January 21, 2025

C964: Computer Science Capstone

Seung Cho (WGU ID: 009661325)

Task 2 parts A, B, C and D

Part A: Letter of Transmittal

January 21st, 2025

R&D Sitewide Manager
The Company
123 Street Road
City, ST 45678

Dear R&D Sitewide Manager,

The current inventory system relies on manual efforts using shared Excel files to manage thousands of chemicals across multi-floor laboratories. As an employee of the Company, I have observed challenges such as misplaced chemicals, increased safety risks with misplaced dangerous combinations of chemicals, and cost and time demands for upkeep, resulting in operational inefficiencies. To ensure the safety of all lab members and improve efficiency, an updated system that complies with OSHA regulations should be implemented while optimizing chemical storage and tracking processes.

Based on these requirements, I propose a machine learning-based solution that can significantly improve the inventory system by offering features such as chemical organization, storage location recommendations, and expiration notifications. The chemical organization will rely on attributes like hazard classification, ownership, lab location, and owner's team. By leveraging a Random Forest algorithm, this solution optimizes storage allocation, reduces human error, and ensures safety compliance in the work environment for all involved.

The project requires minimal additional resources since the only required application tool is a Python IDE (e.g., PyCharm from JetBrains). The current inventory system is in CSV format, which was recently updated during the site-wide year-end organizational effort. The application will be developed as a program housed on a locally shared server. Key deliverables include a trained machine learning model, a notification system for expired chemicals, and a comprehensive evaluation report. The estimated cost is $249 per year for PyCharm Professional, and the projected deadline is February 28th, 2025.

As an employee of the Company, I have firsthand experience with the challenges my peers face daily in managing chemical inventory. With firsthand knowledge and expertise in wet chemistry and machine learning, I can provide a tailored solution that will scale with the Company's growth.

Please review the attached proposal for further details. Thank you for your time, and I look forward to your feedback. Feel free to reach out to me with any questions.

Sincerely,

Seung Ri Cho

Seung Ri Cho
R&D Chemist – MHM Group
The Company

Part B: Project Proposal Plan

Project Summary

This project addresses a real-world challenge that Research and Development (R&D) labs face in managing chemical inventory. Historically, the Company's chemical inventory management relied on manual data entry into shared Excel sheets. This method has proven inadequate as the Company's assets grew to include thousands of chemicals spread across multiple labs, resulting in redundancies and errors, increased safety risks, and operational costs.

The updated inventory system will meet the client's needs as follows:

- Implementing a chemical inventory machine learning model to classify chemicals based on hazard type, ownership, team group, and lab assignment attributes.
- Assigning locations for incoming chemicals using the model's predictions.
- Providing notifications for expired chemicals based on the Company's eight-year rule from the received date.

The application will benefit the organization by improving accuracy, reducing waste, and ensuring compliance with safety regulations. Chemicals will be correctly assigned to storage locations in compliance with hazard type and proximity to the employee's workstation. Each laboratory has storage areas designated by hazard category, and the Random Forest model will facilitate the automatic placement of chemicals based on these complex requirements.

The deliverables include the finished application with a trained machine-learning model, a user guide, and performance evaluation metrics.

Data Summary

This project is purely academic and does not involve proprietary or confidential company information. Raw data will be generated using ChatGPT to simulate attributes from the actual chemical inventory system in CSV format. The attributes include the chemical name, CAS number (a unique identifier provided by the Chemical Abstracts Service), hazard classification, container type, amount, owner initials, manufacturer, and received date. Hazard classifications and chemical names are randomized but accurate; all other attributes are randomly generated.

The dataset will be preprocessed and cleaned. Additional data, such as chemical owners, hazard categories, and storage IDs, from separate CSV files, will be merged with the inventory dataset based on shared attributes. String attributes will be encoded and converted to integers to process the Random Forest model. The dataset will be divided into training (70%) and testing (30%) subsets. Missing attributes will be recognized and transformed by the application if necessary. The deployed application will allow the model to be retrained as the dataset grows or changes.

Implementation

The development will follow the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology as stated below:

1. Business Understanding: The inventory assignment and tracking accuracy will be maintained above 90% by adjusting the machine learning model's parameters. The application will identify expired chemicals with at least 90% accuracy and suggest storage areas for incoming chemicals.

2. Data Understanding: In CSV format, the existing chemical inventory dataset will be analyzed to generate a dataset of approximately 1,000 samples. Attributes like chemical owners, teams, and lab areas will be identified as relevant features.

3. Data Preparation: String-type attributes will be encoded into numerical values. Missing values will be formatted, and relevant CSV data will be merged with the inventory system. The dataset will be split into training (70%) and testing (30%) subsets.

4. Modeling: The preprocessed data will be fed into a Random Forest algorithm. Hyperparameter tuning will optimize model performance. Additional data will be generated to increase sample robustness.

5. Evaluation: The model's performance will be validated using precision, recall, F1-score, and accuracy metrics. The notification system for expired chemicals will also be tested for reliability.

6. Deployment: The Python application with the trained Random Forest model will be deployed on a locally shared database with a fully developed inventory file and notification module. The lab personnel will be provided with a user guide and in-person training.

Timeline

Milestone or deliverable	Duration (hours or days)	Projected start date	Anticipated end date
Initial setup and planning of the project via consulting with system users. Obtain project approval by the R&D manager. Collect the dataset and begin prototyping the ML model.	5 business days	January 27th 2025	January 31st 2025
Collect and compose all relevant data sources, including chemical inventory, chemical storage information, owner/user, etc. Preprocess as needed.	5 business days	February 3rd 2025	February 7th 2025
Feed the preprocessed dataset to the ML model and prototype the framework of the Random Forest algorithm with hyperparameters.	10 business days	February 10th 2025	February 14th 2025
Complete the training and confirm that the evaluation of the model meets satisfactory accuracy and performance	5 business days	February 24th 2025	February 28th 2025
Compose the front end for the users. Finalize the application for review.	5 business days	March 3rd 2025	March 7th 2025
Application deployment after a final sign-off from the R&D manager and IT manager. The train involved lab personnel on the model and the application's navigation. Generate a post-implementation report.	10 business days	March 10th 2025	March 21st 2025

Evaluation Plan

Each stage of development will undergo testing and validation to meet the requirements. Data quality will be ensured during data preparation through checks for missing values, normalization consistency, and encoding accuracy. After modeling, the results from hyperparameter tuning will be validated using cross-validation. Before deployment, real-world data will be tested to ensure the system can classify chemicals and trigger notifications.

Upon project completion, the model's accuracy (>90%) will be validated using precision, recall, and F1-score metrics. The notification system will achieve at least 90% accuracy. The reduction in expired chemicals will be quantified and evaluated quarterly, targeting a minimum 20% decrease by the end of 2025. User feedback will guide future improvements.

Resources and Costs

Resource	Description	Cost
Laptop	Workstation already provided by the Company as a current employee	$0 added cost
Development Tool	Python IDE with machine learning libraries (scikit-learn, pandas, matplotlib, and etc.)	$249/year
Data Set	An existing inventory system in a CSV format utilized internally	$0 added cost
Human Resource	IT manager, Data scientist, and R&D chemists as in-house consultants, already employed in the Company	$0 added cost
Miscellaneous Cost	Necessary training, maintenance, and incidental costs	$5,000/year
	Total	$5,249/year

Part C: Application

This part of the document is intentionally left blank. Please see the user guide at the bottom of this documentation to operate the Python application.

Part D: Post-implementation Report

Solution Summary

The previous chemical inventory system encompassing all laboratories in the Company relied on manual input and upkeep, resulting in lost or expired items, increased hazards with misplaced chemicals, and increased operational costs. The machine learning model with the Random Forest algorithm aided the organizational effort to combat thousands of chemicals spread across seven labs. The existing chemicals with important attributes such as owner name, lab location, owner's team, and hazard category determined the storage placement. Furthermore, the application can present the list of expired chemicals (8 years past the received date) per the chemical owner's request.

This Python application presents 5 activities to the user. The first option trains the Random Forest model with the inventory in CSV format, which can readily replace the pre-trained model in the package. The model is then cross-examined with a set of hyperparameters and trained with the preprocessed dataset, which would then provide an evaluation that reports an accuracy of 97% in the f1-score. The second option from the main menu helps identify the best storage location for the incoming chemicals based on the train model. The third option debugs predictions by applying inputs to see if the model produces the expected result. The fourth option notifies the user of the list of expired chemicals. The fifth option exits the application.

The application helped the lab personnel by automating meaningful storage recommendations on existing and incoming chemicals based on prioritized attributes. The solution allocated human efforts to improve work efficiency within the lab environment. Furthermore, the chemicals were sorted into optimal storage conditions, improving safety and tracking accuracy. The owners could check what expired chemicals allowed space for new chemicals.

Data Summary

The raw data was randomly generated with the help of ChatGPT, which closely mimicked the real-life example. This project was created as an academic exercise, and no proprietary information was used. The owner's name, hazard classification, and storage locations were separately loaded, then merged, and encoded as part of the preprocessing step. The 70% of clean, preprocessed data was then fed into the Random Forest algorithm, in which the hyperparameter tuning enabled the optimization of model performance. A more chemical list of about a thousand items was generated to improve the robustness of the sample size and increase the accuracy of the f1 score to 97%, which was tested with 30% of the dataset.

```
param_dist = {
    'n_estimators': [300, 500, 1000],
    'max_depth': [5, 10, 20, None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 5],
    'max_features': ['sqrt', 'log2', None]
}
```

A set list of hyperparameters for randomized search

Storage ID	Lab Number	Hazard Category
1	1	Health Hazard
2	1	Corrosion
3	1	Flame
4	1	Exclamation Mark
5	1	Toxic
6	2	Health Hazard
7	2	Corrosion
8	2	Flame
	•	
	•	
	•	
33	7	Flame
34	7	Exclamation Mark
35	7	Toxic

ChemicalInventoryManagement\data\storage_locations.csv

The trained model was utilized to optimize the physical location of existing chemicals and automatically allow users to determine optimal placement in the work environment. The features such as ownership (26 total), hazard classification (10), lab location (7), and owner's group (7) were accounted for in the determinization of storage location (each lab has storage catered to 10 unique hazard classifications). The chemical inventory dataset is planned to be periodically backed up as it is due to change and scale up over time.

Machine Learning

- What: The Random Forest is a supervised learning algorithm specializing in binary decisions, a.k.a. trees in a forest, to reach a singular and more accurate prediction. This application classifies chemicals to optimal placement based on complex attributes such as the owner's workstation and hazard classification. Training through the existing chemical inventory list helps organize existing chemicals.
- How: 70% of the inventory dataset comprised about a thousand items were fed into the random forest model, where attributes such as employee's ID, hazard classification, lab number, and employee's team were brought forth to predict the best storage placement. The hyperparameter tuning technique with randomized search checks three options per hyperparameter to improve overall accuracy. 30% of the inventory dataset is then used to test the model. The trained model is saved in the joblib file, which can be accessed as needed.
- Why: The Random Forest model is best at handling large data types and mitigating overfitting, which would ensure robust predictions. All the attributes could be easily encoded as part of preprocessing, and the model was easy to maneuver with the hyperparameters. All indicate accurate predictions, which vastly improved organization in the physical world. The model can easily scale with the growth.

Validation

The model achieved an f1 score accuracy of 97% using the dataset comprised of about a thousand chemicals, which far exceeded the required baseline of 90%. 70% of the dataset was used in training and another 30% for testing. The model was then tested with 20 unique inputs, which recommended the optimal storage solution closest to the chemist's workstation and hazard category. 4 of which is available as option 2, a debugging prediction. The list of expired chemicals was cross-examined with the chemical inventory in the CSV file format to confirm 100% accuracy.

Visualizations

When the application starts, a main menu is presented as displayed below:

```
--- Main Menu ---
1. Train the model
2. Register a new chemical
3. Debug Prediction
4. Check Expired Chemicals
5. Exit
```

Option 1 trains the Random Forest model with the dataset saved in the data file and presents the user with the evaluation report.

```
Optimized Random Forest model trained successfully!

Model saved to '../models/random_forest.joblib'

Model Accuracy: 0.97

-------Classification Report-------
              precision    recall  f1-score   support

           1       1.00      1.00      1.00         3
           2       1.00      1.00      1.00         2
           3       1.00      1.00      1.00        15
           4       1.00      1.00      1.00        32
           5       1.00      1.00      1.00        11
           6       1.00      1.00      1.00         2
           7       1.00      1.00      1.00         3
           8       1.00      1.00      1.00         7
           9       1.00      0.96      0.98        24
          10       0.92      1.00      0.96        12
          11       1.00      1.00      1.00         1
          12       1.00      1.00      1.00         2
          13       1.00      1.00      1.00         3
          14       1.00      1.00      1.00        16
          15       1.00      1.00      1.00         9
          16       1.00      1.00      1.00         1
          18       1.00      1.00      1.00         8
          19       1.00      1.00      1.00        17
          20       1.00      1.00      1.00        11
          22       1.00      1.00      1.00         1
          23       1.00      1.00      1.00         8
          24       1.00      0.94      0.97        16
          25       0.90      1.00      0.95         9
          26       0.00      0.00      0.00         2
          27       0.33      1.00      0.50         1
          28       1.00      1.00      1.00         5
          29       1.00      1.00      1.00        20
          30       1.00      1.00      1.00        12
          31       0.00      0.00      0.00         5
          32       0.38      1.00      0.55         3
          33       1.00      1.00      1.00         5
          34       1.00      1.00      1.00        19
          35       1.00      1.00      1.00         7

    accuracy                           0.97       292
   macro avg       0.89      0.94      0.91       292
weighted avg       0.96      0.97      0.96       292
```

Option 2 returns the optimal chemical storage option based on the trained model based on the series of inputs.

```
Do you want to retrain the model before predicting? (yes/no): no

--- Register a New Chemical ---
Enter the chemical name (ex. Acetone): Acetone

Hazard Classification: 1) Carcinogenic  2) Caustic  3) Corrosive  4) Flammable  5) Generally Safe  6) Harmful  7) Irri
tant  8) Oxidizer  9) Toxic  10) Explosive
Enter the hazard classification (ex.4 for Flammable): 4
Enter the owner's ID (ex. 26): 26

Lab Assignment


Datasets loaded successfully!
     First Name  Last Name  Lab Number
0          John      Smith           1
1          Mary    Johnson           1
2        Robert   Williams           2
3      Patricia      Jones           2
4         David       Zoro           6
5         Linda        Lee           3
6        Thomas      Clark           4
7         Susan      Davis           6
8         Karen     Garcia           5
9       Michael  Rodriguez           5
10        Nancy      Baker           5
11      Charles     Wilson           3
12       Daniel      Lewis           2
13      Barbara   Martinez           2
14    Elizabeth   Gonzalez           1
15        James      White           1
16      Anthony      Moore           4
17       Sandra     Harris           7
18      Kenneth   Robinson           6
19      Deborah      Davis           7
20      Richard   Campbell           7
21  Christopher      Allen           3
22      Jessica   Mitchell           4
23       George     Walker           4
24         Lisa       Hall           2
25        Seung        Cho           1
Enter the lab number: 1

List of Groups: 1)MHM  2)ASD  3)VUE  4)THICK  5)REM
Enter the group number (ex. 1 for MHM): 1

Input Data for Prediction: {'Hazard Class Encoded': '4', 'Lab Number': '1', 'name ID': '26', 'Group Encoded': '1'}
[3]

Chemical 'Acetone' should be stored in: 3
```

Option 4 returns the list of expired chemicals by the owners, per request.

```
--- Main Menu ---
1. Train the model
2. Register a new chemical
3. Debug Prediction
4. Check Expired Chemicals
5. Exit

Enter your choice: 4


Datasets loaded successfully!

 Any chemicals that were received after 8 years ago is considered expired.
What is the owner's initials? SC

--- Expired Chemicals ---
              Chemical Name Hazard Classification Received Date
166     silver subfluoride              Irritant    2015-04-09
255  Arsenic pentafluoride                 Toxic    2015-04-09
347   Barium metaphosphate              Irritant    2015-05-11
896             Octadecane             Flammable    2015-05-11
820         Fluoroimidogen        Generally Safe    2015-08-18
748      Copper(I) bromide                 Toxic    2015-09-24
595     Cerium(III) iodide              Irritant    2015-10-28
163         silver oxalate             Explosive    2016-01-05
526       Calcium fluoride        Generally Safe    2016-01-12
421       Cadmium arsenide                 Toxic    2016-03-15
214       Sodium aluminate             Corrosive    2016-07-08
321     Barium perchlorate              Oxidizer    2016-09-28
617   Chlorine perchlorate              Oxidizer    2016-10-10
553       Calcium peroxide              Oxidizer    2017-01-14

SC has 14 expired chemicals.
```

User Guide

Include an enumerated (steps 1, 2, 3, etc.) guide to execute and use your application.

- Computer software requirements:
  - Python 3
  - Pip
  - Scikit-learn
  - Numpy
  - Joblib
  - Pytest
  - Pandas

- The application has the following files:

```
/ChemicalInventoryManagement
        /data
                chemical_inventory.csv
                hazard_classification.csv
                owner_names.csv
                storage_locations.csv
        /models
                random_forest.joblib
        /src
                /ml
                        __init__.py
                        predict.py
                        train.py
                __init__.py
                main.py
                preprocess.py
                utils.py
        README.md
        requirements.txt
        Capstone_ChemicalInventoryManagment_Documentation_SCho.pdf
```

1. Open a command prompt on the computer.

2. Navigate to the project directory, which contains the main.py file.
   e.g. C:\users\userId\...\ChemicalInventoryManagement\src

3. Type in the command to initiate the application: python main.py
4. The main menu has 5 options:
   1. Train the model

2. Register a new chemical
3. Debug Prediction
4. Check Expired Chemicals
5. Exit

   Navigate to the point of interest.

5. Enter ‘5’ to exit the application.

Note to Evaluators: Option 2 provides an example of an input to test codes, but if you are interested in testing different combinations of inputs, please check the CSV files, which provide the relationship between attributes and storage ID (a.k.a. location + hazard type)

Reference Page

No Reference was used.