**). This suggests a strong skew toward positive sentiment in the dataset.

**Neutral Sentiment Cluster:** A moderate frequency is observed near **0**, indicating a cluster of neutral or slightly negative feedback.

**Limited Negative Sentiment:** Sentiment scores below **0** occur with much lower frequency, indicating fewer instances of strongly negative feedback.

**Implications:**

The histogram underscores a predominance of positive sentiment across the dataset, suggesting overall customer satisfaction. However, the neutral and negative clusters could warrant further investigation to identify specific areas for improvement.

### 5.1.5 Correlation Matrix Analysis

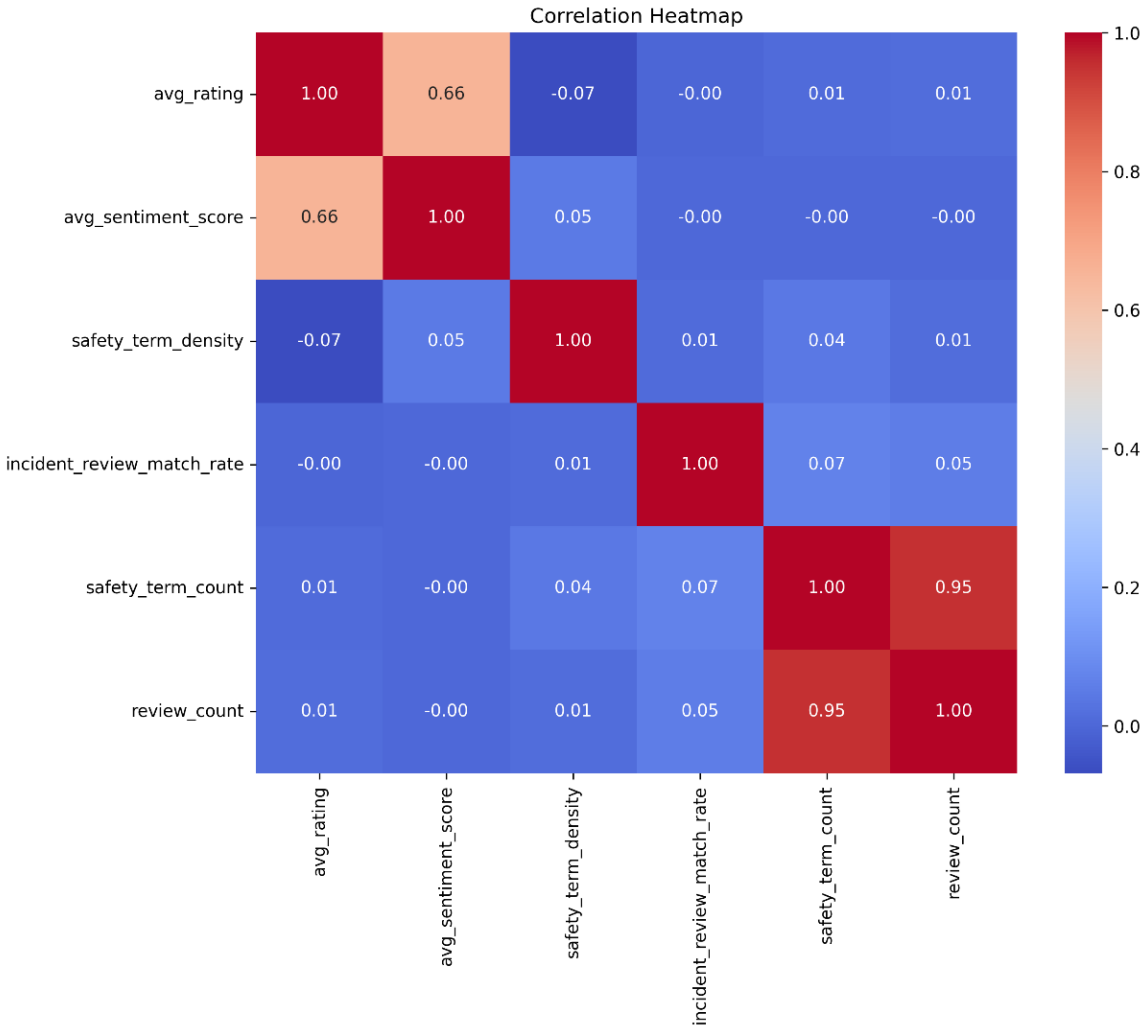



Figure 7: Correlation Heatmap

The correlation matrix serves as a statistical summary to understand the relationships between key numerical variables in the dataset. This analysis is instrumental in identifying patterns and dependencies that can guide further decision-making and model development.

**Key Variables in the Correlation Matrix:**

- **avg_rating**: Average product rating provided by users.

- **avg_sentiment_score**: Sentiment polarity derived from product reviews (ranging from negative to positive).
- **safety_term_density**: The frequency of safety-related terms in reviews normalized by text length.
- **incident_review_match_rate**: The proportion of incidents that overlap with review terms.
- **safety_term_count**: The total count of safety-related terms in the dataset.
- **review_count**: The total number of reviews per product.

**Observations from the Correlation Matrix:**

**Strong Correlations**:

- **safety_term_count and review_count**: Correlation = **0.95**
  This strong positive correlation suggests that products with more reviews tend to have a higher count of safety-related terms, indicating that increased engagement may reveal more safety issues.
- **avg_rating and avg_sentiment_score**: Correlation = **0.66**
  A significant positive relationship exists between user ratings and sentiment scores, indicating that higher ratings align with positive sentiment in the reviews.

**Weak or Insignificant Correlations**:

- **avg_rating and safety_term_density**: Correlation = **-0.07**
  This weak negative correlation implies that the density of safety terms does not strongly influence the overall product rating.
- **avg_sentiment_score and incident_review_match_rate**: Correlation = **-0.00**
  There is no meaningful relationship between sentiment scores and the rate at which incidents match review terms.

**Unique Insights**:

- **safety_term_density and safety_term_count**: Correlation = **0.04**
  The weak correlation suggests that while products may have a high count of safety terms, their density (normalized frequency) might vary due to differences in review lengths or overall engagement.

The correlation matrix provides valuable insights into the interplay between review characteristics and safety-related terms. While some variables exhibit strong dependencies (e.g., safety_term_count and review_count), others show weak or no correlations, suggesting opportunities for further research and feature engineering. This analysis lays the groundwork for improving safety standards and enhancing product quality through data-driven decision-making.

### 5.1.6 Scatter Matrix of Ratings and Sentiments

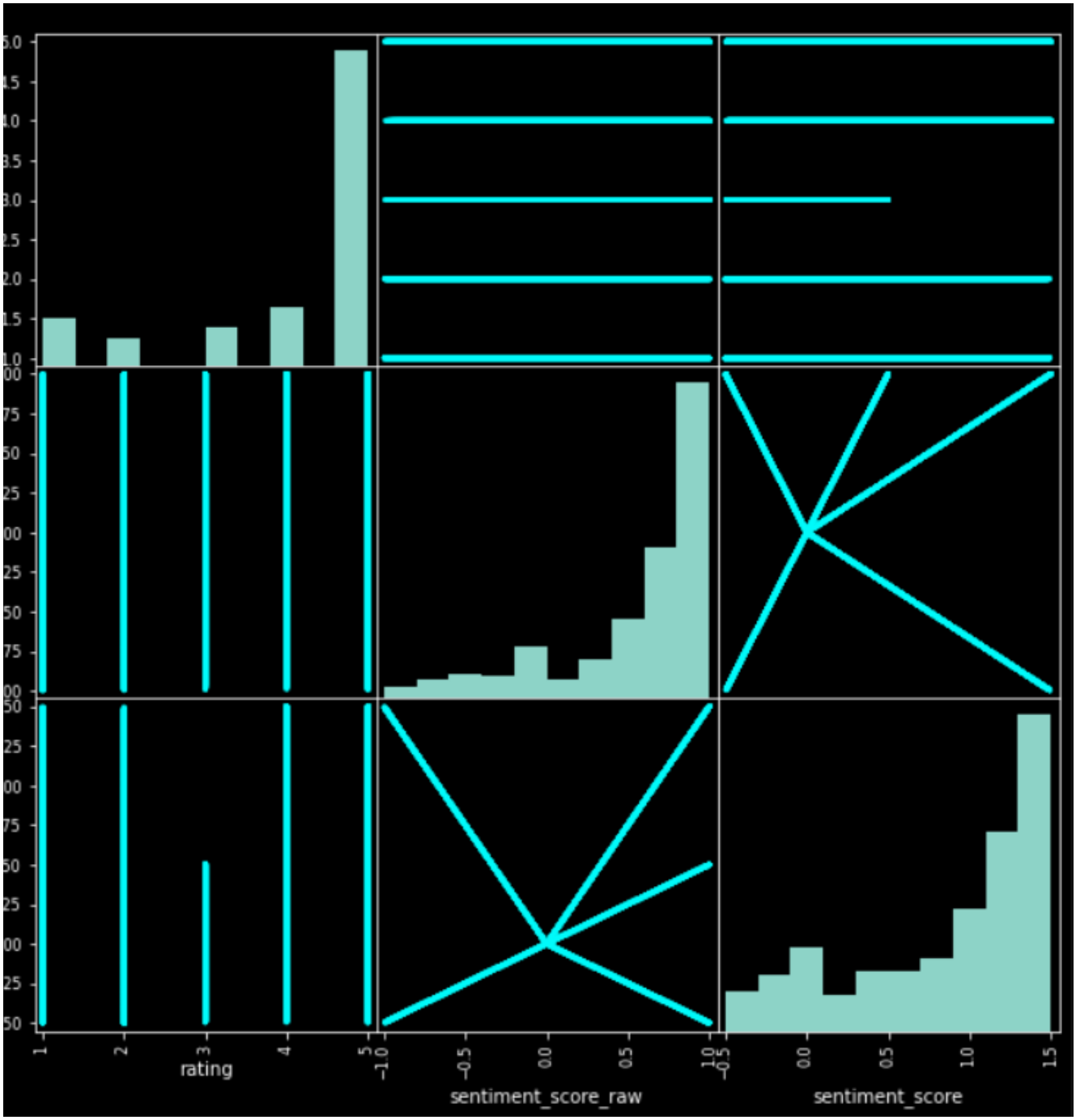



Figure 8: Pair Plot Analysis

The pair plot visualizes the relationships and distributions between rating, sentiment_score_raw, and sentiment_score. Below are the key observations:

**Distributions:**

- **Rating:** Most ratings are concentrated at the maximum value of **5.0**, indicating a predominance of highly favorable feedback.

- **Sentiment Scores:** Both sentiment_score_raw and sentiment_score show positively skewed distributions, reflecting a tendency towards positive sentiment.

**Variable Relationships:**

- **Rating vs. Sentiment Scores:**
  There is a clear alignment between higher ratings and higher sentiment scores, suggesting a strong positive correlation between these variables.
- **Sentiment Scores Relationship:**
  A linear relationship is observed between sentiment_score_raw and sentiment_score, indicating consistency in the scoring methodology.

**Implications:**

- The alignment between high ratings and high sentiment scores indicates the reliability of sentiment analysis as a proxy for user satisfaction.
- The predominance of positive ratings and sentiment scores suggests high overall satisfaction but leaves room for investigating outliers in lower ratings.

### 5.1.7 Highest and Lowest Reviewed Products

As part of our team's effort to analyze product reviews and safety concerns, we've identified two key areas for evaluation: the most-reviewed products (indicative of high engagement and satisfaction) and the lowest-reviewed products (indicating potential dissatisfaction or safety risks). By analyzing customer sentiment, average ratings, and safety-related terms, we aim to uncover recurring themes and provide actionable insights for product improvement and risk management.

Top 10 Reviewed Unique Products:

| parent_asin | title | review_count | avg_rating | involved_in_incident | safety_terms |
|---|---|---|---|---|---|
| b00echytbi | Infant Optics DXR-8 Video Baby Monitor, Non-Wi... | 27282 | 4.198629 | False | [] |
| b075qq8vzw | iHealth No-Touch Forehead Thermometer, Digital... | 14693 | 3.618186 | False | [] |
| b0bg6jynqx | Regalo Easy Step 38.5-Inch Wide Walk Thru Baby... | 13907 | 4.025814 | False | [] |
| b0bb84jxs9 | VTech Sit-to-Stand Learning Walker (Frustratio... | 9197 | 4.639121 | False | [] |
| b07xm9dx9h | Diapers Newborn/Size 1 (8-14 lb), 84 Count - P... | 7959 | 4.099510 | False | [] |
| b0bngy8dwb | Shynerk Baby Car Mirror, Safety Car Seat Mirro... | 7814 | 4.845278 | False | [] |
| b0c5jmrgrd | Regalo Easy Step Extra Tall Walk Thru Baby Gat... | 7788 | 4.026965 | False | [] |
| b00oo9k5qm | Regalo My Cot Portable Travel Bed, Includes Fi... | 7255 | 4.310682 | False | [] |
| b09b8lct53 | Summer Multi-Use Decorative Extra Tall Safety ... | 6981 | 4.317720 | False | [] |
| b00oqczavw | Baby Banana Yellow Banana Infant Toothbrush, E... | 6944 | 4.644297 | False | [] |

| parent_asin | text_safety_terms |
|---|---|
| b00echytbi | [overheating, trap, fracture, injury, unstable... |
| b075qq8vzw | [overheating, trap, injury, fracture, unstable... |
| b0bg6jynqx | [trap, fracture, unstable, injury, damaged, br... |
| b0bb84jxs9 | [trap, fracture, unstable, injury, damaged, mi... |
| b07xm9dx9h | [trap, fracture, unstable, injury, damaged, mi... |
| b0bngy8dwb | [trap, fracture, unstable, injury, damaged, mi... |
| b0c5jmrgrd | [trap, injury, unstable, fracture, damaged, mi... |
| b00oo9k5qm | [trap, injury, fracture, unstable, damaged, mi... |
| b09b8lct53 | [trap, fracture, injury, unstable, damaged, br... |
| b00oqczavw | [trap, fracture, injury, damaged, toxic, missi... |

Figure 9: Top 10 Unique reviewed products

We focused on the 10 most-reviewed products, which are generally associated with higher satisfaction levels. Some highlights include:

- **Infant Optics DXR-8 Video Baby Monitor** led with a staggering **27,282 reviews** and an average rating of **4.20**, showcasing its popularity and customer trust.
- Products like the **Baby Banana Yellow Banana Infant Toothbrush** scored the highest in average ratings (**4.64**) among the top products, suggesting high customer satisfaction.

However, even for these top-performing products, safety-related terms such as **"trap," "fracture," "injury,"** and **"unstable"** were frequently observed in customer feedback. These recurring terms suggest potential minor concerns or design issues that could be proactively addressed.

```
Lowest Reviewed Products with Incidents and Safety Terms:
       parent_asin  product_avg_rating  \
45689  b004vl2vro            3.475957
24712  b004vl2vro            3.475957
35181  b01ad3l1jc            4.121472
31422  b000wjlkqm            4.434572
46697  b003am8cm8            3.500000
12955  b004vl2vro            3.475957
47951  b00e8kjync            3.649015
26779  b002a9iz0q            4.264286
19776  b001i463g2            4.736842
35021  b01ad3l1jc            4.121472

                                            title  sentiment_score  \
45689  Motorola MBP36 Remote Wireless Video Baby Moni...         -0.49830
24712  Motorola MBP36 Remote Wireless Video Baby Moni...         -0.49715
35181  Evenflo Exersaucer Triple Fun Active Learning ...         -0.49515
31422         Graco Doorway Bumper Jumper, Little Jungle         -0.49475
46697            OXO Tot Sprout High Chair, Green/Walnut         -0.49425
12955  Motorola MBP36 Remote Wireless Video Baby Moni...         -0.49330
47951                            Baby Brezza Formula Pro         -0.49280
26779  Storkcraft Tuscany 5-in-1 Convertible Crib (Es...         -0.49260
19776                       Joovy Room2 Portable Playard         -0.49220
35021  Evenflo Exersaucer Triple Fun Active Learning ...         -0.49195

                     incident_description_safety_terms
45689  [faulty, damaged, wound, injury, trauma, damag...
24712  [faulty, damaged, wound, injury, trauma, damag...
35181                                             [fire]
31422  [faulty, death, break, detached, defective, fa...
46697  [stuck, lead, flaw, caught, risk, defect, hazard]
12955  [faulty, damaged, wound, injury, trauma, damag...
47951  [faulty, missing, severe, blood, break, fuse, ...
26779                  [severe, rash, scratch, serious]
```

Figure 10: Low Reviewed Products

In contrast, the lowest-reviewed products presented more serious concerns, reflected in their negative sentiment scores and detailed safety terms. For example:

- The **Motorola MBP36 Remote Wireless Video Baby Monitor** had a negative sentiment score of **-0.49830**, with safety terms like **"faulty," "damaged," "trauma,"** and **"fire"** appearing repeatedly in incident descriptions.

- The **OXO Tot Sprout High Chair** also showed negative sentiment (**-0.49425**) and terms like **"stuck," "lead," "risk,"** and **"defect,"** indicating potential issues with usability and material safety.

These terms highlight critical areas of dissatisfaction and risk, which need immediate attention.

**Customer Feedback Trends**

- The **top-reviewed products** exhibit strong customer engagement and satisfaction, but the presence of terms like **"trap"** and **"unstable"** suggests areas for refinement.
- The **lowest-reviewed products** reveal dissatisfaction due to specific safety risks, with terms like **"fire," "trauma,"** and **"hazard"** underscoring the need for robust quality control.

**Safety Concerns**

- Terms such as **"fracture," "faulty,"** and **"risk"** point to recurring concerns across multiple products, indicating potential systemic issues in product design or manufacturing processes.
- Unique terms like **"overheating"** for the Infant Optics Baby Monitor and **"lead"** for the OXO Tot Sprout High Chair require targeted investigation.

### 5.1.8 Safety Word Extraction (Word Cloud)

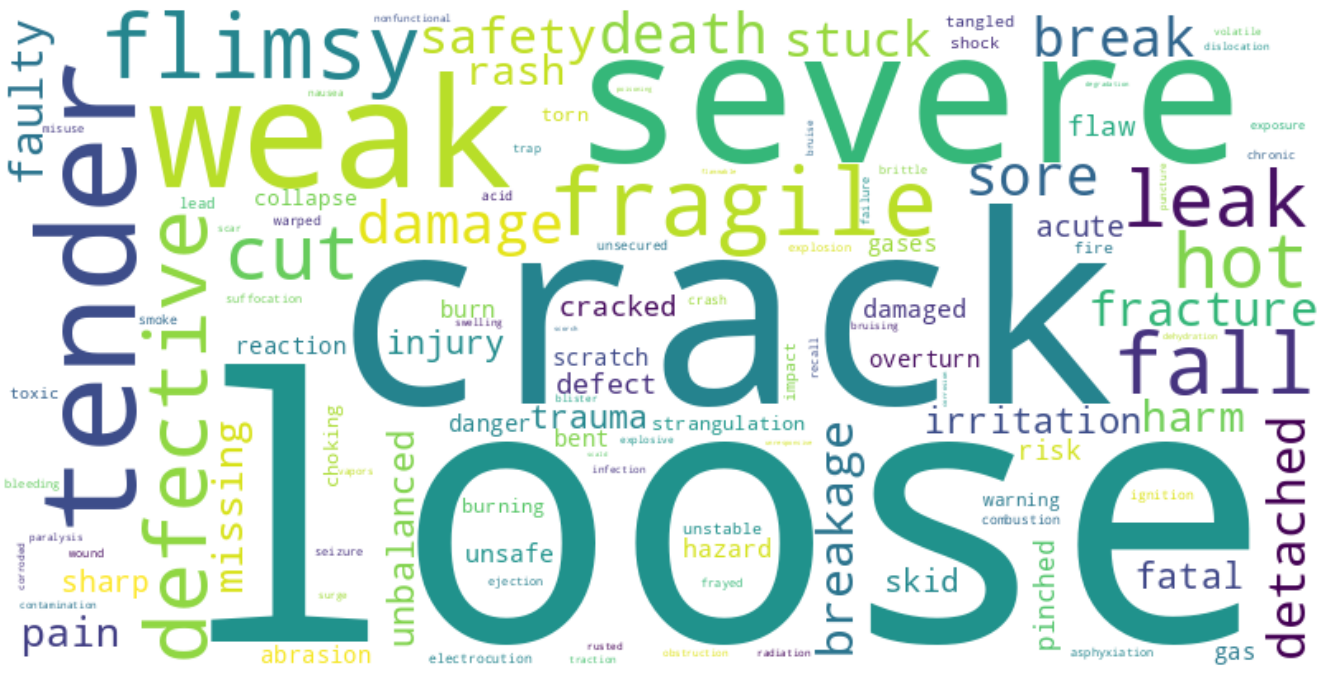



Figure 11: Word Cloud for Safety Word Analysis

The goal of this analysis was to extract and visualize the most frequently occurring safety-related terms from the dataset to identify critical patterns in user feedback and incident reports. This helps in proactively identifying product safety issues and improving risk management strategies. The word cloud visualization highlights the most frequently used terms, making it easier to identify the most common safety-related issues reported by users. The analysis was conducted on the dataset containing product reviews and incident reports. Key fields used for this task include:

- **cleaned_text_safety_terms**: A column containing extracted safety-related terms (e.g., "loose", "crack").
- **incident_safety_terms**: Additional descriptive terms linked to safety incidents.

To prepare the data for analysis:

- Terms were extracted from nested lists in the text fields.
- The extracted terms were flattened into a single list of words.
- The words were normalized to lowercase, and duplicates were retained to preserve their frequency.

**Observations:**

- The most frequent term, **"loose"**, indicates issues related to product assembly or parts that are not securely fastened.
- Terms like **"crack"**, **"fragile"**, and **"defective"** point to material and structural failures that may lead to safety incidents.
- High occurrence of terms like **"severe"** and **"missing"** suggests that these defects have caused significant problems or are critical issues reported by users.

The word cloud analysis successfully identified critical safety-related terms in the dataset. These insights will inform product design improvements and safety standards, enabling proactive mitigation of risks and enhanced user satisfaction.

## 5.2 Machine Learning Model Training, Evaluation, and Validation

The model training process began with a comprehensive data wrangling phase, during which irrelevant columns were removed and key features such as average ratings, review counts, sentiment scores, safety term density, and incident review match rates were selected. Initially, supervised learning approaches were explored, using product recalls as the target variable. A stratified sampling technique was applied, allocating 80% of the dataset for training and 20% for testing.

Several supervised models were evaluated, including Logistic Regression with regularization and higher thresholds, XGBoost, Decision Tree, K-Nearest Neighbors, and Gaussian Naive Bayes. These models were assessed using precision, recall, and F1 score metrics. However, the significant class imbalance led to poor class-wise accuracy, making the detection of unsafe products a challenging, iterative process.

To address these challenges, an unsupervised learning approach was adopted using the Isolation Forest algorithm, which excels at anomaly detection in high-dimensional datasets. This method was chosen due to the lack of sufficient labeled data, rendering supervised approaches less effective. Key features such as sentiment scores and safety term density were identified as crucial dimensions for representing the data.

During preprocessing, non-English reviews were filtered out, and products with fewer than 15 reviews were excluded to mitigate noise from bot-generated data. These steps reduced the dataset from 193,000 baby products to a more manageable and reliable subset of 70,000, which served as a better foundation for training.

The Isolation Forest model was configured with 100 trees and a threshold of 0.005, ensuring effective partitioning of the skewed dataset. This strategy identified 78 outliers, which were flagged as unsafe products due to their low average sentiment scores and high safety term densities.

This multifaceted approach underscored the limitations of supervised learning in the face of class imbalance and demonstrated the effectiveness of unsupervised methods for anomaly detection. The use of Isolation Forest highlighted its potential for addressing safety concerns in product reviews, providing a scalable and robust solution for detecting unsafe products.

## 5.3 Testing and Validation

| Model | Accuracy | Precision (Safe) | Precision (Unsafe) | Recall (Safe) | Recall (Unsafe) | F1-Score (Safe) | F1-Score (Unsafe) |
|---|---|---|---|---|---|---|---|
| **Logistic regression** | 98% | 1.00 | 0.00 | 0.98 | 0.14 | 0.99 | 0.00 |
| **Decision Tree** | 91% | 1.00 | 0.00 | 0.91 | 0.14 | 0.95 | 0.00 |
| **KNN** | 100% | 1.00 | 0.00 | 1.00 | 0.00 | 1.00 | 0.00 |
| **Naive Bayes** | 100% | 1.00 | 0.00 | 1.00 | 0.00 | 1.00 | 0.00 |
| **XGBoost** | 100% | 1.00 | 0.00 | 1.00 | 0.00 | 1.00 | 0.00 |

Table 1: Model Performances

During the testing and evaluation stage, the goal was to assess how different machine learning models performed in predicting product safety, particularly their ability to identify unsafe products. While Logistic Regression achieved a high overall accuracy of 98%, it excelled in identifying safe products but struggled with detecting unsafe ones due to a low recall for the minority class. Decision Tree and K-Nearest Neighbors (KNN) models also showed high precision and recall for the safe class but failed to identify any unsafe products, resulting in an F1 score of 0 for the minority class. Similarly, Naive Bayes achieved 100% accuracy for the safe class but had a recall of 0% for the unsafe class. Even XGBoost failed to improve performance, highlighting the consistent issue of class imbalance and the need for alternative approaches, such as the Isolation Forest algorithm.

To address these challenges, the Isolation Forest model was tested for its ability to efficiently identify unsafe products. The model succeeded in flagging outliers, which were characterized by low sentiment scores and high safety term densities. These outliers were then compared against unsafe products identified in recall and incident datasets. Despite challenges, such as inconsistencies in user-reported data, the Isolation Forest model demonstrated an accuracy of approximately 0.71, measured using the silhouette score.

The validation process also revealed areas for improvement, such as achieving 70% accuracy in negative predictive values and incorporating time-series data for real-time anomaly detection. These findings demonstrate the Isolation Forest model's effectiveness in identifying potentially unsafe products while highlighting areas for refinement to expand its applicability further.

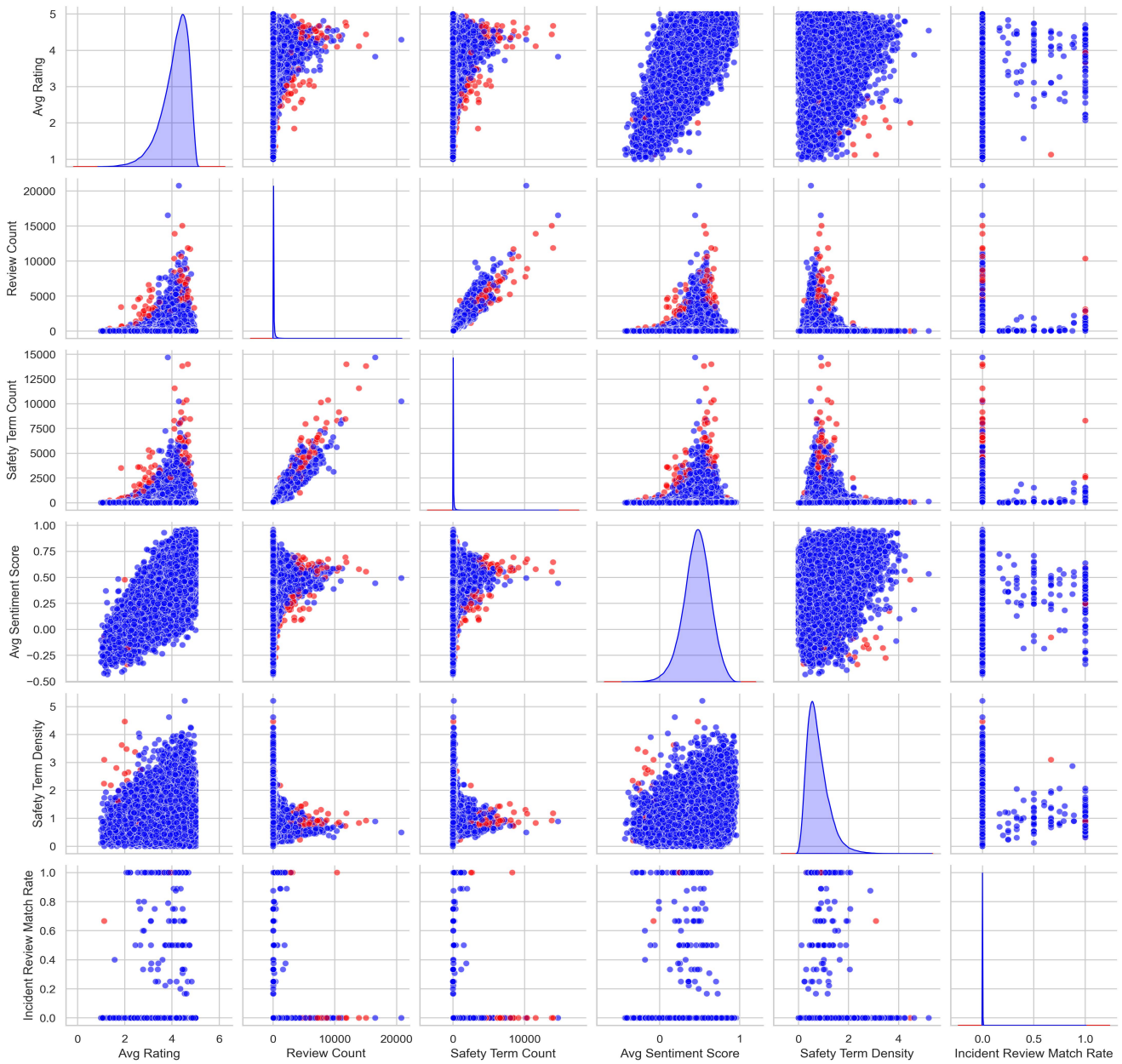



Figure 12: Scatter Plot Matrix for Isolation Forest

# Section 6: Findings

## 6.1 Model Performance Comparison

Among all the machine learning models tested, Logistic Regression and Decision Tree models showed balanced performance in identifying potentially unsafe products. However, their recall for the positive class (unsafe products) was consistently low across all models, indicating difficulty in correctly identifying true unsafe cases.

## 6.2 Isolation Forest Insights

The Isolation Forest model proved effective in detecting potential outliers in the dataset. It highlighted products with unusual patterns, such as high safety term density and low sentiment scores, which could indicate latent safety risks not immediately associated with incidents or recalls.

## 6.3 Key Features Impacting Predictions

- Features like avg_rating, review_count, and incident_review_match_rate significantly influenced model predictions.
- High safety term density often correlated with flagged unsafe predictions, showcasing its importance as a signal for potential product safety issues.

## 6.4 Outlier Clusters

Outliers detected by the Isolation Forest model clustered into two distinct groups:

- Products with overwhelmingly low ratings and sentiment scores, reflecting negative user feedback.
- Products with unusually high safety term counts or densities, which could indicate excessive user reporting of safety concerns.

## 6.5 Correlation with Actual Incidents

The models identified a subset of products that were involved in recalls or incidents. However, some flagged products did not have prior incident records, indicating the potential of predictive models in identifying risks proactively.

## 6.6 Challenges in Predicting Unsafe Products

- Despite high accuracy in most models, the recall for unsafe products remained a challenge. This highlights the need for additional data or improved feature engineering to enhance the detection of rare unsafe cases.
- Imbalanced data distribution (fewer unsafe cases) affected the models' ability to generalize well for the positive class.

## 6.7 General Observations

- Popular products with high review counts are not always safe, as some flagged products with high reviews had concerning safety indicators.
- Products with low ratings but high safety term density might represent consumer dissatisfaction stemming from safety-related concerns.

These findings underscore the importance of leveraging multiple models and features to enhance the identification of potentially unsafe products while addressing data imbalance and improving recall performance.

# Section 7: Summary

The "Proactive Identification of Product Safety Issues" project was conducted to analyze product safety through a combination of data-driven methods, including sentiment analysis, machine learning, and keyword extraction. Leveraging multiple datasets such as Amazon Reviews, product metadata, and Consumer Product Safety Commission (CPSC) incident and recall reports, we aimed to identify patterns that indicate potential safety risks associated with consumer products.

## 7.1 Key Discoveries

### 7.1.1 Sentiment and Review Patterns

- Sentiment analysis of customer reviews highlighted a significant correlation between negative sentiment scores and safety-related terms such as *"fracture," "injury,"* and *"damaged."*
- Products with higher ratings and review counts generally demonstrated positive sentiment but still contained recurring safety terms indicating minor risks.

### 7.1.2 Top and Lowest Reviewed Products

- The analysis of the top-reviewed products revealed customer satisfaction but also showed recurring mentions of safety concerns such as "overheating" and "unstable."
- The lowest-reviewed products exhibited more severe safety issues, such as "fire hazards," "lead contamination," and "trauma risks," further supported by significantly negative sentiment scores.

### 7.1.3 Safety Concerns and Risk Trends

- Frequent safety terms across products, such as "trap," "fracture," and "unstable," indicated systemic risks in product design or manufacturing.
- Analysis of incident and recall data validated the connection between recurring terms in reviews and real-world safety risks.

### 7.1.4 Effectiveness of the Model

The hybrid approach combining sentiment analysis, safety term density thresholds, and use of Heuristic model for classification successfully identified products with potential safety concerns, proving the feasibility of using customer feedback as a predictive indicator for safety issues.

### 7.1.5 Unsupervised Learning was more effective than Supervised Learning:

Regression models such as Logistic Regression, XGBoost, KNN, Naïve Bayes, Decision Tree, and Random Forest did not produce the desired results due to the dataset being unlabeled and highly imbalanced. However, the unsupervised learning approach, specifically Isolation Forest, proved effective in identifying outliers and anomalous patterns, enabling the detection of products or reviews linked to potential safety risks.

## 7.2 What We Proved

### 7.2.1 Customer Feedback as a Risk Indicator

We demonstrated that sentiment analysis and safety term extraction from customer reviews can serve as reliable early indicators of potential safety risks.

### 7.2.2 Proactive Safety Identification

By integrating review data with incident and recall datasets, our approach allows manufacturers and regulators to proactively identify unsafe products before widespread issues arise.

## 7.3 What We Disproved

### 7.3.1 High Ratings Equal Safety

Our analysis disproved the notion that high customer ratings always correlate with product safety. Even highly rated products displayed safety-related terms that require investigation.

## 7.4 Overall Results

The project successfully met its objectives by:

- Identifying patterns and terms indicative of safety concerns in product reviews.
- Proving that sentiment analysis and keyword extraction techniques can assist in identifying unsafe products.
- Highlighting the gap between customer satisfaction and safety, providing actionable insights for manufacturers to improve product quality.
- Supervised Machine learning algorithms like Logistic Regression, XG Boost, KNN, Decision tree was not giving accurate results due to high imbalanced data and the dataset not being labelled. Hence Heuristic Model approach was chosen to label the output with the help of features.
- The isolation forest was successful in detecting the outliers within the given dimensionality.

Our findings provide a foundation for scalable, automated systems to monitor consumer feedback and enhance product safety standards across industries.

## 7.5 Future Implications

This work has laid the groundwork for future enhancements, such as:

- Developing real-time monitoring tools for analyzing customer reviews.
- Expanding the methodology to incorporate more comprehensive datasets and advanced machine learning models.
- Collaborating with regulatory bodies to implement proactive safety frameworks based on our findings.

In summary, this project has not only proven the feasibility of leveraging customer reviews to identify safety risks but also provided a practical framework for improving consumer safety and trust.

# Section 8: Future Work

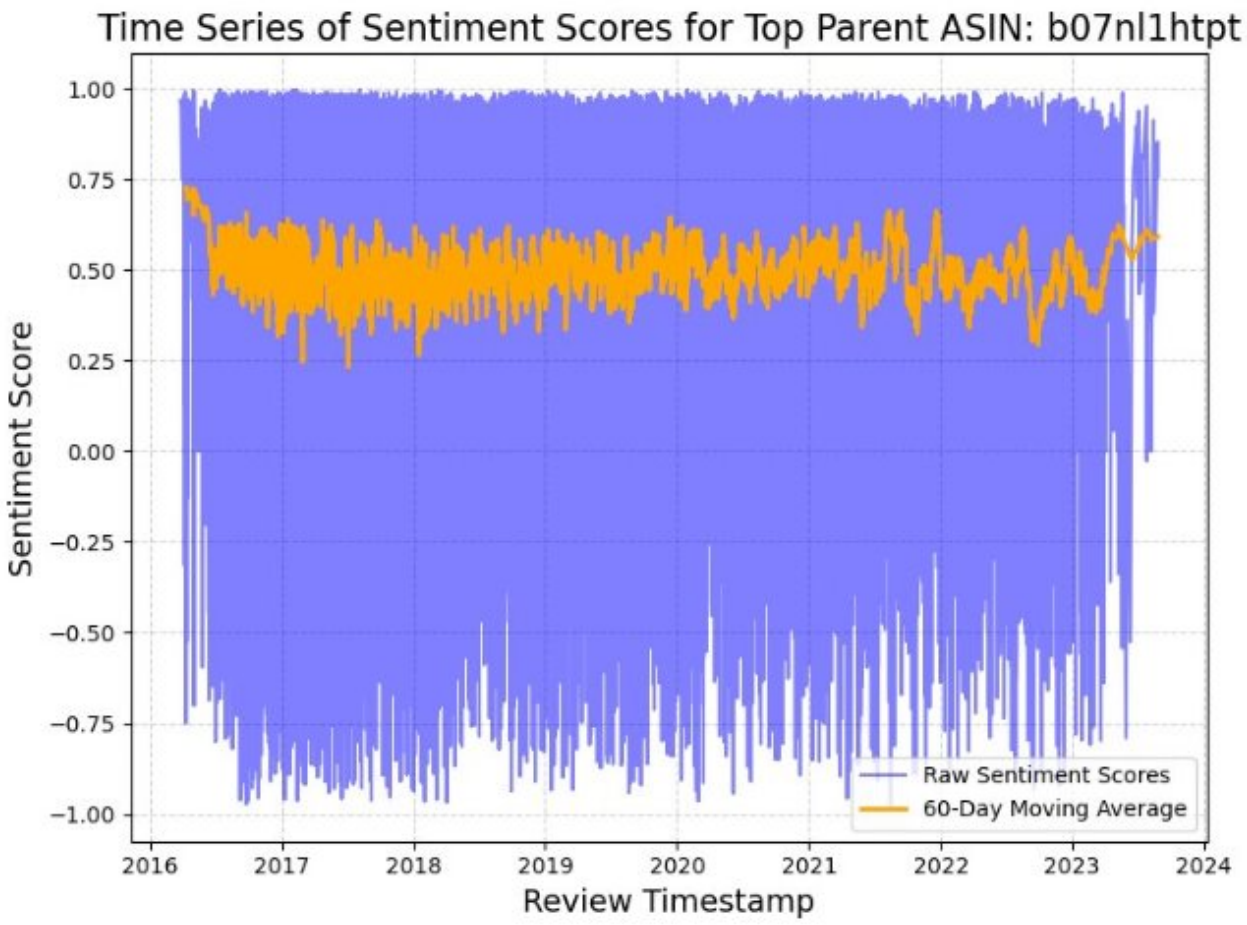



Figure 13: Time Series Analysis of Sentiment Scores

## 8.1 Supervised Machine Learning

As part of the team's initial study, multiple supervised algorithms were attempted to include Logistic Regression, XGBoost and Naïve Bayes. Recall scores ranged from 0% to 14% indicated an extremely imbalanced dataset. But this was the limitation of the dataset itself, not the methodology.

The labelled data was lacking due to human-input values from the CPSC Incident data as well as at free-text inputs from the Recalled products dataset, both combined creating poorly viable joins. The team believes that a supervised machine learning model is still a viable methodology, while still utilizing a combined sentiment analysis and a "dirty word" search approach towards the problemset.

## 8.2 Time Series Analysis

As an initial study and novel approach, the methodology in this study is limited. The total processing functions outside of the bounds of time, while still spanning a 13-year period. Product recall timelines vary considerably from initial identification to rehabilitation, but a basis with time can truly provide much deeper analysis of the underlying manufacturer timelines.

Consider a 60-day moving average of the sentiment of a product within its reviews along with a moving average of the "Safety Terms" (Figure). With a drop in the sentiment and increase of "Safety Terms" within a certain number of standard deviations, this could be an indicator of a necessary human review of the product and can alert manufacturers to potential defects.

## 8.3 Semantic Similarity

Dirty word search within the team's study has certain limitations, given a purely direct-search approach. Consider the word "cut" extracted from the word "cute" or the word "bloody" from the British "bloody fantastic." Semantic Similarity, measuring the relationship of two pieces of text despite their phrasing differences, allows to extract the more nuanced approach of natural language.

Within this study, the team attempted utilizing the spaCy model for semantic similarity word extraction, but it proved too computationally intensive given the dataset size. However, the team believes utilizing this approach would be more powerful in controlling the number of false positives within the current study.

# Appendix

## Appendix A: Domain Background

### Introduction

Product safety is crucial to ensuring that products, particularly baby products, meet established safety standards for their intended use. Failure to comply with these standards can lead to serious consequences, including injuries such as choking, strangulation, and broken bones. Manufacturers play a critical role in prioritizing safety throughout the production process to minimize potential risks. For baby products, this is even more crucial due to the vulnerability of infants, making it vital to identify common safety risks and propose predictive solutions to mitigate them. [2]

### Background on Product Safety and Recalls

Product safety regulations are designed to protect consumers by ensuring that manufacturers comply with established safety standards. When products fail to meet these standards or present unexpected risks, recalls are initiated to prevent harm. The **U.S. Consumer Product Safety Commission (CPSC)** is an independent regulatory body that monitors and enforces safety standards across more than 15,000 types of consumer goods. Their mission is to reduce injury or death by managing recalls and investigating incidents. However, recalls can impose significant burdens on businesses, leading to financial losses and damage to a company's reputation, while consumers rely on recalls to prevent potential injuries or property damage. [3] [4]

### Literature Review on Product Safety

1. **Auto-Detection of Safety Issues in Baby Products (2018):** This study explores how machine learning and natural language processing (NLP) can help identify safety issues in baby products through customer reviews. By leveraging a dataset from Amazon reviews, **SaferProducts.gov** complaints, and CPSC recall data, the research demonstrates that early identification of safety concerns can lead to faster recalls, ultimately preventing injuries or deaths. The study highlights a significant gap between consumer-reported issues and official recalls, suggesting the need for automated solutions to detect potential risks from online reviews. However, the authors note the challenges of false positives and limited training data. [5]

2. **What's Wrong with this Product? Detection of Product Safety Issues (2023)**: This research focuses on detecting product safety issues through data shared by consumers online. By combining data from the **European Union Safety Gate (EUSG)** and product reviews, the study created a framework that regulatory bodies can use to filter unsafe products. Logistic Regression emerged as the best-performing model, achieving high precision and recall in identifying safety issues, underscoring the importance of leveraging large datasets for early detection of potential hazards. [6]

## Common Safety Issues across different Products

1. **Choking Hazards**: Choking hazards are commonly associated with products containing small parts, especially children's toys, but can also occur in items not intended for children. Products that break into smaller pieces or contain detachable components can pose serious risks to young children. Regulatory bodies such as the CPSC enforce strict safety standards to minimize these hazards, especially for products marketed to children under the age of three. [7]
2. **Toxic Substances:** The use of toxic chemicals in products, including phthalates in plastics, lead in paints, and harmful chemicals in household goods, can lead to long-term health issues like developmental delays, cancer, and respiratory problems. In recent years, regulators have increased restrictions on toxic substances, especially in products such as cosmetics, toys, and furniture. Manufacturers are required to follow stringent guidelines to limit exposure to these hazardous substances. [8] [9] [26] [27]
3. **Product Malfunctions**: Malfunctions in products across various categories—whether they be electronics, automobiles, or household appliances—can lead to severe injury or even death. Faulty electronics can cause electrical fires, while defective vehicles can lead to accidents. Regulators emphasize the importance of rigorous safety testing and proper quality control processes to avoid malfunctions that put consumers at risk. [10] [28]
4. **Sharp Edges or Loose Components:** Many products, including furniture, tools, and household items, can pose safety risks if they contain sharp edges or poorly secured components. These defects can lead to lacerations, puncture wounds, or other injuries, especially in products frequently used by children or in high-contact settings. Manufacturers are required to conduct safety testing to ensure that all components are securely fastened and that there are no sharp or hazardous parts. [11][29]
5. **Suffocation Risks**: Suffocation and strangulation risks can arise from a variety of products, including plastic bags, window blinds, and furniture. In children's products, soft bedding or toys can lead to suffocation, while window blind cords have been linked to strangulation incidents. Manufacturers are required to design products that eliminate or significantly reduce these risks, and warning labels must clearly communicate potential dangers to consumers. [11][30]

## Key Recall Triggers

Recalls in the industry are often triggered by a range of safety hazards:

1. **Non-Compliance with Safety Standards**: Products that fail to meet regulatory safety standards, such as fire resistance, electrical safety, or the presence of hazardous materials, are often recalled. This applies across industries, from electronics to furniture, to ensure consumer protection. [12]
2. **Consumer-Reported Incidents**: Safety recalls are frequently triggered by reports from consumers who experience product malfunctions or hazards. Complaints can range from small defects to significant risks, leading to broader investigations and eventual recalls if patterns of risk are identified. [13]
3. **Product Defects**: Defects in design or manufacturing, such as unstable components, overheating electronics, or fragile materials, can lead to widespread recalls. These defects can result in injury, property damage, or even fatalities, making swift action necessary. [14]
4. **Failure to Pass Safety Testing**: Products that fail to meet regulatory or internal safety testing standards are recalled before they reach mass distribution. This preemptive measure prevents harm from faulty products and ensures that only safe items are sold to consumers. [15]

# Appendix B: Glossary

| Term | Definition |
|---|---|
| **Product Recall** | It is a request to return to the maker, a batch or an entire production run of a product, usually over safety concerns or design defects or labelling errors [16] |
| **Consumer Product Safety Commission (CPSC)** | It is a federal agency that works to protect the public from risks of injury or death caused by consumer products through safety standards, education, and enforcement. [17] |
| **Natural Language Processing (NLP)** | It is a machine learning technology that gives computers the ability to interpret, manipulate, and comprehend human language. [18] |
| **Sentiment Analysis** | It is the process of analyzing digital text to determine if the emotional tone of the message is positive, negative, or neutral. [19] |
| **Bigrams/Trigrams** | A bigram is a sequence of two consecutive words in a text, while a trigram is a sequence of three consecutive words, both used in natural language processing to analyze word patterns and relationships in text data. [20] |
| **Random Forest Classification** | It is a machine learning algorithm that builds multiple decision trees and outputs the most common class for reliable predictions. [21] |
| **Phthalates** | They are chemical compounds used to make plastics more durable and act as solvents and stabilizers in cosmetics like nail polishes, hair sprays, and shampoos. [22] |
| **ASIN (Amazon Standard Identification Number)** | a unique ten-digit alphanumeric code used to identify products on Amazon, required for selling items except for books, which use ISBN instead. [23] |
| **Stopword Removal** | A process in NLP where common words (such as "the," "is," "at") are removed from a dataset to improve the quality of text analysis. [24] |
| **Regex (Regular Expression)** | It is a pattern of characters used to search, match, and manipulate text in strings. [25] |
| **Tokenization** | Tokenization in Natural Language Processing (NLP) is the process of breaking down a sentence or phrase into smaller units called tokens. [34] |
| **Stop Words** | Stop words are commonly used words in a language, such as "a," "the," "is," and "are." [35] |
| **Named Entity Recognition (NER)** | Named Entity Recognition (NER) extracts key entities like names and dates from text and organizes them into structured data, making large volumes of text easier to analyze. [36] |
| **TF – IDF** | TF-IDF (Term Frequency - Inverse Document Frequency) is a popular technique that measures the importance of a word in a document relative to a collection of documents. [37] |
| **RoBERTa** | RoBERTa is an AI model developed by Meta's Research team, named after the Italian name Roberta, though it's far from human. [38] |
| **Semantic Similarity** | Semantic similarity refers to the measure of how similar the meaning of two words or phrases is, even if they are phrased differently. [39] |
| **Fuzzy String Matching** | Fuzzy string matching, or fuzzy matching, is a technique used to identify strings that partially match a given string, rather than requiring an exact match. [39] |
| **WordNet** | WordNet is a lexical database that groups words into sets of synonyms and shows their relationships, helping analyze meanings and word connections. [40] |
| **Bag of Words** | Bag of Words (BoW) is a text representation model that converts text into a collection of words without considering grammar or word order, focusing solely on word frequency. [41] |

| | |
|---|---|
| **Logistic Regression** | A statistical method used for binary classification to predict the probability of an event, such as success or failure, based on independent variables. [47] |
| **Gradient Boosting** | A machine learning technique that combines the predictions of several weak models to create a strong model, typically used with decision trees. [48] |
| **XGBoost** | A fast and efficient implementation of gradient boosting designed for structured datasets and high performance. [49] |
| **LightGBM** | A gradient boosting framework optimized for large datasets, using less memory and providing fast computation. [50] |
| **CatBoost** | A gradient boosting algorithm optimized for categorical data, with minimal data preprocessing required. [51] |
| **F1-Score** | A metric that balances precision and recall, providing a harmonic mean of the two. It is particularly useful for evaluating models with imbalanced datasets. [52] |
| **ROC-AUC** | A performance metric for binary classifiers, which measures the trade-off between the true positive rate and false positive rate. The higher the AUC, the better the model at distinguishing between classes. [52] |
| **Confusion Matrix** | A confusion matrix (or error matrix) is a visualization tool for classifier results. It is a table that compares the number of actual instances of each class with the number of instances predicted by the model. [54] |

Table 2: Glossary Table

# Appendix C: GitHub Repository

## Repository Overview

- **Project Description**: This repository contains the code and data used in a study aimed at early detection of retail products not meeting safety standards using crowdsourcing methods through Amazon review data. The project utilizes Natural Language Processing (NLP) techniques to analyze consumer reviews and predict potential safety issues.
- **Objective**: To develop a methodology that leverages review data to predict safety recalls before products cause harm, improving consumer safety and compliance with CPSC standards.

## GitHub Repository Link

https://github.com/saiphanichandrams/DAEN690-safetyDance

**THE SAFETY DANCE**
########## Amazon Review Safety Analysis Pipeline ##########
########## By Jonathan King, Aakash Boenal,Utkarsh Ganjihal, SaiPhani Chandra Vuppala, DAEN690 - Fall 2024 - George Mason University ##########
This program is a large-scale Amazon review analysis tool that downloads, preprocesses, and analyzes review data to identify potentially unsafe products. The tool employs both heuristic analysis and machine learning (Isolation Forest) to flag products based on various features, such as ratings, sentiment, and safety-related terms.

Developed as a Data Analytics Capstone Project (DAEN690 - Fall 2024) by Jonathan King, Aakash Boenal, Utkarsh Ganjihal, SaiPhani Chandra Vuppala, George Mason University, this program is optimized for large datasets and advanced analysis.

Partnered with NIRA, Inc for real-world analysis.

########## FEATURES
########## Category-based Review Download:

- Downloads Amazon review data by category from the McAuley Lab dataset. Preprocessing:
- Cleans, normalizes, and filters non-English reviews using parallel processing. Sentiment Analysis:
- Applies VADER to classify review sentiment (positive, neutral, negative). Safety Term Extraction:
- Matches predefined safety-related terms and calculates overlap with incident descriptions. Heuristic Analysis:
- Flags potentially unsafe products based on quantile thresholds:
- 25th percentile: Low ratings and sentiment scores.
- 75th percentile: High safety term density and incident-review match rates. Isolation Forest:
- Uses machine learning to detect outliers based on safety-related features.

########## SETUP
########## Required Packages

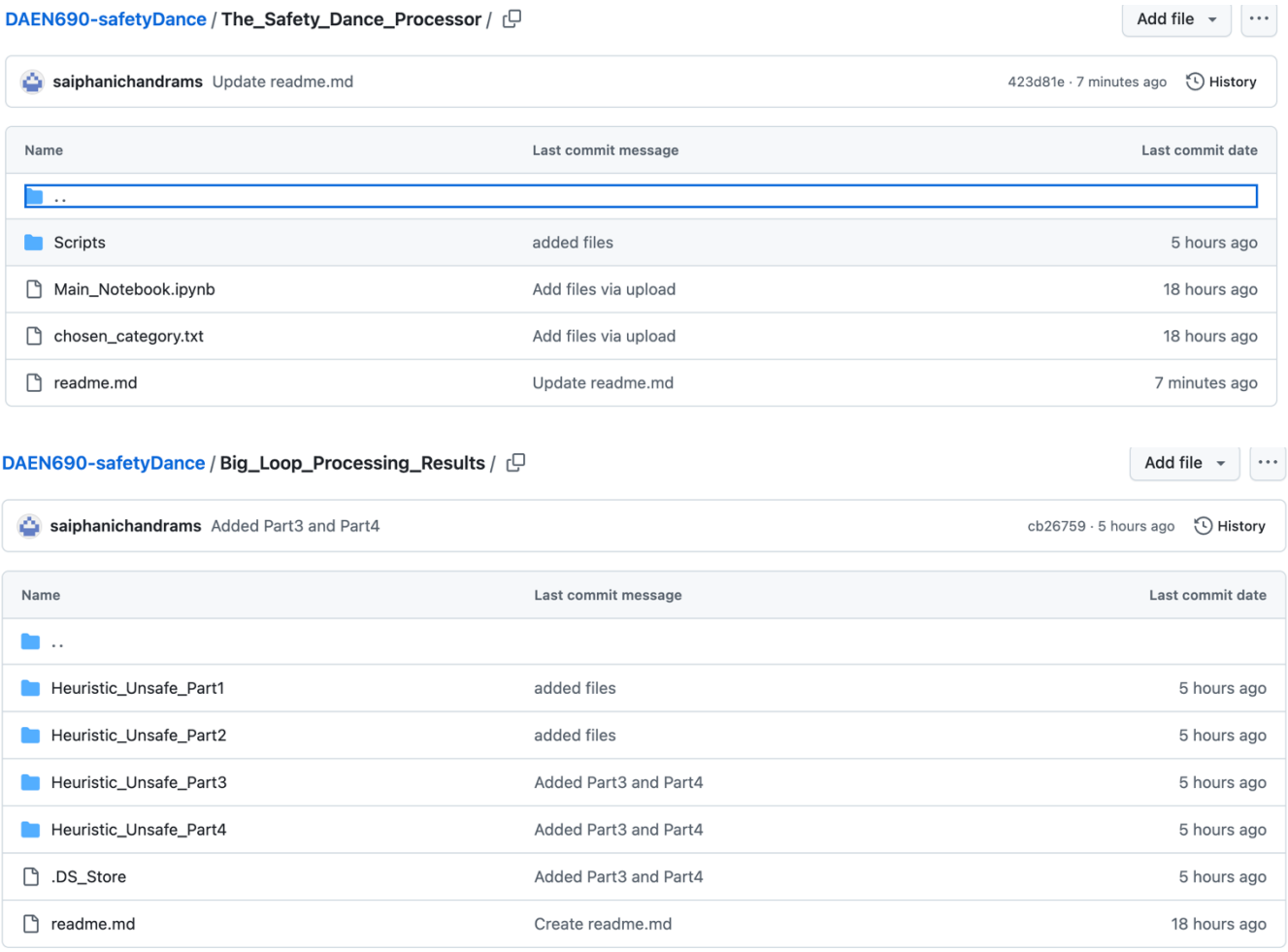
- Installation handled in the first cell of the main notebook
- Or via requirements.txt (for Conda/virtual environments). Core dependencies include:
- scikit-learn, pandas, numpy, tqdm, spacy, nltk, transformers, datasets Data Sources McAuley Lab Amazon Reviews Dataset:
- Installed via pip install datasets from Hugging Face. Incidents Data:
- Static file: Incidents_Extracted_model_number_cleaned2.csv.
- Includes manually extracted ASINs, Model IDs, UPCs, and TCINs. Recalls Data:
- Static file: recall_cleaned_manual.csv.
- Includes attributes manually extracted from recall reports.

** NOTE**: Place all static files in the Scripts directory.

########## PIPELINE OVERVIEW
##########

- Main Notebook: Main_Notebook.ipynb
- This notebook runs the entire pipeline by invoking the following scripts in sequence:

## GitHub Repository Contents



DAEN690-safetyDance / The_Safety_Dance_Processor /

Add file

saiphanichandrams Update readme.md — 423d81e · 7 minutes ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| Scripts | added files | 5 hours ago |
| Main_Notebook.ipynb | Add files via upload | 18 hours ago |
| chosen_category.txt | Add files via upload | 18 hours ago |
| readme.md | Update readme.md | 7 minutes ago |

DAEN690-safetyDance / Big_Loop_Processing_Results /

Add file

saiphanichandrams Added Part3 and Part4 — cb26759 · 5 hours ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| Heuristic_Unsafe_Part1 | added files | 5 hours ago |
| Heuristic_Unsafe_Part2 | added files | 5 hours ago |
| Heuristic_Unsafe_Part3 | Added Part3 and Part4 | 5 hours ago |
| Heuristic_Unsafe_Part4 | Added Part3 and Part4 | 5 hours ago |
| .DS_Store | Added Part3 and Part4 | 5 hours ago |
| readme.md | Create readme.md | 18 hours ago |

DAEN690-safetyDance / Big_Loop_Processing_Results / **Heuristic_Unsafe_Part1** /

Add file

**saiphanichandrams** added files — 9ec9bdb · 5 hours ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| .DS_Store | Added files | 6 hours ago |
| All Beauty_recalls.csv | added files | 5 hours ago |
| Cell Phones and Accessories_recalls.csv | added files | 5 hours ago |
| Incidents_Extracted_model_number_cleaned2.csv | added files | 5 hours ago |
| Industrial and Scientific_recalls.csv | added files | 5 hours ago |
| Toys and Games_recalls.csv | added files | 5 hours ago |
| heuristic_unsafe_products_All_Beauty.csv | added files | 5 hours ago |
| heuristic_unsafe_products_Cell_Phones_and_Accessorie... | added files | 5 hours ago |
| heuristic_unsafe_products_Industrial_and_Scientific.csv | added files | 5 hours ago |
| heuristic_unsafe_products_Toys_and_Games.csv | added files | 5 hours ago |
| incident_involved_recordsAll_Beauty.csv | added files | 5 hours ago |
| incident_involved_recordsCell_Phones_and_Accessories.... | added files | 5 hours ago |
| incident_involved_recordsGift_Cards.csv | added files | 5 hours ago |
| incident_involved_recordsIndustrial_and_Scientific.csv | added files | 5 hours ago |
| incident_involved_recordsToys_and_Games.csv | added files | 5 hours ago |

DAEN690-safetyDance / Big_Loop_Processing_Results / **Heuristic_Unsafe_Part2** /

Add file

**saiphanichandrams** added files — 9ec9bdb · 5 hours ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| Musical Instruments_recalls.csv | added files | 5 hours ago |
| electronics_recalls.csv | added files | 5 hours ago |
| heuristic_unsafe_products_Electronics.csv | added files | 5 hours ago |
| heuristic_unsafe_products_Musical_Instruments.csv | added files | 5 hours ago |
| incident_involved_recordsMusical_Instruments.csv | added files | 5 hours ago |
| incident_involved_records_Electronics.csv | added files | 5 hours ago |

DAEN690-safetyDance / Big_Loop_Processing_Results / **Heuristic_Unsafe_Part3** /

Add file ···

**saiphanichandrams** Added Part3 and Part4 — cb26759 · 5 hours ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| Baby Products_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Grocery and Gourmet Food_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Health_and_Household_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Office Products_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| arts_crafts_sewing_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Art_Crafts_Sewing.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Baby_Products.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Grocery_and_Gourmet_Food.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Health_and_Household.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Office_Products.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Sports_and_Outdoors.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsArts_Crafts_and_Sewing.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsBaby_Products.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsGrocery_and_Gourmet_Food.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsHealth_and_Household.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsOffice_Products.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsSports_and_Outdoors.csv | Added Part3 and Part4 | 5 hours ago |

DAEN690-safetyDance / Big_Loop_Processing_Results / **Heuristic_Unsafe_Part4** /

Add file ···

**saiphanichandrams** Added Part3 and Part4 — cb26759 · 5 hours ago — History

| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| Incidents_Extracted_model_number_cleaned2.csv | Added Part3 and Part4 | 5 hours ago |
| Pet Supplies_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Sports and Outdoors_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Subscription Boxes_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| Tools and Home Improvement_recalls.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Pet_Supplies.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Subscription_Boxes.csv | Added Part3 and Part4 | 5 hours ago |
| heuristic_unsafe_products_Tools_and_Home_Improvement.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsPet_Supplies.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsSubscription_Boxes.csv | Added Part3 and Part4 | 5 hours ago |
| incident_involved_recordsTools_and_Home_Improvement.csv | Added Part3 and Part4 | 5 hours ago |
| isolationforest_unsafe_products_Tools_and_Home_Improvement.csv | Added Part3 and Part4 | 5 hours ago |

# Appendix D: Risks

## Sprint 1 Risks

| Risk | Description | Probability | Impact | Mitigation |
|---|---|---|---|---|
| **Unable to link datasets** | Unable to merge the Incidents dataset and Recall dataset with review dataset | Low | Medium | Workarounds complicates the dataset |
| **NLP Education** | Some Team members have not taken NLP Courses | High | Low | Team members currently educating themselves on NLP |
| **Code base** | There is limited previous research and code base and databases with NLP and Consumer safety | Medium | Medium | Workflow and Code base developed in-house |
| **Messy dataset** | Model ID/Name is all user-input values, resulting in mismatching of values, incorrect values, etc. | High | Medium | Stakeholder notes no alternatives. Manual data cleaning |

Table 3: Sprint 1 Risks

As part of the team, we took an active role in identifying and mitigating the risks associated with this sprint. Here's how we approached the risk management and what we learned along the way:

1. **Risk Identification Process**

   Right from the start, we flagged the challenge of linking datasets as a significant concern. Merging the Incidents, Recall, and review datasets could lead to complex integration issues, so we discussed possible roadblocks and identified workarounds in case the data structures didn't align perfectly. The NLP education gap was something we recognized within the team early on. Since not everyone had formal training in NLP, we knew this could slow down progress, but we were proactive in encouraging self-education to close this gap quickly. The code base risk was another key factor we identified. We were aware that NLP applications in consumer safety were not widely researched, and we had limited pre-existing material to draw from. This prompted us to focus on building an in-house workflow that we could rely on.

2. **What We Got Right**

   Our preemptive focus on the dataset linking issue paid off. By anticipating problems, we managed to avoid any major delays by having workarounds in place. We discussed ways to manage mismatches in data and format early on. We were also successful in identifying the educational gap in NLP. With several team members committed to upskilling themselves, we quickly bridged the knowledge gap and distributed tasks based on everyone's growing expertise.

3. **What We Could Have Done Differently**

   One area we could have improved on was leveraging external resources. In retrospect, we might have sped up development if we had explored more existing NLP tools, open-source projects, or collaborated with NLP experts. Building an in-house workflow from scratch was valuable, but it did consume more time than we initially anticipated. We also could have been quicker in tackling the code base issue. While we anticipated

the lack of available research, we didn't fully account for how long it would take to develop a custom workflow for our needs.

**4. Accuracy in Identifying Risks**

Our initial assessment of risks was quite accurate. The linking of datasets and NLP education were exactly where the challenges arose. Although we handled them well, the code base issue turned out to be more time-intensive than we had expected. That said, by planning for these risks, we mitigated any major setbacks. We were able to adapt our approach as we progressed, particularly by developing resources in-house to overcome the limitations of existing research.

**5. Unanticipated Risks**

Although we anticipated the risk levels accurately, we encountered an issue with the Incidents and Recall datasets that we hadn't fully anticipated. Specifically, extracting model numbers from the text data became a significant challenge. Despite applying multiple regex expressions and experimenting with various NLP techniques, we were unable to devise a reliable pipeline for extracting the model numbers automatically. This ultimately required manual intervention to resolve the issue. The need for human involvement in this aspect of the project introduced an unexpected delay and underscored the complexity of working with inconsistent text data in real-world datasets.

**6. Overall Reflection**

As a team, we managed risks well and anticipated the primary challenges we would face. By focusing on upskilling and preparing for dataset integration issues, we ensured smooth progress. In future projects, we'd benefit from engaging with more external tools and resources to avoid spending too much time building in-house solutions from scratch. The willingness to self-educate and collaborate was a strength, helping us stay adaptable and focused even as we faced technical roadblocks. This project taught us how valuable it is to get ahead of potential risks while staying flexible enough to adjust as needed.

## Sprint 2 Risks

| Risk | Description | Probability | Impact | Mitigation |
|---|---|---|---|---|
| **Model Accuracy** | With manual dataset cleaning, manual model ID inputs and merging might still not make enough incidents for viable model training | Medium | Medium | Workflow would still exist, potential MAP score of ~.40 without incident data |
| **NLP Education** | Some Team members have not taken NLP Courses | High | Low | Team members currently educating themselves on NLP |
| **Increasing complexity of variables** | Adding in Time series along with Sentiment Analysis and Incident/Recalls creates higher levels of complexity programmatically | Medium | High | Research |

Table 4: Sprint 2 Risks

During Sprint 2, the team proactively identified and addressed key risks to ensure the project stayed on track. Here's an overview of the process and the lessons we learned:

**1. Risk Identification and Mitigation Planning**

The team identified risks based on the complexity of tasks, particularly around model accuracy and the introduction of new variables like time series and sentiment analysis. Manual data entry may limit quality of model training, especially due to the scarcity/variable matching accuracy of product incident data. This was correctly addressed by setting realistic expectations for model performance, with a projected MAP score of around 0.40. This is because an average of <1% of all products within the incidents dataset were merging with viable Amazon products, which limits model training for weights and balances. However, the realistic approach is that the CPSC and Amazon reviews are indeed real-world data and represents real-world accuracy. Given that products are indeed being recalled/involved in incidents at a similar rate, there may be no action needed to account for this discrepancy.

The NLP Education risk was flagged due to the gap in NLP knowledge across some team members. The high probability but low impact of this risk allowed us to focus on continuous education, without impacting deliverables. We also identified that increasing the complexity of variables, especially with the addition of time series and sentiment analysis, would pose a risk of higher programmatic complexity. We mitigated this by dedicating time to research and refining our approach to manage these complexities effectively.

**2. What the Team Got Right**

The team was highly effective in foreseeing the Model Accuracy challenge. By addressing this risk early, we were able to manage expectations and avoid significant rework down the line.

The proactive decision to engage in NLP Education for team members helped mitigate potential bottlenecks. The strategy of learning while continuing project tasks allowed us to manage both learning curves and deliverables.

**3. What Could Have Been Done Differently**

One area we could improve is in early engagement with external experts on the topic of increasing complexity. While we anticipated the difficulties in handling time series and sentiment analysis, earlier input from domain experts could have accelerated our research efforts and possibly simplified some workflows.

**4. Accuracy in Identifying Risks**

Overall, the team was highly accurate in predicting the key risks for Sprint 2. The risks identified around model accuracy, NLP education, and increasing complexity were directly encountered during the sprint and handled through preemptive mitigation.

**5. Unanticipated Risks**

Unanticipated technical difficulties arose in merging model IDs, which led to data duplication issues that required additional cleanup. While this was resolved, it did highlight the need for additional focus on technical risks related to data processing.

These insights serve as valuable lessons for future teams. Early risk identification and proactive mitigation were critical to our success, though earlier consultation with experts and more detailed planning around technical processes could have improved outcomes. For teams embarking on similar projects, balancing learning and execution, along with diligent technical planning, are essential for managing complexities.

## Sprint 3 Risks

| Risk | Description | Probability | Impact | Mitigation |
|---|---|---|---|---|
| **Incorrect Model Training** | Poor Usage of Variable Dependencies and algorithm training can result in a skewed model | Medium | Medium | Researching and testing multiple methodologies |
| **Poor Model ID merging with Recalls** | Inaccurate Model IDs may not result in a good merge with the Review dataset for the next steps | Medium | High | Cross Check if the model IDs are correctly extracted |
| **Incomplete Brand and Manufacturer Data** | Difficulty in extracting accurate brand and manufacturer names due to missing info in the Recall dataset. Some names had to be manually checked using images from the CPSC website. | Medium | High | Manually verify important entries and fill in missing info using reliable sources like CPSC |

Table 5: Sprint 3 Risks

During Sprint 3, our team proactively managed risks to ensure we stayed on track. Here's a summary of the identified risks and our approach:

1. **Risk Identification and Mitigation Planning**

   So, the incorrect Model Training was recognized as a first potential risk, given the complexities of variable selection and training approaches. To address this, the team explored and tested different methods to improve model accuracy. Next, we faced the risk of poor Model ID Merging with Recalls was flagged due to challenges in ensuring correct matches between recall and review data. This was managed by implementing double-checks on Model ID matches to maintain data consistency. Then we also had incomplete Brand and Manufacturer Data in the Recall dataset was identified as a risk due to missing brand and manufacturer names. This data deficiency required manual verification using images from CPSC to ensure reliability for downstream processes.

2. **What the Team Got Right**

   One of the things that the team got right is the early identification of Incomplete Brand and Manufacturer Data that enabled the team to allocate time for manual checks, ensuring data quality before analysis. Also, recognizing the risk of Incorrect Model Training allowed the team to avoid extensive rework by testing and validating various training methodologies early on.

3. **What Could Have Been Done Differently**

   To address the Incomplete Brand and Manufacturer Data risk more efficiently, the team could have allocated resources earlier for manual verification, reducing the bottleneck caused by data deficiencies.

**4. Accuracy in Identifying Risks**

The team was successful in predicting the major risks for Sprint 3. Early attention to model training methods, data merging accuracy, and filling data gaps prevented larger challenges as the sprint progressed.

**5. Unanticipated Risks**

Although the team anticipated needing manual verification for certain data points, the actual time required to validate and update brand and manufacturer names from the Recall dataset was greater than expected. This highlighted the importance of resource allocation for manual data processes.

## Sprint 4 Risks

| Risk | Description | Probability | Impact | Mitigation |
|---|---|---|---|---|
| **Incomplete Data for Brand and Manufacturer Extraction** | Missing or incomplete brand and manufacturer information in the Recall dataset may impact the accuracy of our merged dataset. | High | High | Manually verify and supplement missing entries using CPSC images to ensure data integrity. |
| **Model Performance with Imbalanced Data** | The Logistic Regression model may perform poorly due to class imbalance, as there are fewer recalls compared to reviews. | Medium | High | Use resampling techniques, including SMOTE, to balance the dataset and improve model performance. |
| **Inefficiency in Merging Processes** | The process of merging recall and review datasets may have inaccuracies due to inconsistent model IDs or incomplete information. | Medium | Medium | Conduct a thorough cross-check of model IDs during merging and use refactoring techniques to streamline and validate merging efficiency. |
| **Big Loop Execution Delays** | Running the Big Loop script for all categories sequentially may lead to significant delays due to system resource limitations. | High | High | Explore optimizing the script or parallelizing category execution to reduce runtime. |

Table 6: Sprint 4 Risks

Here's a summary of the identified risks and our approach for Sprint 4:

**1. Risk Identification and Mitigation Planning**

During Sprint 4, the team identified several risks that could impact project progress and outcomes. One critical risk was Incomplete Data for Brand and Manufacturer Extraction, where missing or incomplete details in the Recall dataset threatened the accuracy of the merged dataset. The team mitigated this by manually verifying and supplementing data using CPSC images. Another key risk was Model Performance with Imbalanced Data,

as the dataset contained significantly fewer recalls compared to reviews, potentially impacting model accuracy. The team applied resampling techniques such as SMOTE to address this issue. Inefficiency in Merging Processes was another identified risk, with inconsistencies in model IDs and incomplete information posing challenges during the dataset merging. The team mitigated this through thorough cross-checks and refactoring techniques. Lastly, the risk of Big Loop Execution Delays emerged due to system resource limitations when running the script sequentially. The team explored optimizing the script and parallelizing executions to improve runtime efficiency.

**2. What the Team Got Right**

The team successfully identified and addressed the risk of incomplete data by proactively planning manual verification and supplementing missing information. This approach ensured data integrity and improved the quality of the merged dataset. By applying resampling techniques to address dataset imbalance, the team effectively improved the model's ability to handle minority classes, enhancing classification accuracy. Additionally, the team's efforts in streamlining the merging process through refactoring and cross-checking ensured a more reliable dataset for downstream analysis.

**3. What Could Have Been Done Differently**

Allocating more resources earlier for manual data verification could have reduced the time spent resolving data deficiencies. For the Big Loop execution delays, considering optimization and parallelization techniques earlier in the sprint could have prevented the impact on timelines.

**4. Accuracy in Identifying Risks**

The team demonstrated strong accuracy in identifying the major risks for Sprint 4. Addressing data deficiencies, model performance issues, and inefficiencies in merging were all critical to maintaining project progress, and the team's mitigation strategies were timely and effective.

**5. Unanticipated Risks**

The risk of execution delays in the Big Loop script was greater than expected, as system resource limitations significantly slowed down the processing. While the team addressed this by optimizing the script and exploring parallel executions, earlier identification of this risk could have saved additional time.

## Sprint 5 Risks

| Risk | Description | Probability | Impact | Mitigation |
|---|---|---|---|---|
| **Big Loop Execution Delays** | Running the Big Loop script for all categories sequentially may lead to significant delays due to system resource limitations. | High | High | Explore optimizing the script or parallelizing category execution to reduce runtime. |

Table 7: Sprint 5 Risks

During Sprint 5, the team identified the risk of Big Loop Execution Delays emerged due to system resource limitations when running the script sequentially. The team explored optimizing the script and parallelizing executions to improve runtime efficiency. This was carried from Sprint 4.

# Appendix E: Agile Development

## Scrum Framework Team Approach


Figure 14: Sprint project dates.

The team started to adapt to the scrum framework as early as sprint 1 in result to the increasing complexities of the projects' datasets. While difficult at first, the agile structure allowed proactivity and versatility in direction-finding, specifically the synergistic approach of parallel workflow. The team pushed to conduct daily standups to track current activities, despite each having outside lives and occupations, it was conducted every 2-3 days. Using YouTrack was quite a doable task as our scrum master had a idea on the usage, the team also watched the videos on agile SCRUM framework in the Blackboard. Overall sprint 1 was interesting as we tried to practice different approaches on our project requirement and learned a lot on each step.

## Sprint 1 Lessons Learned

1. **Focus on learning NLP**

   We soon found out that not everyone was familiar with Natural Language Processing (NLP), so we dedicated ourselves to learning more about it. This commitment to self-education didn't just bridge our knowledge gaps; it transformed us into a more adaptable and robust team, equipped to tackle any technical issues that arose.

2. **Taking Charge of Potential Risks**

   Right from the start, we recognized that merging various datasets and diving into Natural Language Processing (NLP) without any previous experience would be our most significant challenges. By acknowledging these obstacles early on, we were able to think ahead, come up with solutions, and prepare alternate strategies. This proactive approach allowed us to address potential issues with data integration directly, preventing them from delaying our progress.

3. **Building a Stronger Team Through Effective Communication**

   Our project was as much about collaboration as it was about coding and data. We made a pact to communicate every 2-3 days, ensuring that everyone was informed and in sync. These regular meetings allowed us to face the project's challenges together and made sure no one was isolated with a problem.

4. **Using Tools to Enhance Project Management**

   The decision to use YouTrack as our project management tool proved to be pivotal. It enabled us to monitor our tasks and progress in real-time, fitting seamlessly with our agile methodology. This tool did more than organize our tasks—it made our entire workflow transparent and accessible to all team members.

**5. Things learnt from Data Challenges**

Throughout the project, we encountered numerous data issues, from integrating varied datasets to extracting precise information. Each challenge served as a learning moment, underscoring the need for meticulous data examination from the beginning and encouraging us to think outside the box to devise solutions.

One of the biggest takeaways from this sprint was understanding when to build solutions from scratch and when to lean on existing tools. While crafting custom solutions was intellectually rewarding, it was also time-consuming. In retrospect, tapping into external resources and expertise could have saved us time and enhanced our problem-solving toolkit.

## Sprint 2 Lessons Learned

**1. Team Efforts and User Story Identification**

During Sprint 2, our team focused primarily on enhancing the accuracy of our Natural Language Processing (NLP) model and managing the increasing complexity of the dataset. The User Stories were identified based on the project's evolving needs. We prioritized improving model accuracy by refining data extraction methods and integrating incident/recall data for better insights.

The team engaged in detailed discussions to break down these needs into actionable User Stories. Tasks such as manually cleaning datasets, merging model IDs, and expanding the scope of NLP education were defined. This breakdown helped us ensure that our User Stories aligned with the sprint's goal to improve model performance and complexity management.

**2. Performance and Task Management**

Managing the tasks during this sprint proved to be a mixed experience. On one hand, the team made considerable progress on specific technical tasks such as handling recall data, extracting model IDs, and beginning work on incident/recall modeling. However, we struggled with certain aspects, especially around maintaining dataset integrity during merging. A notable challenge was dealing with the sudden increase in data volume after the merge, which led to duplication issues that we hadn't anticipated.

The complexity of adding multiple variables—such as sentiment analysis, time series, and incident data—also introduced difficulties. While team members managed these tasks effectively, they required significant collaboration and trial-and-error to resolve issues around data accuracy and model input preparation.

**3. Successes**

One of the successes was how quickly the team adapted to the increased project complexity. Despite the new challenges of merging model ID data and adding NLP-based variables, the team was able to stay aligned with the project timeline. We also made significant progress in educating team members about NLP, which enabled them to contribute more effectively to the project.

**4. Areas for Improvement**

One major area for improvement is in our dataset merging approach. The decision to manually clean and merge datasets led to the unintended consequence of generating duplicate rows. Moving forward, we should explore more efficient automated methods for merging datasets and apply stricter validation checks before proceeding with training. Additionally, while we anticipated some issues with data integrity, the actual

number of duplicates and other anomalies exceeded our expectations. Implementing a more robust data validation process earlier in the sprint could have helped mitigate these issues.

**5. Lessons for Future Teams**

Future teams embarking on similar projects should plan for potential data-related risks and dedicate time to automate as much of the process as possible. It's critical to have clear and well-tested methods for handling dataset merges, particularly when working with large, complex datasets. Furthermore, setting aside more time for team education—especially in areas like NLP—will pay off in terms of long-term project success.

Finally, future teams should proactively identify and mitigate risks associated with increased programmatic complexity. This will ensure smoother sprint execution and help the team stay on track with their deliverables.

## Sprint 3 Lessons Learned

During Sprint 3, our team focused on developing critical components of the project and enhancing our process efficiency. Our goal was to train the predictive models, integrate product recall data with review data, and improve our extraction of brand and manufacturer information. Here's a summary of our team's journey through these tasks, including reflections on what went well and areas we can improve upon.

**1. User Story Identification and Task Management**

We began the sprint by breaking down our project's objectives into actionable user stories, focused on achieving key milestones such as accurate model training and recall data integration. Identifying user stories was relatively straightforward, as we already had a clear roadmap from prior sprints. Each story was aligned with broader project goals, ensuring that each sprint moved us toward a fully functional predictive model and recall data integration.

Although task assignment went smoothly, we realized mid-sprint that some stories needed additional breakdown into subtasks. For instance, "Brand and Manufacturer Data Extraction" required more time and granularity due to data inconsistencies.

**2. Task Execution and Team Performance**

Our team's performance was strong, with members demonstrating adaptability in the face of unexpected challenges, such as incomplete brand information in the recall dataset. Tackling this issue required extra research and manual verification, which team members took on effectively. A significant challenge was ensuring consistent model accuracy; multiple methodologies were tested to address dependencies between variables. The team's collaborative effort to test and validate different approaches highlighted a positive aspect of our teamwork. However, certain tasks took longer than expected due to unforeseen complexities, especially in managing data discrepancies between datasets. This pushed us to work late in the sprint, but our combined efforts kept us on track.

**3. Sprint Management: Challenges and Adaptations**

Managing tasks within the sprint posed moderate difficulty due to the data inconsistencies that required time-intensive manual verification. While we maintained flexibility to handle these issues, it highlighted the importance of building buffer time into the sprint plan to accommodate unexpected complications.

We effectively used our daily standups to communicate progress, address blockers, and adjust timelines. However, clearer initial scoping could have reduced some bottlenecks and allowed us to better anticipate resource needs.

**4. Successes**

Our cross-checks in merging model IDs were also effective, preventing data alignment issues from impacting analysis quality. This was a positive outcome resulting from lessons learned in previous sprints about the importance of thorough validation steps. The team also responded well to the challenges posed by incomplete brand and manufacturer data, using additional resources, such as images from the CPSC site, to verify information manually. This effort ensured high data quality, even with limited resources.

**5. Areas for Improvement and Future Recommendations**

Breaking down tasks more granularly would allow us to manage complexities better. Future teams might benefit from incorporating buffer time into sprint planning and considering the likelihood of data inconsistencies in the initial scope. Our experience reinforced the importance of standardized procedures for data validation. Establishing these protocols earlier in the project would streamline processes and save time in later sprints.

Allocating more time for manual verification tasks, especially when working with datasets prone to gaps, would improve efficiency. Future teams should consider additional team members or support for labor-intensive verification processes if similar datasets are used.

Overall, Sprint 3 was a productive sprint, with both successes and valuable insights gained. While we achieved our objectives, we learned the importance of flexibility, careful scoping, and robust validation protocols. These lessons are vital for any team considering a similar project, especially one involving the integration of multiple datasets with variable quality.

## Sprint 4 Lessons Learned

Sprint 4 offered valuable insights into team dynamics, project execution, and challenges associated with a data-intensive project. Here is a detailed narrative of our efforts and lessons learned:

**1. Identifying User Stories**

The team effectively identified user stories by closely aligning tasks with project goals and deliverables. We ensured that each story addressed key milestones, such as improving data accuracy, mitigating risks related to class imbalance, and streamlining the merging process for datasets. However, refining these stories took longer than anticipated due to ambiguities in the initial data requirements. Earlier stakeholder consultations could have clarified expectations and saved time.

**2. Team Performance**

The team collaborated efficiently, leveraging individual strengths to complete tasks such as manual verification of brand and manufacturer data, implementing resampling techniques, and optimizing computational processes. Task ownership was clear, which minimized redundancy and ensured steady

progress. Despite these strengths, time management emerged as a challenge, particularly when certain tasks, like manual verifications, required more effort than expected.

**3. Managing Sprint Activities**

Coordinating activities during the Sprint proved to be moderately challenging. While most tasks followed a clear roadmap, dependencies between tasks, such as merging datasets before applying the heuristic model, introduced delays when prior stages encountered issues. Weekly stand-ups and progress tracking mitigated some delays, but real-time updates could have improved responsiveness. Implementing task automation for repetitive activities, such as dataset checks, would have saved significant time.

**4. What Went Well**

The team worked effectively to address risks, particularly the imbalanced data challenge. The use of SMOTE demonstrated our ability to adapt machine learning techniques to project needs. Manually verifying missing brand and manufacturer data ensured higher dataset quality, critical for accurate analyses. Efforts to parallelize the Big Loop execution resulted in reduced runtime, demonstrating our capacity to improve system efficiency under tight timelines.

**5. What Could Have Been Done Differently**

Some risks, such as inefficiencies in merging processes and the complexity of the Big Loop, were only realized mid-Sprint. Conducting a thorough risk assessment at the Sprint's start could have helped anticipate these challenges earlier. Stronger coordination of interdependent tasks could have prevented bottlenecks. For example, a contingency plan for delays in data cleaning or merging could have minimized downstream impacts. Manual verification of brand and manufacturer data was time-consuming. Investing in automation or scripting for this step would have saved effort and time.

**6. Advice for Future Projects**

Early consultations with stakeholders to define data expectations and dependencies are essential for minimizing rework and confusion. Begin each Sprint with a robust risk assessment to preempt potential challenges and allocate resources effectively. Automating repetitive and labor-intensive tasks can significantly improve efficiency and reduce errors. Clearly map out dependencies between tasks to avoid bottlenecks and create contingency plans for critical paths. Combining rule-based logic with machine learning techniques can be more effective than solely relying on complex algorithms, especially when addressing class imbalance.

This Sprint underscored the importance of balancing thorough planning with adaptability. By reflecting on these lessons, future teams can build a more agile, efficient, and successful project workflow.

## Sprint 5 Lessons Learned

The team worked collaboratively and effectively, utilizing individual strengths to accomplish tasks such as manual verification of brand and manufacturer data, implementing resampling techniques, and optimizing computational workflows. Clear task ownership minimized redundancy and maintained steady progress. However, time management proved to be a challenge, especially when tasks like manual verifications demanded more effort than initially anticipated.

# References

## Works Cited


[1] J. L. Z. H. A. Y. X. C. J. M. Yupeng Hou, *Bridging Language and Items for Retrieval and Recommendation,* Amazon, 2024.

[2] A. Rea, "Product Safety Definition, Regulations & Importance," [Online]. Available: https://study.com/academy/lesson/product-safety-definition-lesson-quiz.html.

[3] selective, "PRODUCT RECALLS," [Online]. Available: https://www.selective.com/info-center/value-added-services/product-recall.

[4] U.S. Consumer Product Safety Commission, RECALL HANDBOOK, Bethesda, Maryland: U.S. Consumer Product Safety Commission, 2012.

[5] G. Bleaney, M. Kuzyk, J. Man, H. Mayanloo and H.R.Tizhoosh, "Auto-Detection of Safety Issues in Baby Products," University of Waterloo, Ontario, Canada, 2018.

[6] M. Fuchs, A. Jadhav, A. Jaishankar, C. Cauffman and G. Spanakis, "“What’s wrong with this product?” - Detection of product safety issues based on information consumers share online," Maastricht University, Maastricht, Netherlands, 2023.

[7] U.S. Consumer Product Safety Commission, "Small Parts Ban and Choking Hazard Labeling," [Online]. Available: https://www.cpsc.gov/Business--Manufacturing/Business-Education/Business-Guidance/Small-Parts-for-Toys-and-Childrens-Products.

[8] U.S. Consumer Product Safety Commission, "Water Beads," [Online]. Available: https://www.cpsc.gov/Safety-Education/Safety-Education-Centers/Water-Beads-Information-Center.

[9] U.S. Consumer Product Safety Commission, "CPSC Prohibits Certain Phthalates in Children’s Toys and Child Care Products," 20 October 2017. [Online]. Available: https://www.cpsc.gov/Newsroom/News-Releases/2018/CPSC-Prohibits-Certain-Phthalates-in-Childrens-Toys-and-Child-Care-Products.

[10] U.S. Consumer Product Safety Commission, "Safe Sleep – Cribs and Infant Products," [Online]. Available: https://www.cpsc.gov/SafeSleep.

[11] Children's Medical Centers of Fresno, "Protect Your Child from Toy Hazards," [Online]. Available: https://cmcfresno.com/blog/protect-your-child-from-toy-hazards/.

[12] U.S. Consumer Product Safety Commission, "Toy Safety Business Guidance," [Online]. Available: https://www.cpsc.gov/Business--Manufacturing/Business-Education/Toy-Safety.

[13] U.S. Consumer Product Safety Commission, "About SaferProducts.gov," [Online]. Available: https://www.saferproducts.gov/About.

[14] U.S. Consumer Product Safety Commission, "Recalls," [Online]. Available: https://www.cpsc.gov/Recalls.

[15] U.S. Consumer Product Safety Commission, "Testing & Certification," [Online]. Available: https://www.cpsc.gov/Business--Manufacturing/Testing-Certification.

[16] S. Das, Quality Characterisation of Apparel, New Delhi: Woodhead publishing india pvt ltd, 2009.

[17] United States Consumer Product Safety Commission, "About CPSC," Bethesda, Maryland.

[18] Amazon, "What is Natural Language Processing (NLP)?," [Online]. Available: https://aws.amazon.com/what-is/nlp/.

[19] Amazon, "What is Sentiment Analysis?," [Online]. Available: https://aws.amazon.com/what-is/sentiment-analysis/.

[20] geeksforgeeks, "TF – IDF for Bigrams & Trigrams," 27 September 2019. [Online]. Available: https://www.geeksforgeeks.org/tf-idf-for-bigrams-trigrams/.

[21] geeksforgeeks, "Random Forest Algorithm in Machine Learning," 12 July 2024. [Online]. Available: https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/.

[22] U.S. Food and Drug Administration, "Phthalates in Cosmetics," 19 May 2022. [Online]. Available: https://www.fda.gov/cosmetics/cosmetic-ingredients/phthalates-cosmetics.

[23] K. Rogers, "What is Amazon ASIN number & how to get it?," [Online]. Available: https://www.datafeedwatch.com/blog/amazon-asin-number-what-is-it-and-how-do-you-get-it.

[24] Lenovo, "What are stop words?," [Online]. Available: https://www.lenovo.com/us/en/glossary/stop-words/.

[25] Computer Hope, "Regex," 16 August 2024. [Online]. Available: https://www.computerhope.com/jargon/r/regex.htm.

[26] U.S. Environmental Protection Agency (EPA), "Assessing and Managing Chemicals under TSCA," 20 June 2024. [Online]. Available: https://epa.gov/assessing-and-managing-chemicals-under-tsca/phthalates.

[27] U.S. Consumer Product Safety Commission (CPSC), "Red Toolbox Recalls Stanley-Branded Jr. Kids Garden Sets Due to Lead Poisoning Hazard; Violation of Federal Ban for Lead in Paint; Sold Exclusively by Costco Wholesale," 12 September 2024. [Online]. Available: https://www.cpsc.gov/Recalls/2024/Red-Toolbox-Recalls-Stanley-Branded-Jr-Kids-Garden-Sets-Due-to-Lead-Poisoning-Hazard-Violation-of-Federal-Ban-for-Lead-in-Paint-Sold-Exclusively-by-Costco-Wholesale.

[28] U.S. Consumer Product Safety Commission (CPSC), "HALO 1000 Portable Power Stations Recalled Due to Serious Fire and Burn Hazards; One Death Reported; Imported by ZAGG; Sold by ACG, QVC and ZAGG," 29 August 2024. [Online]. Available: https://www.cpsc.gov/Recalls/2024/HALO-1000-Portable-Power-Stations-Recalled-Due-to-Serious-Fire-and-Burn-Hazards-One-Death-Reported-Imported-by-ZAGG-Sold-by-ACG-QVC-and-ZAGG.

[29] U.S. Consumer Product Safety Commission (CPSC), "The Good and the Beautiful Recalls Math 1 and Math 3 Boxes Due to Laceration Hazard (Recall Alert)," 25 May 2023. [Online]. Available:

https://www.cpsc.gov/Recalls/2023/The-Good-and-the-Beautiful-Recalls-Math-1-and-Math-3-Boxes-Due-to-Laceration-Hazard-Recall-Alert.

[30] U.S. Consumer Product Safety Commission (CPSC), "Foiresoft Zebra Roller Blinds Recalled Due to Strangulation and Entanglement Hazards; Sold Exclusively on Amazon.com by Softfunch (Recall Alert)," 5 December 2023. [Online]. Available: https://www.cpsc.gov/Recalls/2024/Foiresoft-Zebra-Roller-Blinds-Recalled-Due-to-Strangulation-and-Entanglement-Hazards-Sold-Exclusively-on-Amazon-com-by-Softfunch-Recall-Alert.

[31] Makridis, G., Mavrepis, P. & Kyriazis, D. A deep learning approach using natural language processing and time-series forecasting towards enhanced food safety. Mach Learn 112, 1287–1313 (2023). https://doi.org/10.1007/s10994-022-06151-6

[32] freshbeautyfix.com, "BABY SKINCARE ESSENTIALS FOR FIRST TIME PARENTS," 3 March 2019. [Online]. Available: https://freshbeautyfix.com/2019/03/03/baby-skincare-essentials/.

[33] Gillis, A. S. (2024, August). What is natural language processing (NLP)? Retrieved from techtarget.com: https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP

[34] geeksforgeeks.org. (2024, January 31). NLP | How tokenizing text, sentence, words works. Retrieved from geeksforgeeks.org: https://www.geeksforgeeks.org/nlp-how-tokenizing-text-sentence-words-works/

[35] Ganesan, K. (n.d.). What are Stop Words? Retrieved from opinosis-analytics.com: https://www.opinosis-analytics.com/knowledge-base/stop-words-explained

[36] Awan, A. A. (2023, September 13). What is Named Entity Recognition (NER)? Methods, Use Cases, and Challenges. Retrieved from datacamp.com: https://www.datacamp.com/blog/what-is-named-entity-recognition-ner

[37] geeksforgeeks.org. (2019, September 27). TF – IDF for Bigrams & Trigrams. Retrieved from geeksforgeeks.org: https://www.geeksforgeeks.org/tf-idf-for-bigrams-trigrams/#

[38] Rotulo, M. (2022, April 10). Tweets sentiment analysis with RoBERTa. Retrieved from medium.com: https://medium.com/@monica.rotulo/tweets-sentiment-analysis-with-roberta-1f30cf4e1035

[39] Dutta, N. (2024, August 13). Fuzzy String Matching – A Hands-on Guide. Retrieved from analyticsvidhya.com: https://www.analyticsvidhya.com/blog/2021/07/fuzzy-string-matching-a-hands-on-guide/

[40] geeksforgeeks.org. (2017, October 22). How to get synonyms/antonyms from NLTK WordNet in Python? Retrieved from geeksforgeeks.org: https://www.geeksforgeeks.org/get-synonymsantonyms-nltk-wordnet-python/

[41] Jacob Murel, E. K. (2024, January 19). What is bag of words? Retrieved from ibm.com: https://www.ibm.com/topics/bag-of-words

[42] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16).

Association for Computing Machinery, New York, NY, USA, 785–794. https://doi.org/10.1145/2939672.2939785

[43] Mikolov, T., Chen, K., Corrado, G.S., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. International Conference on Learning Representations.

[44] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[45] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: a highly efficient gradient boosting decision tree. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 3149–3157.

[46] Ruder, S. (2016). An overview of gradient descent optimization algorithms. ArXiv, abs/1609.04747.

[47] IBM, “What Is Logistic Regression?,” IBM, [Online]. Available: https://www.ibm.com.

[48] “Gradient Boosting,” V7 Labs, [Online]. Available: https://www.v7labs.com.

[49] “XGBoost,” V7 Labs, [Online]. Available: https://www.v7labs.com.

[50] “LightGBM,” V7 Labs, [Online]. Available: https://www.v7labs.com.

[51] “CatBoost,” V7 Labs, [Online]. Available: https://www.v7labs.com.

[52] “F1-Score and ROC-AUC,” IBM, [Online]. Available: https://www.ibm.com.

[53] R. Y. Wang and D. M. Strong, “Beyond Accuracy: What Data Quality Means to Data Consumers,” *Journal of Management Information Systems*, vol. 12, no. 4, pp. 5–33, Mar. 1996, doi: https://doi.org/10.1080/07421222.1996.11518099.

[54] J. M. Ph.D. and E. Kavlakoglu, "What is a confusion matrix?," 19 January 2024. [Online]. Available: https://www.ibm.com/topics/confusion-matrix.